

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7150108号
(P7150108)

(45)発行日 令和4年10月7日(2022.10.7)

(24)登録日 令和4年9月29日(2022.9.29)

(51)国際特許分類

F I

A 6 3 F	13/426 (2014.01)	A 6 3 F	13/426
A 6 3 F	13/55 (2014.01)	A 6 3 F	13/55
A 6 3 F	13/803 (2014.01)	A 6 3 F	13/803
A 6 3 F	13/5258(2014.01)	A 6 3 F	13/5258
A 6 3 F	13/53 (2014.01)	A 6 3 F	13/53

請求項の数 15 (全35頁) 最終頁に続く

(21)出願番号	特願2021-122597(P2021-122597)	(73)特許権者	000233778 任天堂株式会社 京都府京都市南区上鳥羽鉾立町1番地1
(22)出願日	令和3年7月27日(2021.7.27)	(74)代理人	100158780 弁理士 寺本 亮
(62)分割の表示	特願2019-519823(P2019-519823)の分割	(74)代理人	100121359 弁理士 小沢 昌弘
原出願日	平成29年5月22日(2017.5.22)	(74)代理人	100130269 弁理士 石原 盛規
(65)公開番号	特開2021-178196(P2021-178196A)	(72)発明者	矢吹 光佑 京都府京都市南区上鳥羽鉾立町1番地1 任天堂株式会社内
(43)公開日	令和3年11月18日(2021.11.18)	(72)発明者	宮本 健 京都府京都市南区上鳥羽鉾立町1番地
審査請求日	令和3年8月25日(2021.8.25)		最終頁に続く

(54)【発明の名称】 ゲームプログラム、情報処理装置、情報処理システム、および、ゲーム処理方法

(57)【特許請求の範囲】

【請求項1】

情報処理装置のコンピュータに、
表示画面上の位置に対応する位置であって、ポインティングデバイスによって検出される入力位置を繰り返し取得する取得ステップと、
前記ポインティングデバイスに対する移動操作入力~~が開始されたとき~~の入力位置が所定領域外であることを少なくとも条件として、前記移動操作入力~~が開始されたとき~~の入力位置に基づいて第1基準位置を設定する第1基準位置設定ステップと、
前記第1基準位置と、前記移動操作入力~~が開始されてから~~継続して行われる入力の入力位置とに基づいて、仮想ゲーム空間においてオブジェクトを第1移動処理で移動させる第1移動処理ステップと、
前記ポインティングデバイスに対する移動操作入力~~が開始されたとき~~の入力位置が前記所定領域内であることを少なくとも条件として、前記移動操作入力~~が開始されたとき~~の入力位置に基づいて第2基準位置を設定する第2基準位置設定ステップと、
前記第2基準位置と、前記移動操作入力~~が開始されてから~~継続して行われる入力の入力位置とに基づいて、前記仮想ゲーム空間において前記オブジェクトを前記第1移動処理とは異なる第2移動処理で移動させる第2移動処理ステップとを~~実行させ、~~
前記第1移動処理においては、前記第1基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、
前記第2移動処理においては、前記第2基準位置と前記入力位置との間の距離に基づい

10

20

て、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、
前記オブジェクトの移動方向の変化量が同じである場合、前記第2移動処理における前記オブジェクトの移動速度は、前記第1移動処理における前記オブジェクトの移動速度よりも大きい、ゲームプログラム。

【請求項2】

情報処理装置のコンピュータに、
表示画面上の位置に対応する位置であって、ポインティングデバイスによって検出される入力位置を繰り返し取得する取得ステップと、

前記ポインティングデバイスに対する移動操作入力が始まったときの入力位置が所定領域外であることを少なくとも条件として、前記移動操作入力が始まったときの入力位置に基づいて第1基準位置を設定する第1基準位置設定ステップと、

前記第1基準位置と、前記移動操作入力が始まってから継続して行われる入力の入力位置とに基づいて、仮想ゲーム空間においてオブジェクトを第1移動処理で移動させる第1移動処理ステップと、

前記ポインティングデバイスに対する移動操作入力が始まったときの入力位置が前記所定領域内であることを少なくとも条件として、前記移動操作入力が始まったときの入力位置に基づいて第2基準位置を設定する第2基準位置設定ステップと、

前記第2基準位置と、前記移動操作入力が始まってから継続して行われる入力の入力位置とに基づいて、前記仮想ゲーム空間において前記オブジェクトを前記第1移動処理とは異なる第2移動処理で移動させる第2移動処理ステップとを実行させ、

前記第1移動処理においては、前記第1基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記第2移動処理においては、前記第2基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記オブジェクトの移動速度が同じである場合、前記第2移動処理における前記オブジェクトの前記移動方向の変化量は、前記第1移動処理における前記オブジェクトの前記移動方向の変化量よりも大きい、ゲームプログラム。

【請求項3】

前記所定領域は、前記ポインティングデバイスの入力面において、前記表示画面において前記オブジェクトが表示される位置に対応する位置を含むように設定される、請求項1または請求項2に記載のゲームプログラム。

【請求項4】

前記第1移動処理および前記第2移動処理を実行可能な所定期間において、前記オブジェクトが前記表示画面上の所定位置に固定的に配置されるように前記仮想ゲーム空間の画像を前記表示画面に表示する画像表示ステップを前記コンピュータにさらに実行させる、請求項3に記載のゲームプログラム。

【請求項5】

前記第1移動処理および前記第2移動処理を実行可能な所定期間において、前記所定領域を表す画像を含まないゲーム空間の画像を前記表示画面に表示する画像表示ステップを前記コンピュータにさらに実行させる、請求項1から請求項3のいずれか1項に記載のゲームプログラム。

【請求項6】

前記所定領域に対する入力が始まってから継続して行われる入力の入力位置から所定距離内となるように前記基準位置を変更する基準位置変更ステップを前記コンピュータにさらに実行させる、請求項1から請求項5のいずれか1項に記載のゲームプログラム。

【請求項7】

前記移動操作入力が始まったときの入力位置が前記所定領域内である場合、当該移動操作入力が始まってから継続して行われる入力が所定の条件を満たす場合、前記オブジェクトに対する所定の処理を実行する処理実行ステップを、前記コンピュータにさらに実行させる、請求項1から請求項6のいずれか1項に記載のゲームプログラム。

【請求項 8】

前記処理実行ステップにおいては、前記所定の処理として、前記オブジェクトの移動に関する処理が実行される、請求項 7 に記載のゲームプログラム。

【請求項 9】

前記処理実行ステップにおいては、前記移動操作入力開始されてから継続して行われる入力の終了に応じて前記所定の処理が実行される、請求項 7 または請求項 8 に記載のゲームプログラム。

【請求項 10】

表示画面の位置に対応する位置であって、ポインティングデバイスによって検出される入力位置を繰り返し取得する取得手段と、

前記ポインティングデバイスに対する移動操作入力開始されたときの入力位置が所定領域外であることを少なくとも条件として、前記移動操作入力開始されたときの入力位置に基づいて第 1 基準位置を設定する第 1 基準位置設定手段と、

前記第 1 基準位置と、前記移動操作入力開始されてから継続して行われる入力の入力位置とに基づいて、仮想ゲーム空間においてオブジェクトを第 1 移動処理で移動させる第 1 移動処理手段と、

前記ポインティングデバイスに対する移動操作入力開始されたときの入力位置が前記所定領域内であることを少なくとも条件として、前記移動操作入力開始されたときの入力位置に基づいて第 2 基準位置を設定する第 2 基準位置設定手段と、

前記第 2 基準位置と、前記移動操作入力開始されてから継続して行われる入力の入力位置とに基づいて、前記仮想ゲーム空間において前記オブジェクトを前記第 1 移動処理とは異なる第 2 移動処理で移動させる第 2 移動処理を実行する第 2 移動処理手段とを備え、

前記第 1 移動処理においては、前記第 1 基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記第 2 移動処理においては、前記第 2 基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記オブジェクトの移動方向の変化量が同じである場合、前記第 2 移動処理における前記オブジェクトの移動速度は、前記第 1 移動処理における前記オブジェクトの移動速度よりも大きい、情報処理装置。

【請求項 11】

表示画面上の位置に対応する位置であって、ポインティングデバイスによって検出される入力位置を繰り返し取得する取得手段と、

前記ポインティングデバイスに対する移動操作入力開始されたときの入力位置が所定領域外であることを少なくとも条件として、前記移動操作入力開始されたときの入力位置に基づいて第 1 基準位置を設定する第 1 基準位置設定手段と、

前記第 1 基準位置と、前記移動操作入力開始されてから継続して行われる入力の入力位置とに基づいて、仮想ゲーム空間においてオブジェクトを第 1 移動処理で移動させる第 1 移動処理手段と、

前記ポインティングデバイスに対する移動操作入力開始されたときの入力位置が前記所定領域内であることを少なくとも条件として、前記移動操作入力開始されたときの入力位置に基づいて第 2 基準位置を設定する第 2 基準位置設定手段と、

前記第 2 基準位置と、前記移動操作入力開始されてから継続して行われる入力の入力位置とに基づいて、前記仮想ゲーム空間において前記オブジェクトを前記第 1 移動処理とは異なる第 2 移動処理で移動させる第 2 移動処理手段とを実行させ、

前記第 1 移動処理においては、前記第 1 基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記第 2 移動処理においては、前記第 2 基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記オブジェクトの移動速度が同じである場合、前記第 2 移動処理における前記オブジェクトの前記移動方向の変化量は、前記第 1 移動処理における前記オブジェクトの前記移

10

20

30

40

50

動方向の変化量よりも大きい、情報処理装置。

【請求項 1 2】

表示画面の位置に対応する位置であって、ポインティングデバイスによって検出される入力位置を繰り返し取得する取得手段と、

前記ポインティングデバイスに対する移動操作入力が始まったときの入力位置が所定領域外であることを少なくとも条件として、前記移動操作入力が始まったときの入力位置に基づいて第 1 基準位置を設定する第 1 基準位置設定手段と、

前記第 1 基準位置と、前記移動操作入力が始まってから継続して行われる入力の入力位置とに基づいて、仮想ゲーム空間においてオブジェクトを第 1 移動処理で移動させる第 1 移動処理手段と、

前記ポインティングデバイスに対する移動操作入力が始まったときの入力位置が前記所定領域内であることを少なくとも条件として、前記移動操作入力が始まったときの入力位置に基づいて第 2 基準位置を設定する第 2 基準位置設定手段と、

前記第 2 基準位置と、前記移動操作入力が始まってから継続して行われる入力の入力位置とに基づいて、前記仮想ゲーム空間において前記オブジェクトを前記第 1 移動処理とは異なる第 2 移動処理で移動させる第 2 移動処理を実行する第 2 移動処理手段とを備え、

前記第 1 移動処理においては、前記第 1 基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記第 2 移動処理においては、前記第 2 基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記オブジェクトの移動方向の変化量が同じである場合、前記第 2 移動処理における前記オブジェクトの移動速度は、前記第 1 移動処理における前記オブジェクトの移動速度よりも大きい、情報処理システム。

【請求項 1 3】

表示画面上の位置に対応する位置であって、ポインティングデバイスによって検出される入力位置を繰り返し取得する取得手段と、

前記ポインティングデバイスに対する移動操作入力が始まったときの入力位置が所定領域外であることを少なくとも条件として、前記移動操作入力が始まったときの入力位置に基づいて第 1 基準位置を設定する第 1 基準位置設定手段と、

前記第 1 基準位置と、前記移動操作入力が始まってから継続して行われる入力の入力位置とに基づいて、仮想ゲーム空間においてオブジェクトを第 1 移動処理で移動させる第 1 移動処理手段と、

前記ポインティングデバイスに対する移動操作入力が始まったときの入力位置が前記所定領域内であることを少なくとも条件として、前記移動操作入力が始まったときの入力位置に基づいて第 2 基準位置を設定する第 2 基準位置設定手段と、

前記第 2 基準位置と、前記移動操作入力が始まってから継続して行われる入力の入力位置とに基づいて、前記仮想ゲーム空間において前記オブジェクトを前記第 1 移動処理とは異なる第 2 移動処理で移動させる第 2 移動処理手段とを実行させ、

前記第 1 移動処理においては、前記第 1 基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記第 2 移動処理においては、前記第 2 基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記オブジェクトの移動速度が同じである場合、前記第 2 移動処理における前記オブジェクトの前記移動方向の変化量は、前記第 1 移動処理における前記オブジェクトの前記移動方向の変化量よりも大きい、情報処理システム。

【請求項 1 4】

情報処理システムにおいて実行されるゲーム処理方法であって、

表示画面の位置に対応する位置であって、ポインティングデバイスによって検出される入力位置を繰り返し取得する取得ステップと、

前記ポインティングデバイスに対する移動操作入力が始まったときの入力位置が前記

10

20

30

40

50

表示画面における所定領域外であることを少なくとも条件として、前記移動操作入力開始されたときの入力位置に基づいて第1基準位置を設定する第1基準位置設定ステップと、
前記第1基準位置と、前記移動操作入力開始されてから継続して行われる入力の入力位置とに基づいて、仮想ゲーム空間においてオブジェクトを第1移動処理で移動させる第1移動処理ステップと、

前記ポインティングデバイスに対する移動操作入力開始されたときの入力位置が前記所定領域内であることを少なくとも条件として、前記移動操作入力開始されたときの入力位置に基づいて第2基準位置を設定する第2基準位置設定ステップと、

前記第2基準位置と、前記移動操作入力開始されてから継続して行われる入力の入力位置とに基づいて、前記仮想ゲーム空間において前記オブジェクトを前記第1移動処理とは異なる第2移動処理で移動させる第2移動処理を実行する第2移動処理ステップとを含み、

前記第1移動処理においては、前記第1基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記第2移動処理においては、前記第2基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記オブジェクトの移動方向の変化量が同じである場合、前記第2移動処理における前記オブジェクトの移動速度は、前記第1移動処理における前記オブジェクトの移動速度よりも大きい、ゲーム処理方法。

【請求項15】

情報処理システムにおいて実行されるゲーム処理方法であって、

表示画面上の位置に対応する位置であって、ポインティングデバイスによって検出される入力位置を繰り返し取得する取得ステップと、

前記ポインティングデバイスに対する移動操作入力開始されたときの入力位置が所定領域外であることを少なくとも条件として、前記移動操作入力開始されたときの入力位置に基づいて第1基準位置を設定する第1基準位置設定ステップと、

前記第1基準位置と、前記移動操作入力開始されてから継続して行われる入力の入力位置とに基づいて、仮想ゲーム空間においてオブジェクトを第1移動処理で移動させる第1移動処理ステップと、

前記ポインティングデバイスに対する移動操作入力開始されたときの入力位置が前記所定領域内であることを少なくとも条件として、前記移動操作入力開始されたときの入力位置に基づいて第2基準位置を設定する第2基準位置設定ステップと、

前記第2基準位置と、前記移動操作入力開始されてから継続して行われる入力の入力位置とに基づいて、前記仮想ゲーム空間において前記オブジェクトを前記第1移動処理とは異なる第2移動処理で移動させる第2移動処理ステップとを含み、

前記第1移動処理においては、前記第1基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記第2移動処理においては、前記第2基準位置と前記入力位置との間の距離に基づいて、前記仮想ゲーム空間において前記オブジェクトの移動方向が変化され、

前記オブジェクトの移動速度が同じである場合、前記第2移動処理における前記オブジェクトの前記移動方向の変化量は、前記第1移動処理における前記オブジェクトの前記移動方向の変化量よりも大きい、ゲーム処理方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ポインティングデバイスを用いて入力を行うゲームプログラム、ゲーム装置、ゲームシステム、および、ゲーム処理方法に関する。

【背景技術】

【0002】

従来、タッチパネル等のポインティングデバイスを用いてゲーム操作を行う技術がある。

10

20

30

40

50

例えば、ゲーム空間のオブジェクトを移動させるゲーム操作が、ポインティングデバイスを用いて行われる（例えば、特許文献1参照）。

【先行技術文献】

【特許文献】

【0003】

【文献】特開2013-127683号公報

【発明の概要】

【発明が解決しようとする課題】

【0004】

ポインティングデバイスを用いて、より多様なゲーム操作を可能とするユーザインターフェースが望まれている。

10

【0005】

それ故、本発明の目的は、ポインティングデバイスを用いて多様なゲーム操作を可能とするゲームプログラム、ゲーム装置、ゲームシステム、および、ゲーム処理方法を提供することである。

【課題を解決するための手段】

【0006】

上記の課題を解決すべく、本発明は、以下の(1)~(11)の構成を採用した。

【0007】

(1)

20

本発明の一例は、情報処理装置のコンピュータに、取得ステップと、第1処理実行ステップと、基準位置設定ステップと、移動処理ステップとを実行させる、ゲームプログラムである。取得ステップにおいて、コンピュータは、表示画面上の位置に対応する位置であって、ポインティングデバイスによって検出される入力位置を繰り返し取得する。第1処理実行ステップにおいて、コンピュータは、ポインティングデバイスに対する入力開始されたときの入力位置が所定領域内であることを少なくとも条件として、仮想ゲーム空間においてオブジェクトに対する所定の第1処理を実行する。基準位置設定ステップにおいて、コンピュータは、ポインティングデバイスに対する入力に関する入力位置に基づいて基準位置を設定する。移動処理ステップにおいて、コンピュータは、基準位置と、ポインティングデバイスに対する入力開始されてから継続して行われる入力の入力位置とに基づいて、オブジェクトを移動させる移動処理を実行する。

30

【0008】

上記(1)の構成によれば、ユーザは、所定領域内から開始される一連の入力によって、オブジェクトに対する第1処理と移動処理という2種類の処理を実行することができる。すなわち、情報処理装置は、一連の入力によるシンプルな入力方法によって多様な操作をユーザに行わせることができる。

【0009】

(2)

所定領域は、ポインティングデバイスの入力面において、表示画面においてオブジェクトが表示される位置に対応する位置を含むように設定されてもよい。

40

【0010】

上記(2)の構成によれば、ユーザは、オブジェクトの位置に対する入力を行うか否かによって、情報処理装置に上記第1処理を実行させるか否かを使い分けることができる。したがって、情報処理装置は、上記第1処理を実行させるか否かの使い分けを、直感的な操作によってユーザに行わせることができる。

【0011】

(3)

ゲームプログラムは、画像表示ステップをコンピュータにさらに実行させてもよい。画像表示ステップにおいて、コンピュータは、第1処理を実行可能な所定期間において、オブジェクトが表示画面上の所定位置に固定的に配置されるように仮想ゲーム空間の画像を

50

表示画面に表示する。

【0012】

上記(3)の構成によれば、ユーザは、オブジェクトの状況によらず、表示画面上における同じ位置から入力を開始することによって、上記第1処理を実行させるか否かの使い分けを行うことができる。これによれば、情報処理装置は、上記使い分けの操作性を向上することができる。

【0013】

(4)

ゲームプログラムは、画像表示ステップをコンピュータにさらに実行させてもよい。画像表示ステップにおいて、コンピュータは、第1処理を実行可能な所定期間において、所定領域を表す画像を含まないゲーム空間の画像を表示画面に表示する。

10

【0014】

上記(4)の構成によれば、情報処理装置は、所定領域の画像を表示しないので、ゲーム画像を簡潔にすることができる。これによって、情報処理装置は、ゲーム画像を生成する処理負荷を軽減することができる。また、所定領域はオブジェクトの位置に設定されるので、所定領域が表示されなくても、ユーザは、オブジェクトの画像によって所定領域の位置を概ね認識することができる。したがって、情報処理装置は、ユーザにとって見やすく、かつ、わかりやすいゲーム画像を提示することができる。

【0015】

(5)

ゲームプログラムは、基準位置変更ステップをコンピュータにさらに実行させてもよい。基準位置変更ステップにおいて、コンピュータは、所定領域に対する入力開始から継続して行われる入力の入力位置から所定距離内となるように基準位置を変更する。

20

【0016】

上記(5)の構成によれば、ユーザは、ポインティングデバイスに対する入力開始から継続して行われる入力において、表示画面上においてオブジェクトから離れた位置での入力によっても移動操作を行うことができる。これによれば、例えばポインティングデバイスに対する入力をユーザが指で行う場合には、ユーザは、入力を行う指によってオブジェクトが見にくくならないようにすることができる。したがって、情報処理装置は、ポインティングデバイスに対する入力の操作性を向上することができる。

30

【0017】

(6)

第1処理実行ステップにおいては、第1処理として、オブジェクトの状態を変化させる処理が実行されてもよい。

【0018】

上記(6)の構成によれば、ユーザは、オブジェクトに対する移動操作と状態を変化させる操作との両方の操作を、ポインティングデバイスに対する一連の入力によって行うことができる。これによれば、ポインティングデバイスに対する一連の入力によって複数種類の操作をユーザに行わせることができるので、より多様なゲーム操作を可能とすることができる。

40

【0019】

(7)

第1処理実行ステップにおいては、第1処理として、オブジェクトの移動に関する状態を変化させる処理が実行されてもよい。移動処理ステップにおいては、コンピュータは、オブジェクトの状態に応じて異なる制御方法でオブジェクトを移動させてもよい。

【0020】

上記(7)の構成によれば、ユーザは、第1処理によって、移動操作による移動の制御方法を変更することができる。したがって、上記(6)の構成によれば、ユーザは、オブジェクトに対する移動操作と、移動操作による移動制御方法の変更操作とを、一連の入力によって行うことができる。これによれば、多様な移動操作を簡易な入力方法によってユ

50

ーザに行わせることができる。

【0021】

(8)

ゲームプログラムは、第2処理実行ステップを、コンピュータにさらに実行させてもよい。第2処理実行ステップにおいて、コンピュータは、ポインティングデバイスに対する入力開始されたときの入力位置が所定領域内である場合、当該入力開始されてから継続して行われる入力が所定の条件を満たす場合、第1処理とは異なる処理であって、オブジェクトに対する第2処理を実行する。

【0022】

上記(8)の構成によれば、ユーザは、情報処理装置に第1処理を実行させた後に、さらに、所定の条件を満たす操作によって情報処理装置に第2処理を実行させることができる。これによれば、ポインティングデバイスに対する一連の入力によって複数種類の操作をユーザに行わせることができるので、より多様なゲーム操作を可能とすることができる。

10

【0023】

(9)

第2処理実行ステップにおいては、第2処理として、オブジェクトの移動に関する処理が実行されてもよい。

【0024】

上記(9)の構成によれば、ユーザは、情報処理装置に第1処理を実行させた後においても、オブジェクトの移動に関する操作を行うことができる。これによれば、オブジェクトの移動に関する多様な操作を簡易な入力方法によってユーザに行わせることができる。

20

【0025】

(10)

第2処理実行ステップにおいては、ポインティングデバイスに対する入力開始されてから継続して行われる入力の終了に応じて第2処理が実行されてもよい。

【0026】

上記(10)の構成によれば、ユーザは、ポインティングデバイスに対する一連の入力を終了する操作によって、情報処理装置に第2処理を実行させることができる。これによれば、一連の入力中の操作に加えて、一連の入力を終了する操作によってもユーザにゲーム操作を行わせることができるので、より多様なゲーム操作を可能とすることができる。

30

【0027】

(11)

第1処理実行ステップにおいては、第1処理として、移動処理とは異なる処理であって、オブジェクトの動作を制御する処理が実行されてもよい。

【0028】

上記(11)の構成によれば、ユーザは、オブジェクトに対する移動操作と、当該オブジェクトに動作を行わせる操作とを、ポインティングデバイスに対する一連の入力によって行うことができる。これによれば、ポインティングデバイスに対する一連の入力によって複数種類の操作をユーザに行わせることができるので、より多様なゲーム操作を可能とすることができる。

40

【0029】

なお、本発明の別の一例は、上記(1)～(11)における各ステップの処理を実行する(プログラムによって各ステップを実行する態様に限らない)情報処理装置または情報処理システムであってもよい。また、本発明の別の一例は、上記(1)～(11)における各ステップの処理をコンピュータが実行するゲーム処理方法であってもよい。

【発明の効果】

【0030】

本発明によれば、ポインティングデバイスを用いて多様なゲーム操作を可能とすることができる。

【図面の簡単な説明】

50

【 0 0 3 1 】

【図 1】本実施形態における情報処理システムの構成の一例を示すブロック図

【図 2】サーバの構成の一例を示すブロック図

【図 3】端末装置の構成の一例を示すブロック図

【図 4】ゲーム画像の一例を示す図

【図 5】本実施形態における判定領域の一例を示す図

【図 6】第 1 の移動操作が行われる場合にタッチ入力開始されたときのゲーム画像の一例を示す図

【図 7】操作画像の一例を示す図

【図 8】基準位置が変更される様子の一例を示す図

10

【図 9】第 2 の移動操作が行われる場合にタッチ入力開始されたときのゲーム画像の一例を示す図

【図 10】加速可能状態におけるオブジェクトを含むゲーム画像の一例を示す図

【図 11】加速状態におけるオブジェクトを含むゲーム画像の一例を示す図

【図 12】情報処理システムにおけるゲーム処理に用いられるデータの一例を示す図

【図 13】端末装置によって実行されるゲーム処理の一例の流れを示すフローチャート

【図 14】端末装置によって実行されるゲーム処理の一例の流れを示すフローチャート

【図 15】第 1 の変形例における判定領域の一例を示す図

【図 16】第 1 の変形例において第 2 の移動操作が行われるときのゲーム画像の一例を示す図

20

【図 17】第 2 の変形例における判定領域の一例を示す図

【図 18】第 2 の変形例において第 2 の移動操作が行われるときのゲーム画像の一例を示す図

【発明を実施するための形態】

【 0 0 3 2 】

[1 . システムの全体構成]

以下、本実施形態に係る情報処理システム、情報処理装置、ゲームプログラム、および、ゲーム処理方法について説明する。まず、本実施形態に係る情報処理システムの全体構成と、情報処理システムに含まれる端末装置およびサーバの構成とについて説明する。図 1 は、本実施形態における情報処理システムの構成の一例を示すブロック図である。図 1 に示すように、情報処理システムは、サーバ 1 と、端末装置 2 とを含む。これらのサーバ 1 および端末装置 2 は、インターネットおよび/またはモバイル通信網等のネットワーク 3 に接続可能である。サーバ 1 および端末装置 2 は、ネットワーク 3 を介して互いに通信可能である。

30

【 0 0 3 3 】

サーバ 1 は、端末装置 2 において実行されるアプリケーションに関するサービスを提供するためのサーバである。本実施形態において、サーバ 1 は、端末装置 2 においてゲームを実行するためのゲームサーバである。つまり、サーバ 1 は、端末装置 2 においてゲーム処理を実行するための環境を提供する。例えば、サーバ 1 は、ゲーム処理を実行する端末装置 2 からの要求に応じて、必要に応じてゲーム処理を実行し、要求に応じたゲームデータを当該端末装置 2 へ送信する。

40

【 0 0 3 4 】

端末装置 2 は、ユーザが有する情報処理装置の一例であり、例えば、スマートフォン、携帯型または据置型のゲーム装置、携帯電話、タブレット端末、あるいは、ウェアラブル端末等である。端末装置 2 は、サーバ 1 がサービスを提供するゲームのためのゲームプログラム（換言すれば、ゲームアプリケーション）を実行することが可能である。

【 0 0 3 5 】

（サーバ 1 の構成の具体例）

図 2 は、サーバ 1 の構成の一例を示すブロック図である。図 2 に示す、サーバ 1 が備える各構成は、1 以上の情報処理装置によって実現される。なお、本実施形態におけるサー

50

サーバのハードウェア構成は、従来のサーバのためのハードウェア構成と同様であってもよい。

【0036】

図2に示すように、サーバ1は、処理部11および記憶部12を備える。処理部11は、サーバ1の各部12～15に電氣的に接続される。処理部11は、CPU(Central Processing Unit)およびメモリを有する。サーバ1においては、CPUがメモリを用いて、記憶部12に記憶されたプログラムを実行することによって各種の情報処理が実行される。記憶部12は、処理部11がアクセス可能な任意の記憶装置(記憶媒体とも言う)である。記憶部12は、処理部11において実行されるプログラム、処理部11による情報処理に用いられるデータ、および、当該情報処理によって得られたデータ等を記憶する。本実施形態においては、記憶部12は、端末装置2において実行されるゲーム処理のためにサーバ側で実行されるゲーム処理のためのプログラムを少なくとも記憶する。

10

【0037】

サーバ1は、通信部13を備える。通信部13は、ネットワーク3に接続し、ネットワーク3を介して他の装置(例えば、端末装置2)と通信を行う機能を有する。また、サーバ1は、入出力インターフェースとして、入力部14および表示部15を備える。

【0038】

(端末装置2の構成の具体例)

図3は、端末装置2の構成の一例を示すブロック図である。図3に示すように、端末装置2は、処理部21および記憶部22を備える。処理部21は、端末装置2の各部22～25に電氣的に接続される。処理部21は、CPUおよびメモリを有する。端末装置2においては、CPUがメモリを用いて、記憶部22に記憶されたプログラム(上述のゲームプログラムを含む)を実行することによって各種の情報処理が実行される。記憶部22は、処理部21において実行されるプログラム、処理部21による情報処理に用いられるデータ、および、当該情報処理によって得られたデータ等を記憶する。なお、記憶部22は、端末装置2に対して着脱可能な記憶媒体(例えば、カード型の記憶媒体)であってもよい。

20

【0039】

端末装置2は、入力部の一例として、ポインティングデバイスを備える。ポインティングデバイスは、入力面上において(例えばユーザによって)指定された位置を検出可能な任意の入力装置である。本実施形態においては、ポインティングデバイスは、表示部24の画面上に設けられるタッチパネル23である。タッチパネル23は、入力面(換言すれば、表示部24の画面上に設定される面)にタッチする入力(「タッチ入力」と呼ぶ)によって指定される位置を検出する。

30

【0040】

ポインティングデバイスは、例えばタッチスクリーン、タッチパネル、または、タッチパッド等、入力面上において(例えばユーザによって)指定された位置を検出することが可能なタッチ入力式の入力装置であってもよい。タッチ入力式の入力装置は、入力面に対する接触または非接触の入力を行うことが可能な入力装置であってもよい。すなわち、タッチ入力式の入力装置は、入力面に対するタッチ入力の位置を検出することが可能な装置に限らず、入力面から少し離れた位置に配置される指等を検知することによって入力位置を検出することが可能な入力装置(例えば、フローティングタッチ機能を有する入力装置)であってもよい。つまり、タッチ入力式の入力装置は、入力面に対するタッチ入力に代えて(またはタッチ入力に加えて)、入力面には接触せずに、入力面から少し離れた位置に指等を配置する入力によって指定される位置を検出してよい。

40

【0041】

さらに、ポインティングデバイスは、タッチ入力式の入力装置に限らず、マウスであってもよいし、または、ユーザによって動かされる操作装置であって、当該操作装置の位置および/または向きによって入力面上の位置を指定することが可能な操作装置であっても

50

よい。

【 0 0 4 2 】

端末装置 2 は、表示部 2 4 を備える。表示部 2 4 は、端末装置 2 の処理部 2 1 において実行される情報処理によって生成された画像（例えば、ゲーム画像等）を表示する。

【 0 0 4 3 】

端末装置 2 は、通信部 2 5 を備える。本実施形態において、通信部 2 5 は、モバイル通信網（換言すれば、携帯電話通信網）に接続して通信を行う機能を有する。すなわち、端末装置 2（具体的には、処理部 2 1）は、通信部 2 5 を用いて（換言すれば、通信部 2 5 を介して）モバイル通信網を介してネットワーク 3 に接続し、他の装置（例えば、サーバ 1 等）と通信を行う。なお、端末装置 2 がネットワーク 3 を介した通信を行うための通信部の構成は任意である。例えば、通信部 2 5 は、Wi-Fi の認証を受けた通信モジュールによって無線 LAN に接続する機能を有していてもよいし、モバイル通信網に接続する機能と、無線 LAN に接続する機能との両方を有していてもよい。

10

【 0 0 4 4 】

[2 . 情報処理システムにおける処理の概要]

以下、図 4 ~ 図 1 1 を参照して、情報処理システムにおいて実行されるゲーム処理の概要について説明する。なお、本実施形態においては、情報処理システムにおいて、車を模したオブジェクトをユーザ（換言すれば、プレイヤー）が操作するレースゲーム（図 4 参照）のためのゲーム処理が実行される場合を例として説明する。

【 0 0 4 5 】

本実施形態においては、ユーザは、タッチパネル 2 3 に対するタッチ入力によってオブジェクトに対する操作を行う。すなわち、端末装置 2 は、タッチパネル 2 3 に対するタッチ入力に基づいてオブジェクトの動作（より具体的には、移動）を制御する。

20

【 0 0 4 6 】

[2 - 1 . タッチ入力が行われていない場合における動作]

まず、タッチ入力が行われていない場合における動作について説明する。図 4 は、ゲーム画像の一例を示す図である。なお、図 4 は、タッチ入力が行われていない場合におけるゲーム画像である。図 4 に示されるように、端末装置 2 は、仮想のゲーム空間（本実施形態においては、レースコース）と、当該ゲーム空間に登場するオブジェクト 3 1 とを含むゲーム画像を表示部 2 4 に表示する。オブジェクト 3 1 は、車（および、車を運転するキャラクター）を模したオブジェクトであり、ユーザによって操作される。

30

【 0 0 4 7 】

本実施形態では、タッチ入力が行われていない場合、端末装置 2 は、オブジェクト 3 1 の移動を自動的に（すなわち、予め定められたアルゴリズムに従って）制御する。この場合、オブジェクト 3 1 は、レースコースに沿って進むように自動的に制御される。なお、他の実施形態においては、端末装置 2 は、タッチ入力が行われていない場合、直進するようにオブジェクト 3 1 の移動を制御してもよい。

【 0 0 4 8 】

また、本実施形態においては、オブジェクト 3 1 のスピードは、後述する加速動作が行われる場合を除いて、自動的に（例えば、一定速度で）制御される。なお、他の実施形態においては、オブジェクト 3 1 のスピードは、ユーザの入力に基づいて制御されてもよい。例えば、端末装置 2 は、ユーザによる指示（例えば、画面上に表示されるアクセルボタンに対するタッチ入力による指示）に基づいてオブジェクト 3 1 のスピードを制御してもよい。

40

【 0 0 4 9 】

ここで、本実施形態においては、オブジェクト 3 1 は、通常状態とドリフト状態という、2 種類の状態をとることが可能である。詳細は後述するが、通常状態とドリフト状態とは、ユーザによるタッチ入力に基づいてオブジェクト 3 1 の移動を制御する処理の内容が異なっている。本実施形態においては、タッチ入力が行われていない場合には少なくとも、オブジェクト 3 1 は通常状態となる。

50

【 0 0 5 0 】

[2 - 2 . タッチ入力が行われている場合における動作]

本実施形態では、ユーザは、タッチ入力によってオブジェクト 3 1 に対する移動操作を行う。ここで、本実施形態においては、ユーザは、画面上においてオブジェクト 3 1 とは異なる位置に対してタッチ入力を開始する第 1 の移動操作と、オブジェクト 3 1 の位置に対してタッチ入力を開始する第 2 の移動操作とを行うことが可能である。第 1 の移動操作が行われる場合、端末装置 2 は、上記の通常状態でオブジェクト 3 1 の移動制御を行う。一方、第 2 の移動操作が行われる場合、端末装置 2 は、上記のドリフト状態でオブジェクト 3 1 の移動制御を行う。

【 0 0 5 1 】

上記のように、本実施形態においては、タッチ入力のタッチオン時の入力位置（以下、「タッチオン位置」と呼ぶ）がオブジェクト 3 1 の位置であるか否かによって、行われる移動操作の種類が異なる。ここで、本明細書においては、「タッチオン」とは、タッチ入力が開始されたこと、換言すれば、タッチ入力が行われていない状態からタッチ入力が行われている状態へと変化したことを意味する。また、本明細書においては、タッチ入力が終了したこと、換言すれば、タッチ入力が行われている状態からタッチ入力が行われていない状態へと変化したことを、「タッチオフ」と呼ぶ。

【 0 0 5 2 】

(判定領域)

本実施形態においては、タッチオン位置がオブジェクト 3 1 の位置であるか否かの判定は、タッチパネル 2 3 の入力面上に設定される判定領域を用いて行われる。図 5 は、本実施形態における判定領域の一例を示す図である。端末装置 2 は、タッチオン位置が判定領域 3 0 内であるか否かによって、第 2 の移動操作が行われたか、それとも、第 1 の移動操作が行われたかを判定する。すなわち、タッチオン位置が判定領域 3 0 の内側である場合、第 2 の移動操作が行われたと判定され、タッチオン位置が判定領域 3 0 の外側である場合、第 1 の移動操作が行われたと判定される。

【 0 0 5 3 】

なお、例えばタッチパネルのように、ポインティングデバイスの入力面が表示装置 2 4 の画面上に設定される場合、ポインティングデバイスによって検出される入力位置は、正確に言えば、表示画面上の位置に対応する位置である。ただし、入力位置は実質的には表示画面上の位置を示す。そのため、本実施形態の説明においては、説明をわかりやすくする目的で、入力位置および判定領域を表示画面上の位置として説明することがある。なお、「入力位置（または判定領域）が表示画面上の特定の位置にある」とは、厳密には、入力位置（または判定領域）が、表示画面上における特定の位置に対応する位置であって、入力面上における位置にある、という意味である。

【 0 0 5 4 】

図 5 に示すように、本実施形態においては、判定領域 3 0 は、オブジェクト 3 1 が表示される位置を含むように設定される（厳密には、入力面において、表示画面においてオブジェクト 3 1 が表示される位置に対応する位置を含むように設定される）。本実施形態における判定領域 3 0 は円形の形状であるが、他の実施形態においては、判定領域 3 0 は、例えば正方形または長方形等、任意の形状であってよい。また、判定領域 3 0 は、オブジェクト 3 1 を概ね含むように設定されてもよく、必ずしもオブジェクト 3 1 の全体を含むように設定されなくてもよい。

【 0 0 5 5 】

上記より、本実施形態においては、ユーザは、ユーザの操作対象であるオブジェクト 3 1 の位置からタッチ入力を開始するか、それとも、オブジェクト 3 1 とは異なる位置からタッチ入力を開始するかによって、2 種類の移動操作を使い分けることができる。これによれば、（本実施形態のように、判定領域 3 0 が表示されないとしても）ユーザは、直感的でわかりやすい方法で移動操作の使い分けを行うことができる。

【 0 0 5 6 】

10

20

30

40

50

なお、他の実施形態においては、判定領域 30 は、オブジェクト 31 の位置に基づく位置に限らず、他の位置に設定されてもよい（後述する変形例を参照）。

【0057】

本実施形態においては、端末装置 2 は、移動操作が可能な期間（例えば、レースゲーム中）においては、オブジェクト 31 が表示画面上の所定位置（本実施形態においては、画面の中央よりやや下側の位置）に固定的に配置されるように、仮想ゲーム空間の画像を表示部 24 に表示する（図 5 等参照）。これによって、判定領域 30 は、上記所定位置を含むように固定的に設定される。したがって、本実施形態によれば、ユーザは、第 2 の移動操作を行う場合と同じ位置からタッチ入力を開始すればよく、端末装置 2 は、第 2 の移動操作の操作性を向上することができる。

10

【0058】

なお、他の実施形態においては、オブジェクト 31 の表示画面上における位置は、適宜変化してもよい。このとき、端末装置 2 は、オブジェクト 31 の位置に基づいて判定領域 30 を設定してもよい。つまり、判定領域 30 は、画面上で移動するオブジェクト 31 の位置に応じて移動するように（具体的には、オブジェクト 31 の位置を含むように）設定されてもよい。

【0059】

また、図 5 においては、判定領域 30 を点線で囲まれる斜線領域として示しているが、本実施形態において、判定領域 30 を示す画像は、実際には表示されない。つまり、端末装置 2 は、移動操作が可能な期間において、判定領域 30 を表す画像を含まないゲーム空間の画像を表示部 24 に表示する。これによって、端末装置 2 は、ゲーム画像を簡潔にすることができ、オブジェクト 31 等を見やすくユーザに提示することができる。なお、本実施形態においては、判定領域 30 はオブジェクト 31 の位置に設定されるので、判定領域 30 が表示されなくても、ユーザは、オブジェクト 31 の画像によって判定領域 30 の位置を概ね認識することができる。したがって、本実施形態においては、判定領域 30 の位置をわかりやすくユーザに提示できるとともに、見やすいゲーム画像をユーザに提供することができる。

20

【0060】

なお、他の実施形態においては、端末装置 2 は、判定領域 30 を示す画像を表示部 24 に表示してもよい（後述する変形例における図 17 参照）。

30

【0061】

以上のように、本実施形態においては、ユーザは、移動操作の開始位置をオブジェクト 31 の位置（すなわち、判定領域 30 の内側）とするか、それとも、オブジェクト 31 の位置とは異なる位置（すなわち、判定領域 30 の外側）とするかによって、2 種類の移動操作を使い分けることができる。以下、移動操作の詳細について説明する。

【0062】

[2 - 2 - 1 . 第 1 の移動操作]

図 6 ~ 図 8 を参照して、第 1 の移動操作が行われる場合における動作について説明する。図 6 は、第 1 の移動操作が行われる場合にタッチ入力開始されたときのゲーム画像の一例を示す図である。図 6 に示すように、上述の判定領域 30 の外側の位置に対してタッチ入力開始されると、端末装置 2 は、第 1 の移動操作が行われたと判定する。

40

【0063】

本実施形態においては、移動操作（すなわち、第 1 または第 2 の移動操作）のタッチ入力開始されると、端末装置 2 は、まず、タッチオン時の入力位置、すなわちタッチ入力開始時における入力位置に基づいて基準位置を設定する。詳細は後述するが、基準位置は、タッチ入力による入力指示の内容を特定するために用いられる位置である。本実施形態においては、タッチオン時における入力位置が基準位置として設定される。

【0064】

なお、他の実施形態においては、タッチオン時に設定される基準位置は、タッチオン時の入力位置に基づいて設定される任意の位置であってもよい。例えば、タッチオン時の入

50

力位置が表示部 2 4 の画面の端部付近に位置する場合、端末装置 2 は、当該入力位置に基づいて、画面の端部から所定距離となる位置を基準位置として設定してもよい。これは、上記の場合において、基準位置から任意の方向へ所定距離だけ離れた入力位置をユーザが指定することができるようにするためである。また、上記の場合、基準位置は、画面の端部から、後述する基準位置画像の半径 r_2 の距離となる位置に設定されてもよい。また、他の実施形態においては、タッチオン時に設定される基準位置は、タッチオン時の入力位置にかかわらず、固定的に設定されてもよい。すなわち、端末装置 2 は、予め定められた位置（例えば、表示部 2 4 の画面の中心位置）を基準位置として設定してもよい。

【0065】

また、他の実施形態においては、基準位置は、当該基準位置が設定されるときの入力位置に依存しない位置に設定されてもよい。例えば、移動操作の他にタップ操作（例えば、タッチオンからタッチオフまでの時間が所定時間よりも短いタッチ入力操作）を受け付ける場合、端末装置 2 は、開始されたタッチ入力による操作が、移動操作であるのか、それとも、タップ操作であるのかを判別する目的で、タッチオンからの時間が所定時間を経過したか否かを判定してもよい。この場合、端末装置 2 は、移動操作におけるタッチオン時から上記所定時間が経過したときの入力位置を基準位置として設定してもよい。

【0066】

なお、本実施形態においては、ユーザは、タッチパネル 2 3 の入力面上における任意の位置を入力開始位置として、移動操作のためのタッチ入力を開始することができる。すなわち、端末装置 2 は、入力面のうちの任意の位置でタッチ入力開始されたことに応じて、基準位置を設定する処理と、後述する入力指示を特定する処理とを実行する。ただし、他の実施形態においては、端末装置 2 は、入力面のうちの所定の入力受付領域内においてタッチ入力開始されたことを条件として、これらの処理を実行するようにしてもよい。上記入力受付領域は、例えば、オブジェクト 3 1 の位置を基準として設定される領域（ただし、判定領域 3 0 とは異なる領域）であってもよいし、予め定められた領域であってもよい。

【0067】

図 6 に示すように、タッチパネル 2 3 に対するタッチ入力が行われている場合、端末装置 2 は、ゲーム画像に加えて操作画像 3 2 を表示部 2 4 に表示する。なお、図 6 では、タッチ入力を行っているユーザの指を点線で示している。本実施形態においては、操作画像 3 2 は、入力位置画像 3 3 と、基準位置画像 3 4 とを含む。

【0068】

入力位置画像 3 3 は、タッチパネル 2 3 の入力面上の位置であって、現在の入力位置を示す画像である。具体的には、入力位置画像 3 3 は、入力位置を中心とする半径 r_1 の円形領域を表す（図 7 参照）。基準位置画像 3 4 は、タッチパネル 2 3 の入力面上の位置であって、基準位置を示す画像である。具体的には、基準位置画像 3 4 は、基準位置を中心とする半径 r_2 ($> r_1$) の円形領域を表す（図 7 参照）。

【0069】

以上のように、本実施形態においては、操作画像 3 2 が表示されることによって、ユーザは入力位置および基準位置を視認することができる。これによって、ユーザは、意図に沿った入力を行いやすくなり、端末装置 2 は、タッチ入力の操作性を向上することができる。なお、図 6 に示すように、本実施形態における操作画像 3 2 は、方向入力装置の一例であるスライドパッドを模した画像である。ただし、他の実施形態においては、操作画像 3 2 は、基準位置と入力位置とを表す任意の画像であってもよい。また、他の実施形態においては、操作画像 3 2 は表示部 2 4 に表示されなくてもよい。

【0070】

タッチ入力を開始した後、ユーザは、タッチ入力を継続することによって、具体的には、スライド操作によって、オブジェクト 3 1 に対する移動操作を行う。つまり、本実施形態において、ユーザは、タッチ入力によって、オブジェクト 3 1 の移動方向を指示するための入力指示を行う。以下、図 7 を参照して、入力指示を行う方法について説明する。

【 0 0 7 1 】

図 7 は、操作画像の一例を示す図である。なお、図 7 では、表示部 2 4 に表示される操作画像 3 2 に加えて、基準位置と入力位置とを示している。なお、本実施形態において、入力位置は、入力面上の位置を示す 2 次元座標としてタッチパネル 2 3 から取得される。また、入力位置および基準位置は、2 次元座標として端末装置 2 に記憶される。

【 0 0 7 2 】

図 7 は、タッチオン後において、入力位置 P_t がタッチオン時の位置（すなわち、基準位置 P_r ）から移動した状態を示している。タッチ入力による入力指示は、タッチオンからタッチ入力が続く間、行われる。ここで、タッチオンから継続して行われる（すなわち、タッチオンから途切れずに入力される）タッチ入力を、「継続タッチ入力」と呼ぶ。端末装置 2 は、継続タッチ入力が行われている間において、基準位置と入力位置とに基づいて入力指示の内容を特定する。本実施形態においては、入力指示の内容を表す値として、入力方向と、入力距離とが算出される。

10

【 0 0 7 3 】

入力方向は、基準位置 P_r から入力位置 P_t へのベクトル V に基づいて算出される。ベクトル V は、基準位置 P_r を始点とし、入力位置 P_t を終点とするベクトルである。本実施形態においては、入力方向は、上記ベクトル V の x 軸成分（換言すれば、入力面における横方向の成分）の向きとして算出される。すなわち、本実施形態においては、入力方向は、右方向（すなわち、 x 軸正方向）または左方向（すなわち、 x 軸負方向）として算出される。

20

【 0 0 7 4 】

入力距離は、上記ベクトル V の大きさに基づいて算出される。本実施形態においては、入力距離は、上記ベクトル V の x 軸成分の大きさ V_x として算出される。

【 0 0 7 5 】

上記のように、本実施形態においては、入力指示は、1 次元の方向（具体的には、 x 軸方向）に関する方向および大きさである。したがって、入力指示は、例えば、「- 2」や「1」のように、入力方向を正負の符号で表し、入力距離を絶対値で表す数値として表現されてもよい。

【 0 0 7 6 】

なお、他の実施形態においては、入力指示は、2 次元に関する方向および大きさであってもよい。すなわち、入力指示は、上記ベクトル V の向きを入力方向とし、上記ベクトル V の大きさを入力距離とするものであってもよい。また、他の実施形態においては、入力指示は、入力方向および入力距離のいずれか一方のみを含むものであってもよい。

30

【 0 0 7 7 】

端末装置 2 は、上記の入力指示に基づいてオブジェクト 3 1 の動作（具体的には、移動）を制御する。換言すれば、オブジェクト 3 1 は、入力位置と基準位置とに基づいて制御される。オブジェクト 3 1 の具体的な制御方法は任意であるが、本実施形態においては、端末装置 2 は、入力方向に応じた方向に、入力距離に応じた曲がり量で曲がるように、オブジェクト 3 1 を制御する。つまり、オブジェクト 3 1 は、基準位置 P_r に対して入力位置 P_t が右側に位置するほど、右に大きく曲がるように動作し、基準位置 P_r に対して入力位置 P_t が左側に位置するほど、左に大きく曲がるように動作する。なお、オブジェクト 3 1 が曲がる方向および曲がる量は、直進方向を基準に算出されてもよいし、オブジェクト 3 1 が自動走行を行う場合（つまり、タッチ入力が行われていない場合）におけるオブジェクト 3 1 の進行方向を基準に算出されてもよい。また、上述のように、本実施形態においては、オブジェクト 3 1 のスピードは、加速動作が行われる場合を除いて、自動的に制御される。

40

【 0 0 7 8 】

上記のように、本実施形態においては、端末装置 2 は、基準位置とタッチオン後に継続して取得される入力位置との間の距離である入力距離と、基準位置から入力位置への方向である入力方向とに基づいてオブジェクト 3 1 の移動を制御する。これによれば、ユーザ

50

は、タッチオンから継続する一連のタッチ入力（すなわち、継続タッチ入力）という簡単な操作方法によって、オブジェクトに対する操作を行うことができる。なお、他の実施形態においては、上記入力距離と入力方向との少なくとも一方に基づいてオブジェクト 3 1 の移動が制御されてもよい。

【 0 0 7 9 】

また、端末装置 2 は、第 1 の移動操作において（なお、第 2 の移動操作についても同様である）、入力距離が所定の制御開始距離を超えたことを条件として、オブジェクト 3 1 の制御を開始するようにしてもよい。このとき、入力開始されてから、入力位置が基準位置からある程度離れるまでの期間においては、オブジェクト 3 1 に対する制御が行われない。これによれば、例えばユーザが移動操作を行う意図がなく誤ってタッチ入力を行ってしまっ

10

【 0 0 8 0 】

（基準位置の変更処理）

本実施形態においては、タッチオン時に設定された基準位置は、タッチオンから継続するタッチ入力が行われている間に変更されることがある。以下、基準位置の変更処理について説明する。

【 0 0 8 1 】

図 8 は、基準位置が変更される様子の一例を示す図である。図 8 において、位置 P_{r0} は、継続タッチ入力中のある時点 $t = 0$ における基準位置であり、位置 P_{t0} は、当該ある時点 $t = 0$ に検出された入力位置である。図 8 に示す例においては、上記時点 $t = 0$ の次に入力位置が検出される時点 $t = 1$ において、入力位置 P_{t1} が検出される。本実施形態においては、位置 P_{t1} の入力位置が検出されたことに応じて、基準位置は、位置 P_{r0} から位置 P_{r1} に変更される（図 8 参照）。

20

【 0 0 8 2 】

具体的には、端末装置 2 は、継続タッチ入力によって入力位置が移動した結果、基準位置から入力位置までの距離が所定距離よりも大きくなる場合、端末装置 2 は基準位置を変更する。ここで、所定距離は、予め定められており、例えば、上記基準位置画像 3 4 の半径 r_2 から上記入力位置画像の半径 r_1 を減算した値である。したがって、本実施形態においては、基準位置は、入力位置画像 3 3 が基準位置画像 3 4 の内側に含まれるように設定（換言すれば、変更）される。つまり、入力位置画像 3 3 は、基準位置画像 3 4 の円形領域内に含まれるように配置される。なお、他の実施形態においては、基準位置は、入力位置（すなわち、入力位置画像 3 3 の中心）が基準位置画像 3 4 の内側に含まれるように設定されてもよい。このとき、上記所定距離は、基準位置画像 3 4 の半径 r_2 に設定される。また、他の実施形態においては、上記所定距離は、入力位置画像 3 3 の半径 r_1 および基準位置画像 3 4 の半径 r_2 とは独立した値に設定されてもよい。つまり、入力位置画像 3 3 および基準位置画像 3 4 の大きさは、基準位置の変更処理に用いられなくてもよい。

30

【 0 0 8 3 】

上記のように、端末装置 2 は、基準位置から入力位置までの距離が上記所定距離以内となるように、基準位置を変更する。本実施形態においては、変更後の基準位置 P_{r1} は、変更前の基準位置 P_{r0} と、現在の入力位置 P_{t1} とを結ぶ線分上の位置であって、変更後の基準位置 P_{r1} から現在の入力位置 P_{t1} までの距離が上記所定距離となる位置に決定される（図 8 参照）。

40

【 0 0 8 4 】

なお、変更後の基準位置の具体的な算出方法は任意である。例えば、他の実施形態においては、端末装置 2 は、基準位置を変更する前の入力位置 P_{t0} から現在の入力位置 P_{t1} への方向に応じた方向へ、変更前の基準位置 P_{r0} を移動させるように、変更後の基準位置 P_{r1} を算出してもよい。

【 0 0 8 5 】

50

上記のように基準位置が変更された場合、端末装置 2 は、オブジェクト 3 1 の移動制御のために用いられる上記入力指示（入力方向および入力距離）を、変更後の基準位置に基づいて算出する。なお、入力指示に基づいてオブジェクト 3 1 を制御する処理は、基準位置の変更前と変更後とにおいて同様である。

【 0 0 8 6 】

以上のように、本実施形態においては、端末装置 2 は、一定条件下で入力位置の移動に追従するように基準位置を変更する。これによれば、入力位置と基準位置との距離が上記所定距離以内に維持されるので、ユーザは、現在の入力位置を基準位置に対して反対側へ移動させる操作を容易に行うことができる。例えば、ユーザは、基準位置に対して右側にある入力位置を、基準位置に対して左側の位置へと容易に変更することができる。つまり、ユーザは、右方向を指示する入力指示を、左方向を指示する入力指示へと容易に変更することができる。

10

【 0 0 8 7 】

なお、他の実施形態においては、端末装置 2 は、基準位置を変更する処理を実行しなくてもよい。すなわち、入力位置から基準位置までの距離が上記所定距離より大きくなった場合であっても、基準位置が変更されなくてもよい。なお、このとき、端末装置 2 は、入力距離の大きさに制限をかける（すなわち、入力距離の最大値を設定する）ようにしてもよい。すなわち、端末装置 2 は、入力位置から基準位置までの距離が上記所定距離より大きい場合には、入力距離を所定距離としてもよい。

【 0 0 8 8 】

[2 - 2 - 2 . 第 2 の移動操作]

図 9 ~ 図 1 1 を参照して、第 2 の移動操作が行われる場合における動作について説明する。図 9 は、第 2 の移動操作が行われる場合にタッチ入力が増加されたときのゲーム画像の一例を示す図である。図 9 に示すように、上述の判定領域 3 0 の内側の位置に対してタッチ入力が増加されると、端末装置 2 は、第 2 の移動操作が行われたと判定する。

20

【 0 0 8 9 】

図 9 に示すように、第 2 の移動操作が行われたと判定された場合、第 1 の移動操作と同様に、基準位置が設定される。なお、第 2 の移動操作においても第 1 の移動操作と同様、端末装置 2 は、タッチオン時の入力位置を基準位置として設定する。なお、他の実施形態においては、第 2 の移動操作についても第 1 の移動操作と同様に、基準位置は、タッチオン時の入力位置に基づかずに設定されてもよい。

30

【 0 0 9 0 】

また、第 2 の移動操作においても第 1 の移動操作と同様、基準位置の変更処理が実行される。つまり、端末装置 2 は、第 2 の移動操作が行われる間において、継続タッチ入力の入力位置から所定距離内となるように基準位置を変更する。これによれば、ユーザは、第 2 の移動操作中において入力位置をオブジェクト 3 1 の位置から遠ざけることによって、基準位置を移動させることができる。これによって、ユーザは、表示画面上におけるオブジェクト 3 1 の表示位置から離れた位置でも第 2 の移動操作を行うことができ、タッチ入力を行う指によってオブジェクト 3 1 が見にくくならないようにすることができる（図 1 0 参照）。したがって、本実施形態によれば、オブジェクト 3 1 の位置をタッチオン位置とする第 2 の移動操作の操作性を向上することができる。

40

【 0 0 9 1 】

図 9 に示すように、第 2 の移動操作が行われたと判定された場合、端末装置 2 は、オブジェクト 3 1 の状態をドリフト状態へと遷移させる。したがって、第 2 の移動操作は、ドリフト状態となったオブジェクト 3 1 に対して行われる。

【 0 0 9 2 】

図 9 は、ドリフト状態におけるオブジェクトを含むゲーム画像の一例を示す図である。上述のように、ドリフト状態は、通常状態における移動制御とは異なる方法で移動制御が行われる状態である。図 9 に示すように、ドリフト状態においては、オブジェクト 3 1 がドリフトしている様子（具体的には、タイヤから煙が上がっている様子）が表示される。

50

【 0 0 9 3 】

上記のように、本実施形態においては、ドリフト状態への遷移処理は、判定領域 3 0 内に対するタッチ入力の開始に応じて（換言すれば、タッチオン時に）実行される。ただし、遷移処理が実行されるタイミングは、タッチオン時に限らない。他の実施形態においては、判定領域 3 0 内に対するタッチ入力開始された後、当該タッチ入力に関する条件が満たされたタイミングで遷移処理が実行されてもよい。例えば、上記「タッチ入力に関する条件」は、入力位置と基準位置（換言すれば、タッチオン位置）との距離が、所定のドリフト開始距離以上となったことであってもよい。このとき、入力位置と基準位置との距離は、入力面における 2 次元の距離であってもよいし、入力面における所定の方向（例えば、x 軸成分）の距離であってもよい。また、このとき、端末装置 2 は、上記条件が満たされた場合における入力位置と基準位置との方向に応じて、ドリフト状態の開始時におけるオブジェクトの向きを制御してもよい。すなわち、上記条件が満たされた場合における入力位置が基準位置に対して左側にある場合、オブジェクトは、左向きのドリフト状態（つまり、左に曲がるドリフト状態）に制御され、上記条件が満たされた場合における入力位置が基準位置に対して右側にある場合、オブジェクトは、右向きのドリフト状態に制御されてもよい。

10

【 0 0 9 4 】

また、上記「タッチ入力に関する条件」は、継続タッチ入力における入力位置が基準位置（換言すれば、タッチオン位置）から移動したことであってもよいし、継続タッチ入力における入力位置が判定領域外となったことであってもよい。このように、上記「タッチ入力に関する条件」は、基準位置と入力位置とに基づく条件であってもよい。

20

【 0 0 9 5 】

本実施形態においては、ドリフト状態における第 2 の移動操作は、通常状態における第 1 の移動操作と同様の操作方法である。すなわち、端末装置 2 は、オブジェクト 3 1 がドリフト状態である場合においても、通常状態である場合と同様に、タッチ入力に基づいて入力指示（具体的には、入力方向および入力距離）を特定する。このように、ユーザは、オブジェクト 3 1 が通常状態である場合もドリフト状態である場合も、同じ操作方法でオブジェクト 3 1 を操作することができる。ただし、入力指示に基づいてオブジェクト 3 1 の移動を制御する処理が、通常状態とドリフト状態とでは異なる。

【 0 0 9 6 】

本実施形態においては、ドリフト状態においては、オブジェクト 3 1 は、同じスピードで通常状態よりも鋭くカーブを曲がるように移動制御される。なお、ドリフト状態における移動制御の具体的な方法は任意である。例えば、他の実施形態においては、ドリフト状態においては、オブジェクト 3 1 は、カーブを通常状態よりも速く曲がるように移動制御されてもよい。このように、端末装置 2 は、ドリフト状態では、ユーザが通常状態に比べてゲーム上有利にオブジェクトの移動を操作することができるように、オブジェクト 3 1 の移動制御を行ってもよい。

30

【 0 0 9 7 】

以上のように、本実施形態においては、端末装置 2 は、タッチ入力のタッチオン位置が判定領域内となることを少なくとも条件として実行される処理として、オブジェクト 3 1 の移動に関する状態を変化させる処理（すなわち、通常状態からドリフト状態へと遷移させる処理）が実行される。そして、端末装置 2 は、オブジェクト 3 1 の状態に応じて異なる制御方法でオブジェクトを移動させる。すなわち、端末装置 2 は、通常状態とドリフト状態とで異なる制御方法でオブジェクト 3 1 を移動させる。したがって、本実施形態においては、ユーザは、タッチ入力を開始する位置を変化させることによって、異なる 2 種類の移動操作を使い分けすることができる。これによって、端末装置 2 は、簡単の方法で 2 種類の移動操作の使い分けをユーザに行わせることができ、多様な操作をユーザに行わせることができる。

40

【 0 0 9 8 】

なお、第 2 の移動操作の継続タッチ入力終了すると、端末装置 2 は、オブジェクト 3

50

1をドリフト状態から通常状態へと遷移させる。上述のように、タッチ入力が行われていない期間においては、端末装置2はオブジェクト31を自動的に制御する。

【0099】

ここで、本実施形態においては、第2の移動操作が、所定の加速可能時間の間、継続して行われた場合、オブジェクト31は、(ドリフト状態であって、かつ、)加速可能状態に設定される。すなわち、本実施形態においては、オブジェクト31がドリフト状態である状況において所定の加速条件が満たされると、端末装置2は、オブジェクト31を加速可能状態に設定する。詳細は後述するが、オブジェクト31が加速可能状態となった後で移動操作が終了された(すなわち、タッチオフとなった)場合、オブジェクト31は、通常よりも加速した加速状態で移動が制御される。

10

【0100】

図10は、加速可能状態におけるオブジェクトを含むゲーム画像の一例を示す図である。図10に示すように、加速可能状態においては、オブジェクト31は、ドリフト状態を表す表示態様(具体的には、タイヤから煙が上がっている表示態様)に加えて、加速可能状態を表す表示態様(具体的には、タイヤから火花が出ている表示態様)で表示される。なお、本実施形態においては、ドリフト状態であるオブジェクト31のスピードは、加速可能状態であるか否かによって変わらない。ただし、他の実施形態においては、ドリフト状態であるオブジェクト31が加速可能状態である場合、加速可能状態でない場合に比べてオブジェクト31のスピードが速くてもよい。

【0101】

本実施形態においては、上記加速条件は、ドリフト状態が所定の加速可能時間の間継続されることである。つまり、ドリフト状態となってから(換言すれば、第2の移動操作が開始されてから)加速可能時間が経過したことに応じて、端末装置2は、オブジェクト31を加速可能状態に設定する。

20

【0102】

なお、他の実施形態においては、加速条件の内容は任意である。例えば、加速条件は、「ドリフト状態において継続タッチ入力があるジェスチャを表すこと」という条件を含んでもよいし、「ドリフト状態において継続タッチ入力によって所定の操作が行われたこと」という条件を含んでもよい。このように、加速条件は、入力方向および/または入力距離に関する所定の条件を満たす入力が行われた状態となったことであってもよいし、その状態が所定時間継続したことであってもよい。

30

【0103】

以上のように、本実施形態においては、端末装置2は、第2の移動操作の入力が開始されてから継続して行われる入力が所定の条件(すなわち、加速条件)を満たす場合、第1処理(すなわち、オブジェクト31をドリフト状態へ遷移する処理)とは異なる処理であって、オブジェクト31に対する第2処理(すなわち、オブジェクト31を加速可能状態に設定する処理)を実行する。これによれば、ユーザは、第2の移動操作中において、端末装置2に第1処理を実行させることに加えてさらに、第2処理を実行させることができる。すなわち、端末装置2は、一連のタッチ入力によって、より多様なゲーム操作をユーザに行わせることができる。

40

【0104】

なお、上記第2処理は、オブジェクト31の移動に関する任意の処理であってもよい。例えば、第2処理は、オブジェクト31を所定の制御方法で移動させる処理であってもよい。したがって、本実施形態においては、後述する「加速状態でオブジェクト31を移動させる処理」も第2処理の一例であるということが出来る。なお、このとき、上記所定の条件は、「第2の移動操作が終了すること」である。このように、上記所定の条件は、「第2の移動操作における継続タッチ入力がある状態が所定時間継続したこと」であってもよいし、「第1処理の実行後における継続タッチ入力がある状態を満たす状態が所定時間継続した後で、第1処理の実行後における継続タッチ入力があったこと」であってもよい。また、他の実施形態においては、上記所定の条件は、「第1処理

50

の実行後における継続タッチ入力終了したこと」であってもよい。

【0105】

また、本実施形態において、オブジェクト31が加速可能状態となった後のタイミングで、加速操作（すなわち、タッチ入力を終了する操作）が行われた場合、オブジェクト31は加速状態で走行する。すなわち、オブジェクト31が加速可能状態となった状態で第2の移動操作が終了した場合、端末装置2は、オブジェクト31を加速状態で移動させる。なお、加速状態とは、通常状態におけるオブジェクト31のスピードよりも速いスピードでオブジェクト31が移動する状態である。

【0106】

図11は、加速状態におけるオブジェクトを含むゲーム画像の一例を示す図である。図11に示すように、加速状態においては、オブジェクト31が加速状態である様子（具体的には、後方へ火花を噴射している様子）が表示される。

10

【0107】

本実施形態においては、加速状態は、予め定められた所定の加速継続時間の間、継続される。すなわち、端末装置2は、第2の移動操作の終了時から上記加速継続時間が経過するまでの間、オブジェクト31を加速状態で移動させ、その後、通常状態における自動制御を行う。

【0108】

以上のように、ユーザは、第2の移動操作において、ドリフト状態においてオブジェクト31を加速可能状態となるように操作した後で第2の移動操作を終了することによって、オブジェクト31をより速く走行させることができる。例えば、ユーザは、オブジェクト31をドリフト状態に遷移させた状態でレースコースのカーブを曲がり、レースコースが直線になった位置からオブジェクト31を加速状態で走行させることによって、オブジェクト31を速く走行させることができる。

20

【0109】

なお、他の実施形態においては、オブジェクト31は、複数段階の加速可能状態をとることが可能であってもよい。例えば、ドリフト状態が第1の加速可能時間T1の間継続されることに応じて、オブジェクト31は第1加速可能状態となり、ドリフト状態が第2の加速可能時間T2（なお、 $T2 > T1$ とする）の間継続されることに応じて、オブジェクト31は第2加速可能状態となるように制御されてもよい。このとき、端末装置2は、オブジェクト31が第2加速可能状態から（第2の移動操作の終了に応じて）加速状態になる場合には、第1加速可能状態から（第2の移動操作の終了に応じて）加速状態になる場合に比べて、加速状態が長く継続するようにオブジェクト31を制御する。なお、他の実施形態においては、端末装置2は、上記の場合に、第1加速可能状態から加速状態になる場合に比べて、加速状態におけるスピードが速くなるようにオブジェクト31を制御してもよい。

30

【0110】

また、端末装置2は、複数段階の加速可能状態を互いに区別可能となるようにオブジェクト31を表示してもよい。例えば、端末装置2は、第1加速可能状態と第2加速可能状態とで火花の色および/または大きさが異なるようにオブジェクト31を表示してもよい。

40

【0111】

以上に説明したように、本実施形態においては、ユーザは、第2の移動操作における一連のタッチ入力によって、複数種類の操作（すなわち、ドリフト状態への遷移操作、移動操作、加速状態の設定操作、および、加速操作）を行うことが可能である。このように、本実施形態においては、端末装置2は、ポインティングデバイス（すなわち、タッチパネル）を用いて多様なゲーム操作をユーザに行わせることができる。

【0112】

また、本実施形態においては、各移動操作は、いずれも、ユーザが1本の指で行うことができる操作である。したがって、本実施形態によれば、ユーザは、1本の指でオブジェクト31を操作することができ、例えば、端末装置2を片手で把持するとともに、その手

50

の親指でゲーム操作を行うことができる。

【0113】

なお、上記においては、オブジェクト31に対する移動操作が行われる場合を例として説明したが、上記実施形態においては、タッチ入力によって、当該各操作以外の操作が可能であってもよい。例えば、端末装置2は、アイテムを示すアイコンを表示部24に表示してもよく、オブジェクト31にアイテムを使用させるための操作として、アイコンを指定するタッチ入力を受け付けるようにしてもよい。

【0114】

[3. 情報処理システムにおける処理の具体例]

図12～図14を参照して、上記において概要を説明したゲーム処理が情報処理システムにおいて実行される場合の具体例について説明する。

10

【0115】

[3-1. 処理に用いられるデータ]

図12を参照して、情報処理システムにおけるゲーム処理に用いられるデータの具体例について説明する。図12は、情報処理システムにおけるゲーム処理に用いられるデータの一例を示す図である。図12に示すデータは、端末装置2の記憶装置（具体的には、処理部11のメモリ、および/または、記憶部12）に記憶される。図12に示すように、端末装置2は、入力位置データ、基準位置データ、入力指示データ、オブジェクト位置データ、オブジェクト状態データ、判定領域データ、および、加速フラグデータを記憶する。

【0116】

入力位置データは、上記入力位置を示すデータである。本実施形態では、入力位置データは、現在の入力位置（換言すれば、最新の入力位置）を示すデータを含むことに加えて、過去の入力位置を示すデータを含む。具体的には、入力位置データは、タッチオンから継続して入力される一連のタッチ入力に関する入力位置を示すデータを含む。なお、入力位置データの内容は、タッチオフ時にリセットされてもよい。

20

【0117】

基準位置データは、上記基準位置を示すデータである。なお、上記入力位置データおよび基準位置データは、例えば、タッチパネル23の入力面上における位置を示す2次元座標値のデータである。

【0118】

入力指示データは、入力位置に基づいて特定される上記入力指示の内容を示す。具体的には、入力指示データは、上記入力方向および入力距離を示す。上述したように、入力指示データは、入力方向を正負の符号で表し、入力距離を絶対値で表す数値を示すデータであってもよい。

30

【0119】

オブジェクト位置データは、ゲーム空間におけるオブジェクト31の位置を示す。例えば、オブジェクト位置データは、ゲーム空間における位置を示す3次元または2次元の座標値を示すデータである。

【0120】

オブジェクト状態データは、オブジェクト31の状態を示す。具体的には、オブジェクト状態データは、通常状態またはドリフト状態であることを示すデータである。なお、本実施形態においては、ゲーム開始時（すなわち、レースゲームのスタート時）においては、オブジェクト31は通常状態であるとし、通常状態を示すオブジェクト状態データが記憶される。

40

【0121】

判定領域データは、入力面上に設定される判定領域の位置および範囲を示す。本実施形態においては、判定領域は円形状であるので、判定領域データは、例えば、円の中心位置の座標と、円の半径とを示す。本実施形態においては、判定領域は固定的に設定されるので、判定領域データの内容は、ゲームプログラムにおいて予め定められている。

【0122】

50

加速フラグデータは、オブジェクト 3 1 を加速した状態で移動させる制御を行うか否かを表す加速フラグを示すデータである。詳細は後述するが、オブジェクト 3 1 が上述の加速可能状態または加速状態となる場合、加速フラグがオンに設定され、オブジェクト 3 1 が加速可能状態でもなく加速状態でもない場合、加速フラグがオフに設定される。なお、本実施形態においては、ゲーム開始時においては、加速フラグがオフに設定される。

【 0 1 2 3 】

[3 - 2 . ゲーム処理例]

図 1 3 および図 1 4 は、端末装置によって実行されるゲーム処理の一例の流れを示すフローチャートである。なお、図 1 3 および図 1 4 に示す一連の処理は、記憶部 2 2 に記憶される上記ゲームプログラムの起動後、レースゲームの開始に応じて開始される。

10

【 0 1 2 4 】

なお、本実施形態では、端末装置 2 の処理部 2 1 の CPU (換言すればプロセッサ) が、記憶部 2 2 に記憶されているゲームプログラム (具体的には、レースゲームのためのゲームプログラム) を実行することによって、図 1 3 および図 1 4 に示す各ステップの処理を実行するものとして説明する。ただし、他の実施形態においては、上記各ステップの処理のうちの一部の処理を、CPU とは別のプロセッサ (例えば、専用回路等) が実行するようにしてもよい。また、図 1 3 および図 1 4 に示す各ステップの処理は、単なる一例に過ぎず、同様の結果が得られるのであれば、各ステップの処理順序を入れ替えてもよいし、各ステップの処理に加えて (または代えて) 別の処理が実行されてもよい。

【 0 1 2 5 】

20

また、他の実施形態においては、図 1 3 および図 1 4 に示す各ステップの処理は、端末装置 2 とサーバ 1 との協働によって実行されてもよい。つまり、図 1 3 および図 1 4 に示す各ステップの処理のうちの一部の処理は、サーバ 1 によって実行されてもよい。このとき、サーバ 1 は、処理に用いるデータを、ネットワーク 3 を介して端末装置 2 から受信する。サーバ 1 は、処理の結果得られたデータを、ネットワーク 3 を介して端末装置 2 へ送信する。

【 0 1 2 6 】

また、端末装置の処理部 2 1 は、図 1 3 および図 1 4 に示す各ステップの処理を、メモリを用いて実行する。すなわち、処理部 2 1 の CPU は、各処理ステップによって得られる情報 (換言すれば、データ) をメモリに記憶し、それ以降の処理ステップにおいて当該情報を用いる場合には、メモリから当該情報を読み出して利用する。

30

【 0 1 2 7 】

図 1 3 に示すステップ S 1 において、処理部 2 1 は、タッチパネル 2 3 から入力位置を取得する。すなわち、タッチパネル 2 3 は、入力面上の入力位置を検出し、検出した入力位置を示す検出結果データを出力する。処理部 2 1 は、タッチパネル 2 3 から検出結果データを取得し、取得した検出結果データを含む入力位置データをメモリに記憶する。なお、本実施形態においては、タッチパネル 2 3 は、タッチパネル 2 3 に対するタッチ入力が行われていない場合、タッチ入力がないことを示す検出結果データを出力する。この場合、処理部 2 1 は、タッチ入力がないことを示す検出結果データを取得する。ステップ S 1 の次に、ステップ S 2 の処理が実行される。

40

【 0 1 2 8 】

ステップ S 2 において、処理部 2 1 は、タッチ入力が行われたか否かを判定する。処理部 2 1 は、ステップ S 2 における判定を、メモリに記憶されている入力位置データ (換言すれば、上記ステップ S 1 の処理においてタッチパネル 2 3 から取得されたデータ) に基づいて行う。ステップ S 2 の判定結果が肯定である場合、ステップ S 3 の処理が実行される。一方、ステップ S 2 の判定結果が否定である場合、後述するステップ S 1 9 の処理が実行される。

【 0 1 2 9 】

ステップ S 3 において、処理部 2 1 は、タッチ入力があるかどうかを判定する。処理部 2 1 は、ステップ S 3 における判定を、メモリに記憶されている入力

50

位置データに基づいて行う。すなわち、前回の処理ループ（具体的には、ステップS1～S23の処理ループ）におけるステップS1で取得された検出結果データが、タッチ入力がないことを示す値である場合、タッチ入力はタッチオン時の入力であると判定され、当該検出結果データが、タッチ入力があることを示す値である場合、タッチ入力はタッチオン時の入力でないとして判定される。ステップS3の判定結果が肯定である場合、ステップS4の処理が実行される。一方、ステップS3の判定結果が否定である場合、ステップS4の処理がスキップされて、後述するステップS5の処理が実行される。

【0130】

ステップS4において、処理部21は、タッチオン位置が判定領域内であるか否かを判定する。すなわち、処理部21は、メモリに記憶されている入力位置データおよび判定領域データとに基づいて、入力位置データが示す入力位置（すなわち、タッチオン位置）が、判定領域データが示す判定領域内に位置するか否かを判定する。ステップS4の判定結果が肯定である場合、ステップS5の処理が実行される。一方、ステップS4の判定結果が否定である場合、ステップS5の処理がスキップされて、後述するステップS6の処理が実行される。

10

【0131】

ステップS5において、処理部21は、オブジェクト31の状態をドリフト状態へと遷移させる。すなわち、処理部21は、ドリフト状態を示すオブジェクト状態データをメモリに記憶する。ステップS5の次に、ステップS6の処理が実行される。

【0132】

ステップS6において、処理部21は、基準位置を設定する。すなわち、処理部21は、今回の処理ループにおけるステップS1の処理で取得された入力位置を、基準位置として設定する。処理部21は、設定された基準位置を示す基準位置データをメモリに記憶する。ステップS6の次に、ステップS7の処理が実行される。

20

【0133】

ステップS7において、処理部21は、基準位置を変更するか否かを判定する。すなわち、処理部21は、メモリに記憶されている入力位置データおよび基準位置データに基づいて、上記“（基準位置の変更処理）”で説明した方法によって上記の判定を行う。ステップS7の判定結果が肯定である場合、ステップS8の処理が実行される。一方、ステップS7の判定結果が否定である場合、ステップS8の処理がスキップされて、後述するステップS9の処理が実行される。

30

【0134】

ステップS8において、処理部21は、基準位置を変更する。すなわち、処理部21は、メモリに記憶されている入力位置データおよび基準位置データに基づいて、上記“（基準位置の変更処理）”で説明した方法によって、変更後の基準位置を算出する。さらに、処理部21は、メモリに記憶されている基準位置データを、算出された変更後の基準位置を示す内容に更新する。ステップS8の次に、ステップS9の処理が実行される。

【0135】

ステップS9において、処理部21は、入力指示の内容を特定する。すなわち、処理部21は、メモリに記憶されている入力位置データおよび基準位置データを参照して、入力方向および入力距離を算出する。なお、入力方向および入力距離の具体的な算出方法は、上記“[2-2-1.第1の移動操作]”で説明した方法である。処理部21は、算出された入力方向および入力距離を示す入力指示データをメモリに記憶する。ステップS9の次に、ステップS10（図14）の処理が実行される。

40

【0136】

図14を参照して、ステップS10～S14の処理について説明する。ステップS10において、処理部21は、メモリに記憶されているオブジェクト状態データに基づいて、オブジェクト31がドリフト状態であるか否かを判定する。ステップS10の判定結果が否定である場合（すなわち、オブジェクト31が通常状態である場合）、ステップS11の処理が実行される。一方、ステップS10の判定結果が肯定である場合（すなわち、オ

50

プロジェクト 3 1 がドリフト状態である場合)、ステップ S 1 2 の処理が実行される。

【 0 1 3 7 】

ステップ S 1 1 において、処理部 2 1 は、メモリに記憶されている入力指示データに基づいて、通常状態におけるオブジェクト 3 1 の移動制御を行う。具体的には、処理部 2 1 は、1 フレーム時間におけるオブジェクト 3 1 の移動方向および移動距離を算出し、算出された移動方向および移動距離と、現在のオブジェクト 3 1 の位置とに基づいて移動後の位置を算出する。なお、ステップ S 1 1 における移動制御は、上記“ [2 - 2 - 1 . 第 1 の移動操作] ”で説明した方法によって行われる。処理部 2 1 は、算出された移動後の位置を示すオブジェクト位置データをメモリに記憶する。ステップ S 1 1 の次に、後述するステップ S 2 2 の処理が実行される。

10

【 0 1 3 8 】

ステップ S 1 2 において、処理部 2 1 は、ドリフト状態におけるオブジェクト 3 1 の移動制御を行う。ステップ S 1 2 においてもステップ S 1 1 と同様、処理部 2 1 は、メモリに記憶されている入力指示データに基づいて、オブジェクト 3 1 の移動後の位置を算出する。ただし、ステップ S 1 2 における移動制御は、ステップ S 1 1 における移動制御とは異なり、上記“ [2 - 2 - 2 . 第 2 の移動操作] ”で説明した方法によって行われる。処理部 2 1 は、算出された移動後の位置を示すオブジェクト位置データをメモリに記憶する。ステップ S 1 2 の次に、ステップ S 1 3 の処理が実行される。

【 0 1 3 9 】

ステップ S 1 3 において、処理部 2 1 は、オブジェクト 3 1 がドリフト状態へと遷移してから上述の加速可能時間が経過したか否かを判定する。ステップ S 1 3 の判定結果が肯定である場合、ステップ S 1 4 の処理が実行される。一方、ステップ S 1 3 の判定結果が否定である場合、ステップ S 1 4 の処理がスキップされて、後述するステップ S 2 2 の処理が実行される。

20

【 0 1 4 0 】

ステップ S 1 4 において、処理部 2 1 は、加速フラグをオンに設定する。すなわち、処理部 2 1 は、加速フラグがオンであることを示す加速フラグデータをメモリに記憶する。これによって、オブジェクト 3 1 は、(ドリフト状態であって、かつ、) 加速可能状態に設定されたことになる。ステップ S 1 4 の次に、後述するステップ S 2 2 の処理が実行される。

30

【 0 1 4 1 】

再び図 1 3 を参照して、ステップ S 1 5 および S 1 6 の処理について説明する。ステップ S 1 6 において、処理部 2 1 は、タッチ入力終了した(すなわち、タッチオフとなった)か否かを判定する。処理部 2 1 は、ステップ S 1 5 における判定を、メモリに記憶されている入力位置データに基づいて行う。すなわち、前回の処理ループ(具体的には、ステップ S 1 ~ S 2 3 の処理ループ)におけるステップ S 1 で取得された検出結果データが、タッチ入力があることを示す値である場合、タッチ入力終了したと判定され、当該検出結果データが、タッチ入力がないことを示す値である場合、タッチ入力終了したタイミングでないと判定される。ステップ S 1 5 の判定結果が肯定である場合、ステップ S 1 6 の処理が実行される。一方、ステップ S 1 5 の判定結果が否定である場合、ステップ S 1 6 の処理がスキップされて、後述するステップ S 1 7 (図 1 4) の処理が実行される。

40

【 0 1 4 2 】

ステップ S 1 6 において、処理部 2 1 は、オブジェクト 3 1 の状態を通常状態へと遷移させる。すなわち、処理部 2 1 は、通常状態を示すオブジェクト状態データをメモリに記憶する。ステップ S 1 6 の次に、ステップ S 1 7 の処理が実行される。

【 0 1 4 3 】

再び図 1 4 を参照して、ステップ S 1 7 ~ S 2 3 の処理について説明する。ステップ S 1 7 において、処理部 2 1 は、メモリに記憶されている加速フラグデータに基づいて、加速フラグがオンに設定されているか否かを判定する。ステップ S 1 7 の判定結果が否定である場合(すなわち、加速フラグがオフに設定されている場合)、ステップ S 1 8 の処理

50

が実行される。一方、ステップS 17の判定結果が肯定である場合（すなわち、加速フラグがオンに設定されている場合）、後述するステップS 19の処理が実行される。

【0144】

ステップS 18において、処理部21は、オブジェクト31の自動制御を行う。すなわち、処理部21は、記憶部12に記憶されているゲームアプリケーションのプログラムにおいて予め定められたアルゴリズムに従ってオブジェクト31の移動を制御することによって、オブジェクト31の移動後の位置を算出する。処理部21は、算出された移動後の位置を示すオブジェクト位置データをメモリに記憶する。ステップS 18の次に、後述するステップS 22の処理が実行される。

【0145】

一方、ステップS 19において、処理部21は、加速状態でオブジェクト31の自動制御を行う。ステップS 19においてもステップS 18と同様、処理部21は、上記ゲームアプリケーションのプログラムにおいて予め定められたアルゴリズムに従ってオブジェクト31の移動を制御することによって、オブジェクト31の移動後の位置を算出する。ただし、ステップS 19においては、ステップS 18において設定される移動速度よりも速く移動するように、オブジェクト31の移動後の位置が算出される。処理部21は、算出された移動後の位置を示すオブジェクト位置データをメモリに記憶する。ステップS 19の次に、ステップS 20の処理が実行される。

【0146】

ステップS 20において、処理部21は、オブジェクト31に対する加速状態での移動制御を開始してから（換言すれば、タッチオフとなつてから）上述の加速継続時間が経過したか否かを判定する。ステップS 20の判定結果が肯定である場合、ステップS 21の処理が実行される。一方、ステップS 20の判定結果が否定である場合、ステップS 21の処理がスキップされて、後述するステップS 22の処理が実行される。

【0147】

ステップS 21において、処理部21は、加速フラグをオフに設定する。すなわち、処理部21は、加速フラグがオフであることを示す加速フラグデータをメモリに記憶する。これによって、オブジェクト31は、（通常状態であつて、かつ、）加速状態に設定されたことになる。ステップS 21の次に、ステップS 22の処理が実行される。

【0148】

ステップS 22において、処理部21は、ゲーム画像を生成して、表示部24に表示させる。すなわち、処理部21は、メモリに記憶されているオブジェクト位置データに基づいてゲーム画像を生成し、生成されたゲーム画像を表示部24に表示する。ステップS 22で生成されるゲーム画像は、オブジェクト位置データにより示される位置にオブジェクト31が配置されたゲーム空間を表す。上述のように、本実施形態においては、処理部21は、オブジェクト31が表示部24の画面における所定位置（画面の中央よりやや下側の位置。図4等参照）に配置されたゲーム画像を生成する。また、表示部24には、判定領域30の画像は表示されない。なお、ステップS 22の処理において、処理部21は、必要に応じて、他のオブジェクト（例えば、他のキャラクタやアイテム）の動作を制御し、これらのオブジェクトを含むゲーム空間を表すゲーム画像を生成する。ステップS 22の次に、ステップS 23の処理が実行される。

【0149】

ステップS 23において、処理部21は、ゲームを終了するか否かを判定する。すなわち、処理部21は、予め定められたゲーム終了条件（例えば、ユーザが操作するオブジェクトがゴールしたこと）が満たされた、または、ユーザによって所定の終了指示が行われた場合、ゲームを終了する。また、処理部21は、上記ゲーム終了条件が満たされず、かつ、ユーザによって所定の終了指示が行われていない場合、ゲームを終了しないと判定する。ステップS 23の判定結果が否定である場合、ステップS 1の処理が実行される。なお、処理部21は、ステップS 23においてゲームを終了すると判定するまで、ステップS 1～S 23の一連の処理を繰り返し実行する。一方、ステップS 23の判定結果が肯定

10

20

30

40

50

である場合、処理部 21 は、図 13 および図 14 に示すゲーム処理を終了する。

【0150】

[4. 本実施形態の作用効果および変形例]

以上に説明した実施形態によれば、ゲームプログラムは、ポインティングデバイス（例えば、タッチパネル 23）を備える情報処理装置（例えば、端末装置 2）のコンピュータに、以下の処理ステップを実行させる。

- ・表示画面上の位置に対応する位置であって、ポインティングデバイスによって検出される入力位置を繰り返し取得する取得ステップ（S1）

- ・ポインティングデバイスに対する入力が増加されたときの入力位置が所定領域（すなわち、判定領域）内であることを少なくとも条件として、仮想ゲーム空間においてオブジェクトに対する所定の第 1 処理を実行する第 1 処理実行ステップ（S5）

- ・ポインティングデバイスに対する入力に関する入力位置（例えば、入力が増加されたときの入力位置であってもよいし、入力が増加されてから所定時間が経過したときの入力位置であってもよい）に基づいて基準位置を設定する基準位置設定ステップ（S6）

- ・基準位置と、ポインティングデバイスに対する入力が増加されてから継続して行われる入力の入力位置とに基づいて、オブジェクトを移動させる移動処理を実行する移動処理ステップ（S11 または S12）

なお、「表示画面上の位置に対応する位置」とは、上記実施形態のように、ポインティングデバイスの入力面が表示画面上に設けられる（換言すれば、表示画面が入力面である）場合には、表示画面上の位置を示すものであり、表示画面上の位置と言うこともできる。また、ポインティングデバイスの入力面が表示画面とは異なる場合（例えば、ポインティングデバイスがタッチパッドである場合）には、「表示画面上の位置に対応する位置」とは、表示画面上の位置を一義的に特定することが可能な入力面上の位置である。

【0151】

上記によれば、ユーザは、所定領域内から開始される一連の入力によって、オブジェクトに対する第 1 処理と移動処理という 2 種類の処理を実行することができる。したがって、情報処理装置は、シンプルな入力方法によって多様な操作をユーザに行わせることができる。具体的には、上記実施形態によれば、ユーザは、1 本の指で可能な一連のタッチ入力によって、オブジェクトに対する複数種類の操作を行うことができる。

【0152】

また、上記によれば、入力が増加されたときの入力位置が所定領域内であるか否かによって、第 1 処理が実行されるか否かが変化する。したがって、ユーザは、入力を開始する位置を変化させることによって、異なる 2 種類の操作を使い分けることができる。この意味でも、情報処理装置は、シンプルな入力方法によって多様な操作をユーザに行わせることができる。

【0153】

なお、上記実施形態においては、基準位置設定ステップにおいては、ポインティングデバイスに対する入力が増加されたときの入力位置が第 1 領域内である場合も第 1 領域内ではない場合も（換言すれば、入力位置が第 1 領域内であるか否かにかかわらず）、基準位置が設定される。また、移動処理ステップにおいては、ポインティングデバイスに対する入力が増加されたときの入力位置が第 1 領域内である場合も第 1 領域内ではない場合も、移動処理が実行される。

【0154】

また、上記基準位置の設定は、上記実施形態のように、ポインティングデバイスに対する入力が増加されたタイミングで実行されてもよいし、後述する“（判定領域に関する第 1 の変形例）”において記載するように、入力が増加された後のタイミング（具体的には、入力位置がタッチオン位置から変化した後のタイミング）で実行されてもよい。

【0155】

なお、上記「ポインティングデバイスに対する入力が増加されてから継続して行われる入力」は、入力が厳密な意味で連続していることを要求するものではない。例えば、上述

10

20

30

40

50

の継続タッチ入力に関して言えば、タッチパネル 2 3 は、実際にはタッチ入力が行われているにもかかわらず、所定期間の間、誤検出によって一時的にタッチ入力が出検されない場合も考えられる。この場合、端末装置 2 は、当該所定期間の前において検出されていたタッチ入力と、当該所定期間の後において検出されたタッチ入力とを、まとめて 1 回の継続タッチ入力として取り扱ってもよい。具体的には、端末装置 2 は、あるタッチ入力が出検されなくなってから、次のタッチ入力が出検されるまでの期間が所定の長さよりも短い場合、当該あるタッチ入力と当該次のタッチ入力とをまとめて 1 回の継続タッチ入力として取り扱ってもよい。

【0156】

上記第 1 処理は、上記実施形態においては、オブジェクト 3 1 の状態（より具体的には、オブジェクトの移動に関する状態）を変化させる処理（ステップ S 5）であるということが出来る。ここで、他の実施形態においては、端末装置 2 は、第 1 処理として、上記移動処理とは異なる処理であって、オブジェクトの動作を制御する処理を実行してもよい。例えば、端末装置 2 は、第 1 処理として、ステップ S 5 を実行するタイミングにおいて、ステップ S 5 の処理に代えて（または、ステップ S 5 の処理とともに）、オブジェクトにジャンプまたは攻撃の動作を行わせる処理を実行してもよい。

【0157】

（判定領域に関する第 1 の変形例）

他の実施形態においては、判定領域は、表示画面上におけるオブジェクトの位置に基づいて設定されなくてもよい。図 1 5 は、上記実施形態の第 1 の変形例における判定領域の一例を示す図である。図 1 5 に示すように、第 1 の変形例においては、表示部 2 4 の画面における左右の両端に判定領域 4 0 a および 4 0 b がそれぞれ固定的に設定される。左判定領域 4 0 a は、画面の左辺から所定距離までの範囲を含む長方形の形状である。右判定領域 4 0 b は、画面の右辺から所定距離までの範囲を含む長方形の形状である。なお、以下においては、判定領域 4 0 a および 4 0 b を総称して、「判定領域 4 0」と記載することがある。

【0158】

第 1 の変形例においては上記実施形態と同様、判定領域 4 0 を示す画像は、実際には表示されない。ただし、図 1 5 に示すように、第 1 の変形例においては、左判定領域 4 0 a に左マーカ画像 4 1 a が表示され、右判定領域 4 0 b に右マーカ画像 4 1 b が表示される。これらのマーカ画像 4 1 a および 4 1 b によって、判定領域の大まかな位置（換言すれば、第 2 の移動操作のタッチ入力を開始すべき位置）をユーザに示唆することができる。

【0159】

タッチ入力第 1 の移動操作であるか第 2 の移動操作であるかの判定方法は、第 1 の変形例においても上記実施形態と同様である。すなわち、タッチオン位置が判定領域 4 0 の内側である場合、第 2 の移動操作が行われたと判定され、タッチオン位置が判定領域 4 0 の外側である場合、第 1 の移動操作が行われたと判定される。

【0160】

図 1 6 は、第 1 の変形例において第 2 の移動操作が行われるときのゲーム画像の一例を示す図である。図 1 6 は、ユーザが左判定領域 4 0 a 内の位置に対してタッチ入力を開始して、その後、入力位置を画面中央へ方向（すなわち、右方向）へ移動させた状態におけるゲーム画像を示している。なお、第 1 の変形例においても上記実施形態と同様、移動操作のためのタッチ入力が行われたことに応じて操作画像 3 2 が表示部 2 4 に表示される。

【0161】

第 1 の変形例においては、第 2 の移動操作が行われたと判定された場合、各判定領域 4 0 a および 4 0 b のうち、タッチ入力開始された判定領域内に配置されるマーカ画像が、画面中央の方向へ移動して表示される。図 1 6 に示す例においては、左判定領域 4 0 a 内の左マーカ画像 4 1 a が、画面中央へ方向（すなわち、右方向）へ移動して表示される。このように、第 1 の変形例においては、端末装置 2 は、第 2 の移動操作が行われたことに応じてマーカ画像を移動させて表示する。これによって、端末装置 2 は、判定領域内

10

20

30

40

50

をタッチオン位置とする入力を行っているか（つまり、第2の移動操作を行っているか）否かを、ユーザに対してわかりやすく認識させることができる。

【0162】

第1の変形例においては、第2の移動操作が行われる場合、タッチオン位置は表示画面の領域のうちの左右の端部のいずれかとなる。したがって、第1の変形例においては、端末装置2は、タッチオン位置が判定領域40内である場合、タッチオン位置を、画面の端部から所定距離となるように画面の中央方向へ移動させた位置を基準位置として設定する。例えば、基準位置は、判定領域40外の位置となるように設定されてもよい。これによって、ユーザは、基準位置から任意の方向へ所定距離だけ離れた入力位置を容易に指定することができる。例えば、タッチオン位置が画面の左端部付近（すなわち、左判定領域40a内）である場合、基準位置はタッチオン位置よりも右側に設定されるので、ユーザは、（画面の左端部付近からタッチ入力を開始した場合でも）基準位置に対して左方向を入力方向とするタッチ入力を容易に行うことができる。

10

【0163】

なお、他の実施形態においては、端末装置2は、第2の移動操作のタッチオン時にはオブジェクト31のドリフト状態への遷移を行わず、第2の移動操作中においてタッチオン位置（ここでは、基準位置とは異なる）と入力位置との距離が所定のドリフト開始距離以上となったことに応じて、ドリフト状態への遷移を行うようにしてもよい。なお、入力位置とタッチオン位置との距離は、入力面における2次元の距離であってもよいし、入力面における所定の方向（例えば、x軸成分）の距離であってもよい。また、端末装置2は、タッチオン時においては、タッチオン位置から画面の中央方向へ移動した位置に基準位置を設定し、タッチオン位置と入力位置との距離がドリフト開始距離以上となったことに応じて（すなわち、オブジェクト31がドリフト状態へ遷移したことに応じて）、タッチオン位置に基準位置を設定（換言すれば、変更）するようにしてもよい。

20

【0164】

ここで、第1の変形例においては、第2の移動操作のタッチオン時（換言すれば、ドリフト状態への遷移時）において、入力位置と基準位置とが異なるので、ドリフト状態への遷移タイミングで急に入力距離が大きくなる結果、ユーザが第2の移動操作に違和感を抱く可能性がある。これに対して、上記の他の実施形態によれば、ドリフト状態への遷移タイミングではタッチオン位置が基準位置となるので、当該遷移タイミングで急に入力距離が大きくなる可能性を低減することができる。そのため、第2の移動操作においてユーザが抱く違和感を低減することができ、第2の移動操作の操作性を向上することができる。

30

【0165】

（判定領域に関する第2の変形例）

図17は、上記実施形態の第2の変形例における判定領域の一例を示す図である。図17に示すように、第2の変形例においては、表示部24にはスライダ画像45が表示され、このスライダ画像45の領域が判定領域として設定される。つまり、第2の変形例においては、スライダ画像45の内側がタッチ入力のタッチオン位置となる場合、第2の移動操作が行われたと判定され、スライダ画像45の外側がタッチ入力のタッチオン位置となる場合、第1の移動操作が行われたと判定される。このように、端末装置2は、判定領域を表す画像を表示部24に表示してもよい。

40

【0166】

図18は、第2の変形例において第2の移動操作が行われるときのゲーム画像の一例を示す図である。図18は、ユーザがスライダ画像45（換言すれば、判定領域）内の位置に対してタッチ入力を開始して、その後、入力位置を右方向へ移動させた状態におけるゲーム画像を示している。なお、第1の変形例においても上記実施形態と同様、移動操作のためのタッチ入力が行われたことに応じて操作画像32が表示部24に表示される。

【0167】

図18に示すように、スライダ画像45はマーカ画像46を含む。第2の変形例においては、端末装置2は、第2の移動操作中において、入力方向および入力距離に応じた位置

50

にマーカ画像 4 6 を移動して表示する。図 1 8 に示す例においては、入力方向は右方向であるので、マーカ画像 4 6 は、スライダ画像 4 5 の中央から右側に移動した位置に表示される。このように、第 2 の変形例においては、第 2 の移動操作中においては、操作画像 3 2 に加えてマーカ画像 4 6 が移動して表示される。これによって、端末装置 2 は、判定領域内をタッチオン位置とする入力を行っているか（つまり、第 2 の移動操作を行っているか）否かを、ユーザに対してわかりやすく認識させることができる。

【0168】

なお、第 2 の変形例においては、タッチ入力第 1 の移動操作であるか第 2 の移動操作であるかの判定方法や、移動操作に基づくオブジェクトの移動制御の方法等は、上記実施形態と同様である。

【0169】

（ゲーム内容に関する変形例）

他の実施形態においては、端末装置 2 において実行されるゲームの内容は、レースゲームに限らず、任意の内容のゲームであってもよい。例えば、他の実施形態においては、端末装置 2 において実行されるゲームは、アクションゲームにおけるオブジェクト（すなわち、プレイヤーキャラクタ）を、ポインティングデバイスに対する入力によって操作するゲームであってもよい。例えば、端末装置 2 は、ステップ S 1 1 の移動処理において上記オブジェクトを歩いて移動させ、ステップ S 1 2 の移動処理において当該オブジェクトを走って移動させるように、移動制御を行ってもよい。このとき、端末装置 2 は、オブジェクトの移動方法に関する状態（すなわち、歩くまたは走る）を管理し、ステップ S 5 において、オブジェクトの状態を、歩く状態から走る状態へと遷移させてもよい。さらにこのとき、端末装置 2 は、ステップ S 1 9 の処理において、ステップ S 1 2 の移動処理における速度よりも速い速度でオブジェクトが走るように移動制御を行ってもよい。

【0170】

また、他の実施形態においては、端末装置 2 において実行されるゲームは、射撃ゲームにおけるオブジェクト（すなわち、プレイヤーキャラクタ）を、ポインティングデバイスに対する入力によって操作するゲームであってもよい。例えば、端末装置 2 は、各移動操作を、射撃に関する動作をオブジェクトに行わせる処理に適用することができる。具体的には、端末装置 2 は、第 1 の移動操作に応じて照準を移動させるとともに、第 1 の移動操作のタッチオフに応じて弾を発射させる。また、端末装置 2 は、第 2 の移動操作のタッチオンに応じて、パワーを溜める処理を実行し、その後の第 2 の移動操作に応じて、照準を移動させるとともに、第 2 の移動操作のタッチオフに応じて弾を発射させる。なお、上記の例において、判定領域は、例えば、オブジェクトが持つ銃のオブジェクトを含むように設定されてもよい。上記のゲーム例によれば、ユーザは、オブジェクトに射撃動作を行わせるための操作において、第 2 の移動操作であるか第 1 の移動操作であるかによって、パワーを溜めてから弾を発射するか否かを使い分けることができる。

【産業上の利用可能性】

【0171】

上記実施形態は、ポインティングデバイスを用いて多様なゲーム操作を可能とすること等を目的として、例えばゲームプログラムやゲーム装置等として利用することができる。

【符号の説明】

【0172】

- 1 サーバ
- 2 端末装置
 - 2 1 処理部
 - 2 2 記憶部
 - 2 3 タッチパネル
 - 2 4 表示部
 - 2 5 通信部
- 3 0 判定領域

10

20

30

40

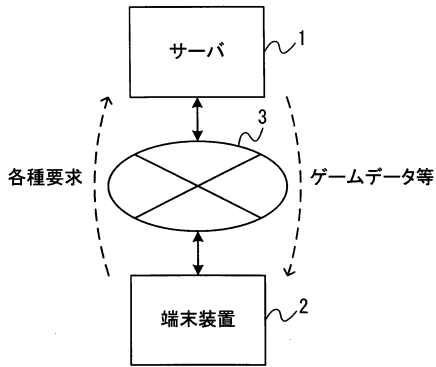
50

3 1 オブジェクト

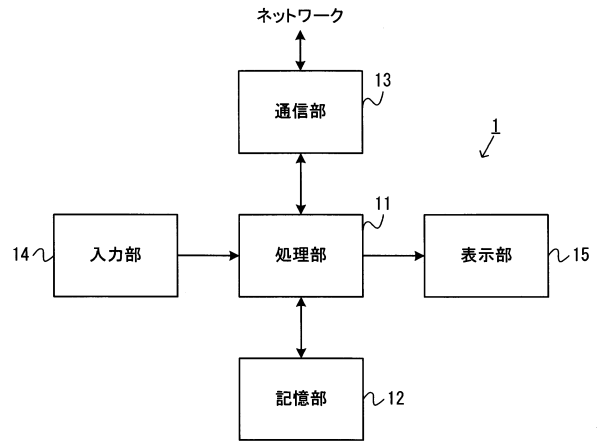
3 2 操作画像

【図面】

【図 1】



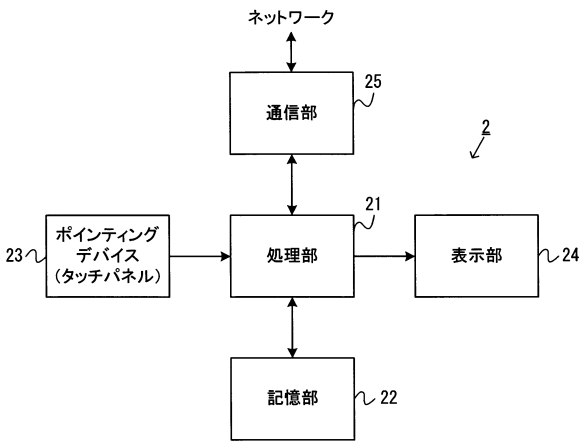
【図 2】



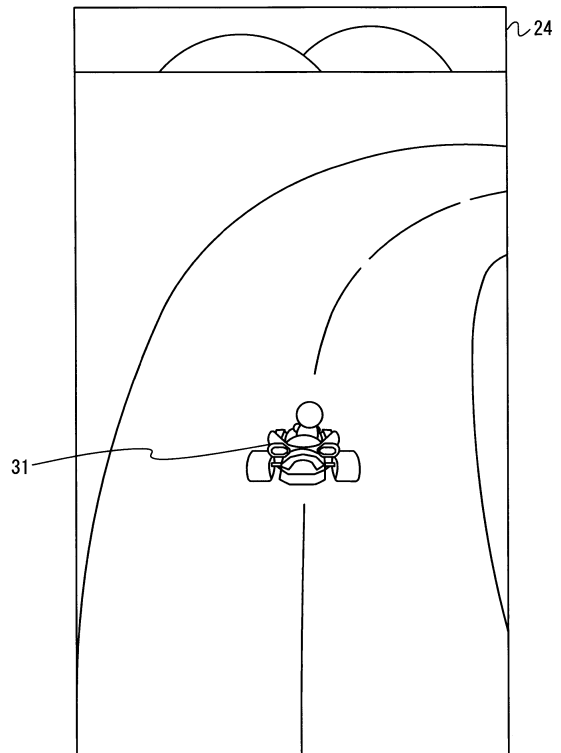
10

20

【図 3】



【図 4】

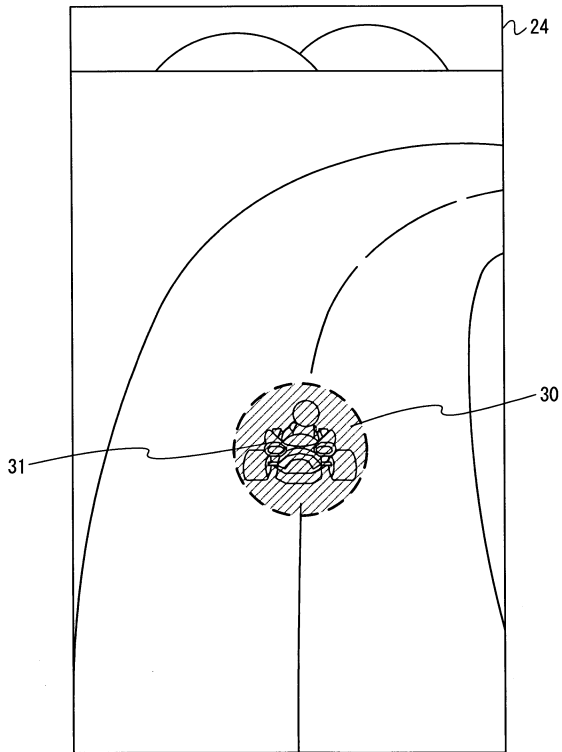


30

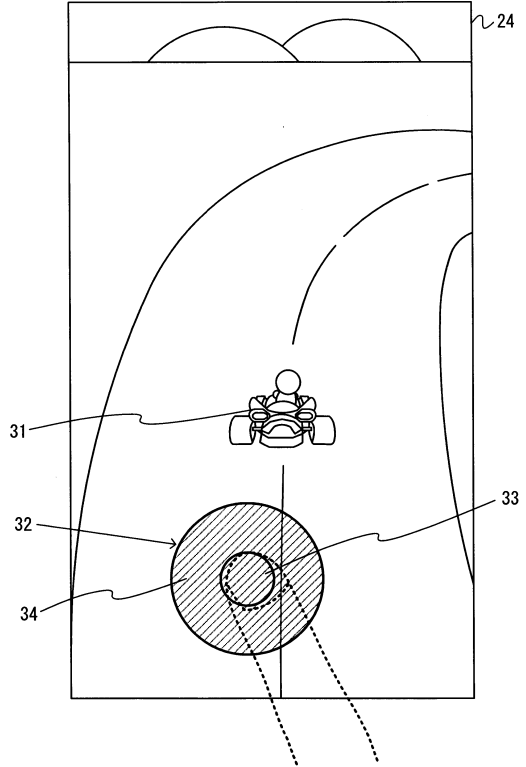
40

50

【図 5】



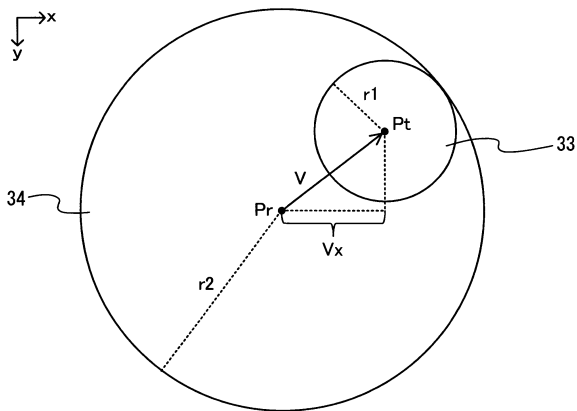
【図 6】



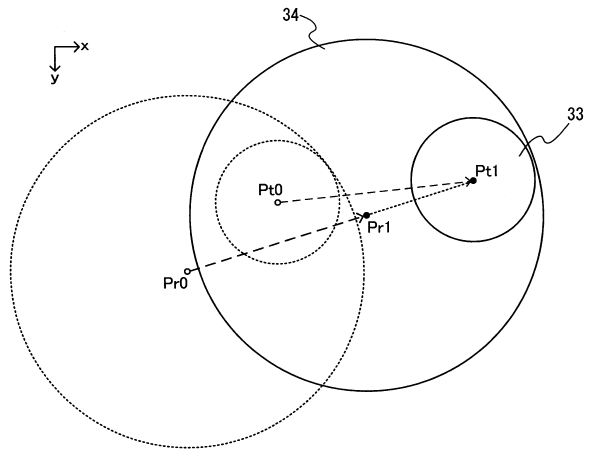
10

20

【図 7】



【図 8】

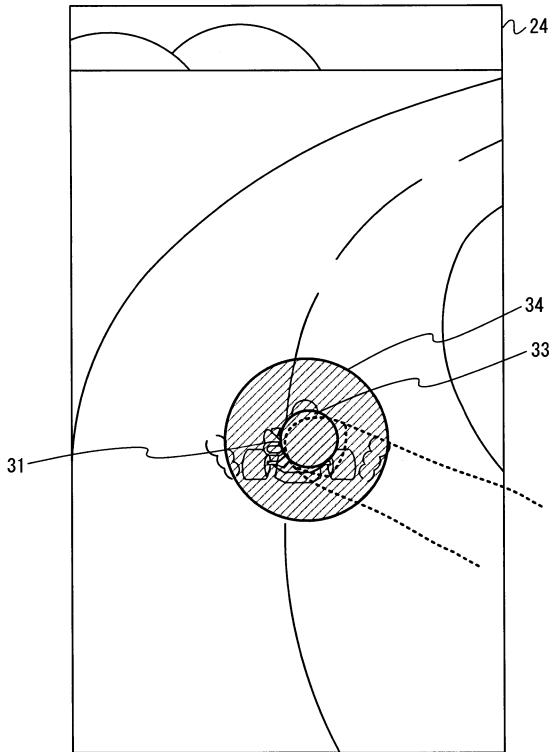


30

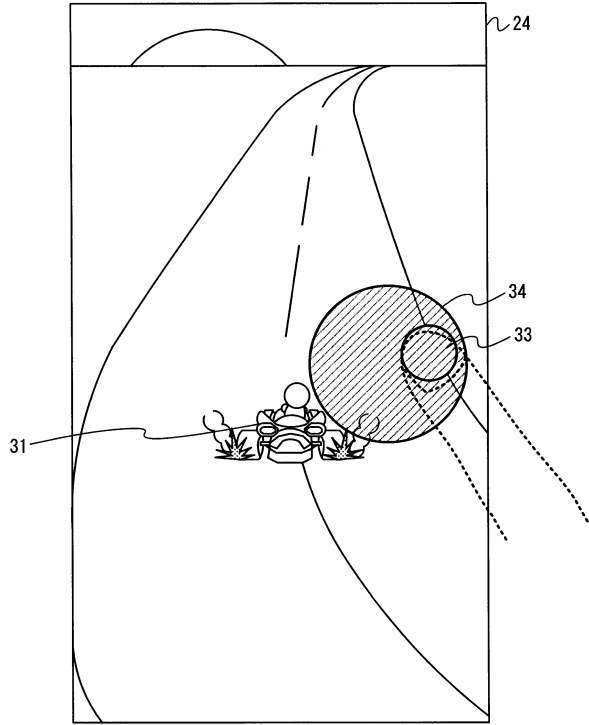
40

50

【 図 9 】



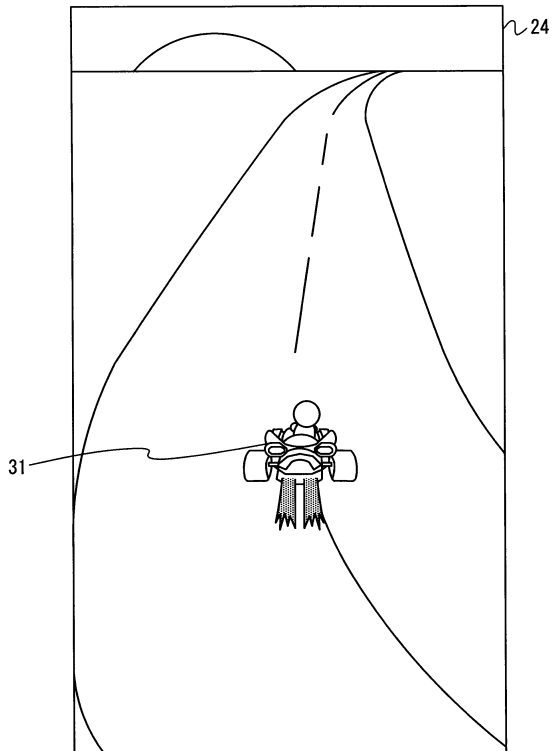
【 図 1 0 】



10

20

【 図 1 1 】



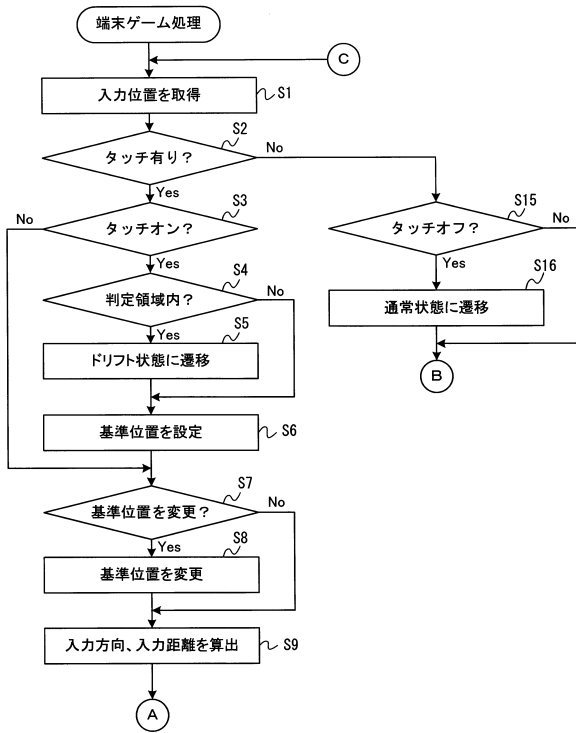
【 図 1 2 】

入力位置データ
基準位置データ
入力指示データ
オブジェクト位置データ
オブジェクト状態データ
判定領域データ
加速フラグデータ

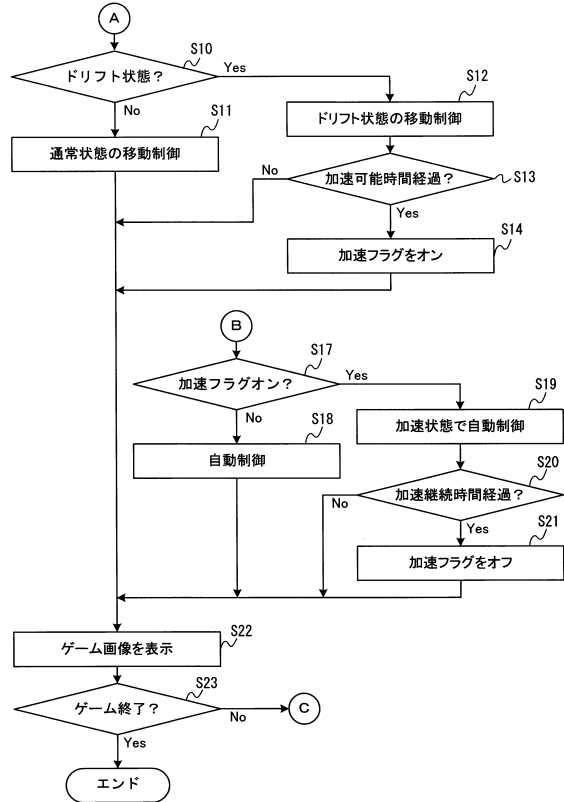
30

40

【図13】



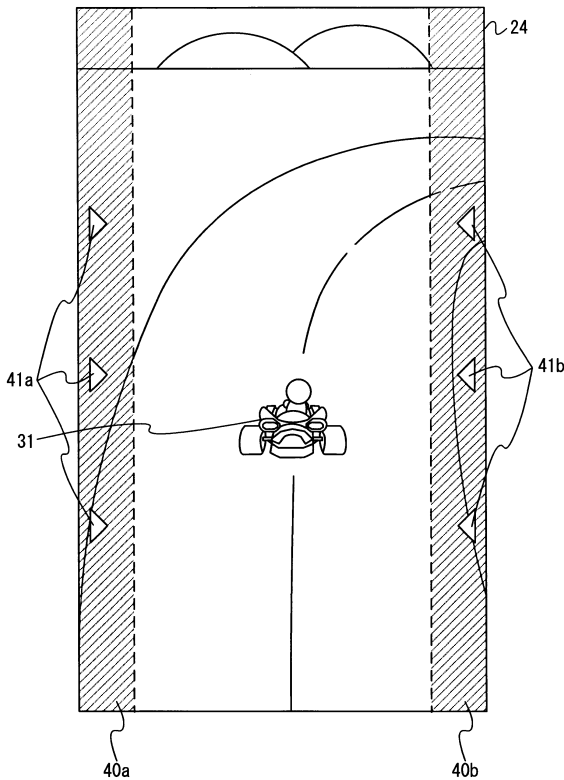
【図14】



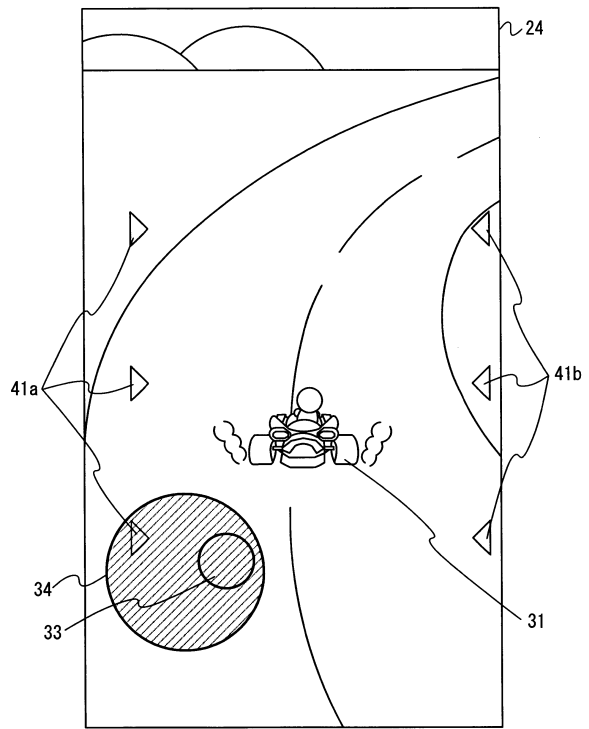
10

20

【図15】



【図16】

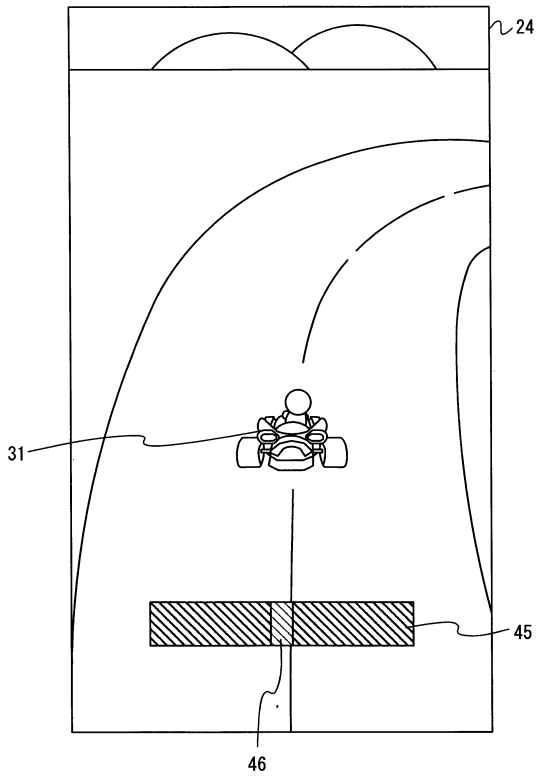


30

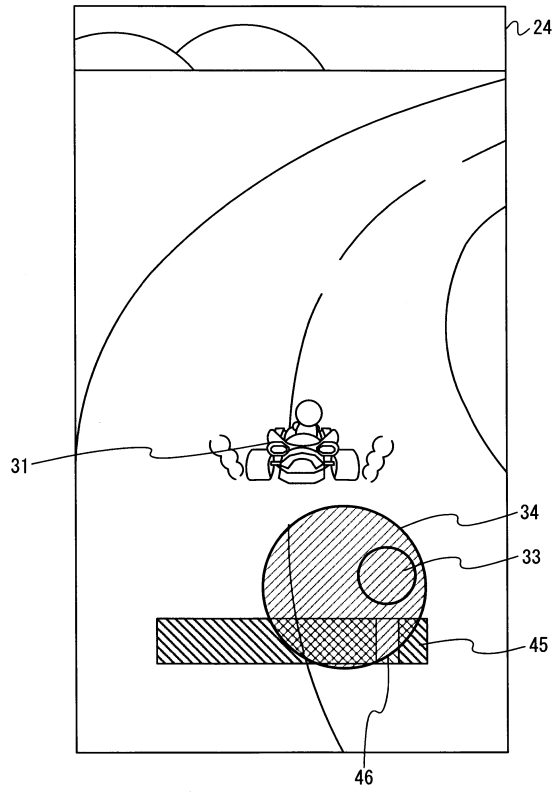
40

50

【 図 17 】



【 図 18 】



10

20

30

40

50

フロントページの続き

(51)国際特許分類

G 0 6 F 3/0488(2022.01)

F I

G 0 6 F 3/0488

1 任天堂株式会社内

(72)発明者 村田 茂人

東京都江東区永代2 - 37 - 25 株式会社バンダイナムコスタジオ内

(72)発明者 中野 慎也

東京都江東区永代2 - 37 - 25 株式会社バンダイナムコスタジオ内

審査官 宇佐田 健二

(56)参考文献

特開2016 - 218774 (JP, A)

特開2011 - 000167 (JP, A)

特開2002 - 000939 (JP, A)

特開2016 - 206740 (JP, A)

五十嵐 達雄, 「任天堂ゲーム攻略本 Nintendo DREAM マリオカートDS」,

初版第1刷, 日本, 株式会社毎日コミュニケーションズ, 2006年01月25日, pp.2-5, ISBN

:4-8399-1983-6

T.S, “ 簡単操作で本格ドリフトレース! 「ドリフトスピリッツ」 ”, アプリノ, 日本, 株式会社Lighthouse Studio, 2016年08月01日, pp.1-5, <https://applinote.com/ios/article/219363>, [2022年08月31日検索]

(58)調査した分野 (Int.Cl., DB名)

A 6 3 F 13 / 0 0 - 13 / 9 8 , 9 / 2 4

G 0 6 F 3 / 0 4 8 - 3 / 0 4 8 9 5