

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5758085号
(P5758085)

(45) 発行日 平成27年8月5日(2015.8.5)

(24) 登録日 平成27年6月12日(2015.6.12)

(51) Int.Cl.		F 1
A 6 3 F 13/577 (2014.01)		A 6 3 F 13/577
A 6 3 F 13/428 (2014.01)		A 6 3 F 13/428
A 6 3 F 13/211 (2014.01)		A 6 3 F 13/211

請求項の数 9 (全 41 頁)

(21) 出願番号	特願2010-115795 (P2010-115795)	(73) 特許権者	000233778
(22) 出願日	平成22年5月19日 (2010.5.19)		任天堂株式会社
(65) 公開番号	特開2011-240000 (P2011-240000A)		京都府京都市南区上鳥羽鉾立町1番地1
(43) 公開日	平成23年12月1日 (2011.12.1)	(74) 代理人	100158780
審査請求日	平成25年4月22日 (2013.4.22)		弁理士 寺本 亮
		(74) 代理人	100121359
			弁理士 小沢 昌弘
		(74) 代理人	110001276
			特許業務法人 小笠原特許事務所
		(74) 代理人	100098291
			弁理士 小笠原 史朗
		(74) 代理人	100151541
			弁理士 高田 猛二
		(74) 代理人	100130269
			弁理士 石原 盛規

最終頁に続く

(54) 【発明の名称】 ゲームプログラム、ゲーム装置、ゲームシステム、及びゲーム処理方法

(57) 【特許請求の範囲】

【請求項1】

入力装置の位置及び姿勢の少なくとも一方を検出する状態検出手段から出力された状態信号に基づいてゲーム処理を行うゲーム装置のコンピュータを、

前記状態信号に基づいて、前記入力装置の動作方向を判別する方向判別手段と、

前記状態信号に基づいて仮想ゲーム空間内で動作する第1オブジェクトが判定対象と衝突するか否かを判定する衝突判定手段と、

前記第1オブジェクトが前記判定対象に衝突すると前記衝突判定手段によって判定された場合に、前記判定対象が、当該判定対象に対して前記第1オブジェクトがヒット可能なヒット方向を示すデータである指向性データを有しているか否かを判定する条件判定手段と、

前記判定対象が前記指向性データを有していると前記条件判定手段によって判定された場合に、前記方向判別手段によって判別された動作方向と前記判定対象が有する指向性データとに基づいて、当該判定対象に対してヒット処理を行うか否かを決定する衝突処理決定手段と、

前記判定対象に対してヒット処理を行うと前記衝突処理決定手段によって決定された場合、又は前記判定対象が前記指向性データを有していないと前記条件判定手段によって判定された場合に、前記判定対象に対してヒット処理を行うヒット処理手段として機能させる、ゲームプログラム。

【請求項2】

前記ゲーム装置のコンピュータを、前記仮想ゲーム空間内における前記第1オブジェクトと前記判定対象との位置関係を検出する位置関係検出手段として更に機能させ、

前記衝突処理決定手段は、前記方向判別手段によって判別された動作方向、及び前記位置関係検出手段によって検出された位置関係に基づいて、前記判定対象に対してヒット処理を行うか否かを決定する、請求項1に記載のゲームプログラム。

【請求項3】

前記ゲーム装置のコンピュータを、前記第1オブジェクトが複数の判定対象と衝突すると前記衝突判定手段によって判定された場合に、前記方向判別手段によって判別された動作方向、及び前記位置関係検出手段によって検出された位置関係に基づいて、前記複数の判定対象に対して優先順位を設定する優先順位設定手段として更に機能させ、

10

前記衝突処理決定手段は、前記優先順位が最上位の判定対象に対してヒット処理を行うか否かを決定する、請求項2に記載のゲームプログラム。

【請求項4】

前記ゲーム装置のコンピュータを、前記状態信号に基づいて、前記入力装置の振り操作が開始されたか否かを判定する開始判定手段として更に機能させ、

前記方向判別手段は、前記入力装置の振り操作が開始されたときと前記開始判定手段によって判定されたときに、当該入力装置の動作方向を判別する、請求項1に記載のゲームプログラム。

【請求項5】

前記状態検出手段は、角速度センサを含み、

20

前記状態信号は、前記角速度センサから出力される角速度データを含む、請求項1に記載のゲームプログラム。

【請求項6】

前記ゲーム装置のコンピュータを、前記ヒット処理を行わないと前記衝突処理決定手段によって決定された場合に、前記判定対象と衝突する前後で前記第1オブジェクトの動作方向を変化させる動作方向変化手段として更に機能させる、請求項1に記載のゲームプログラム。

【請求項7】

入力装置の位置及び姿勢の少なくとも一方を検出する状態検出手段から出力された状態信号に基づいてゲーム処理を行うゲーム装置であって、

30

前記状態信号に基づいて、前記入力装置の動作方向を判別する方向判別手段と、

前記状態信号に基づいて仮想ゲーム空間内で動作する第1オブジェクトが判定対象と衝突するか否かを判定する衝突判定手段と、

前記第1オブジェクトが前記判定対象に衝突すると前記衝突判定手段によって判定された場合に、前記判定対象が、当該判定対象に対して前記第1オブジェクトがヒット可能なヒット方向を示すデータである指向性データを有しているか否かを判定する条件判定手段と、

前記判定対象が前記指向性データを有していると前記条件判定手段によって判定された場合に、前記方向判別手段によって判別された動作方向と前記判定対象が有する指向性データとに基づいて、当該判定対象に対してヒット処理を行うか否かを決定する衝突処理決定手段と、

40

前記判定対象に対してヒット処理を行うと前記衝突処理決定手段によって決定された場合、又は前記判定対象が前記指向性データを有していないと前記条件判定手段によって判定された場合に、前記判定対象に対してヒット処理を行うヒット処理手段とを備える、ゲーム装置。

【請求項8】

入力装置の位置及び姿勢の少なくとも一方を検出する状態検出手段から出力された状態信号に基づいてゲーム処理を行うゲームシステムであって、

前記状態信号に基づいて、前記入力装置の動作方向を判別する方向判別手段と、

前記状態信号に基づいて仮想ゲーム空間内で動作する第1オブジェクトが判定対象と衝

50

突するか否かを判定する衝突判定手段と、

前記第1オブジェクトが前記判定対象に衝突すると前記衝突判定手段によって判定された場合に、前記判定対象が、当該判定対象に対して前記第1オブジェクトがヒット可能なヒット方向を示すデータである指向性データを有しているか否かを判定する条件判定手段と、

前記判定対象が前記指向性データを有していると前記条件判定手段によって判定された場合に、前記方向判別手段によって判別された動作方向と前記判定対象が有する指向性データとに基づいて、当該判定対象に対してヒット処理を行うか否かを決定する衝突処理決定手段と、

前記判定対象に対してヒット処理を行うと前記衝突処理決定手段によって決定された場合、又は前記判定対象が前記指向性データを有していないと前記条件判定手段によって判定された場合に、前記判定対象に対してヒット処理を行うヒット処理手段とを備える、ゲームシステム。

【請求項9】

入力装置の位置及び姿勢の少なくとも一方を検出する状態検出手段から出力された状態信号に基づいてゲーム処理を行うためのゲーム処理方法であって、

前記状態信号に基づいて、前記入力装置の動作方向を判別するステップと、

前記状態信号に基づいて仮想ゲーム空間内で動作する第1オブジェクトが判定対象と衝突するか否かを判定するステップと、

前記第1オブジェクトが前記判定対象に衝突すると判定された場合に、前記判定対象が、当該判定対象に対して前記第1オブジェクトがヒット可能なヒット方向を示すデータである指向性データを有しているか否かを判定するステップと、

前記判定対象が前記指向性データを有していると判定された場合に、前記判別された動作方向と前記判定対象が有する指向性データとに基づいて、当該判定対象に対してヒット処理を行うか否かを決定するステップと、

前記判定対象に対してヒット処理を行うと決定された場合、又は前記判定対象が前記指向性データを有していないと判定された場合に、前記判定対象に対してヒット処理を行うステップとを含む、ゲーム処理方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ゲーム装置のコンピュータに実行させるゲームプログラム、ゲーム装置、ゲームシステム、及びゲーム処理方法に関し、より特定的には、仮想ゲーム空間内のオブジェクト同士の衝突判定を行い、その判定結果に応じてゲーム処理を行うゲーム装置のコンピュータに実行させるゲームプログラム、ゲーム装置、ゲームシステム、及びゲーム処理方法に関する。

【背景技術】

【0002】

いわゆるシューティングゲームやアクションゲームでは、仮想ゲーム空間内に登場するプレイヤーオブジェクトをプレイヤーが操作するコントローラの動きに応じて移動させ、そのプレイヤーオブジェクトが他のオブジェクトと衝突したか否かを判定する衝突判定が行われている（例えば特許文献1及び特許文献2参照）。

【0003】

特許文献1に記載のゲーム装置では、コントローラに内蔵された加速度センサから出力される加速度データに基づいてコントローラの動きが検出され、検出された動きがゲーム処理に反映される。具体的には、プレイヤーによってコントローラが振り動かされると、仮想ゲーム空間内の剣オブジェクト（プレイヤーオブジェクト）がコントローラの振り方向に応じた方向に振り動かされる。その際、剣オブジェクトが丸太オブジェクトに衝突したか否かが判定される。そして、剣オブジェクトが丸太オブジェクトに衝突したと判定された場合、丸太オブジェクトが上記振り方向に応じた方向に切断される。

10

20

30

40

50

【 0 0 0 4 】

特許文献 2 に記載のゲーム装置は、コントローラの位置を検出するための CCD カメラを備えている。コントローラには、赤外光を発する LED が設けられており、ゲーム装置では、この赤外光を CCD カメラで検出することによってコントローラの位置が検出され、その検出結果がゲーム処理に反映される。具体的には、プレイヤーによってコントローラが振り動かされると、CCD カメラによる赤外光の検出結果に応じて、剣を振ったり、或いは剣で突くといった仮想ゲーム空間内の剣オブジェクトの動作が画面に表示される。これに伴い、剣オブジェクトが例えば敵オブジェクトに衝突したか否かが判定される。そして、衝突したと判定されると、剣オブジェクトの速度、及び敵オブジェクトに剣オブジェクトのどの部分が衝突したかに基づいて、当たり度の判定が行われる。ここで、当たり度の判定は、剣オブジェクトの動作によって、「服だけが切れた」、「敵オブジェクトの身が切れた」、及び「敵オブジェクトの骨まで切れた」のいずれの結果になったかを判定するものである。

10

【先行技術文献】

【特許文献】

【 0 0 0 5 】

【特許文献 1】特開 2 0 0 8 - 1 7 3 3 4 5 号公報

【特許文献 2】特許第 4 0 0 9 4 3 3 号明細書

【発明の概要】

【発明が解決しようとする課題】

20

【 0 0 0 6 】

ところで、上述した従来のゲーム装置では、コントローラの動きに基づいて剣オブジェクトが他のオブジェクトと衝突したときの反応がリアルに表現される。しかしながら、従来のゲーム装置は、コントローラの振り方向（動作方向）に応じて、剣オブジェクト等の衝突オブジェクトが丸太や敵等の被衝突オブジェクトに衝突した際の反応を示す演出を容易に切り替えられるものではなかった。

【 0 0 0 7 】

それ故に、本発明の目的は、入力装置の動作方向に応じて、被衝突オブジェクトの反応を示す演出を容易に切り替えることを可能にするゲームプログラム、ゲーム装置、ゲームシステム、及びゲーム処理方法を提供することである。

30

【課題を解決するための手段】

【 0 0 0 8 】

本発明に係るゲームプログラムは、入力装置の位置及び姿勢の少なくとも一方を検出する状態検出手段から出力された状態信号に基づいてゲーム処理を行うゲーム装置のコンピュータを、方向判別手段、衝突判定手段、及び衝突処理決定手段として機能させる。方向判別手段は、状態信号に基づいて、入力装置の動作方向を判別する。衝突判定手段は、状態信号に基づいて仮想ゲーム空間内で動作する第 1 オブジェクトが判定対象と衝突するかどうかを判定する。衝突処理決定手段は、第 1 オブジェクトが判定対象に衝突すると衝突判定手段によって判定された場合に、方向判別手段によって判別された動作方向に基づいて、その判定対象に対してヒット処理を行うか否かを決定する。

40

【 0 0 0 9 】

プレイヤーによって操作される入力装置は、状態検出手段によってその状態が検出される。状態検出手段の検出結果である状態信号は、ゲーム装置に入力される。ゲーム装置では、この状態信号に基づいて、プレイヤーによって操作される仮想ゲーム空間内の第 1 オブジェクトを移動させたり第 1 オブジェクトの姿勢を変化させたりするゲーム処理が行われる。なお、プレイヤーが入力装置の位置や姿勢を変化させることによって、ゲーム装置に入力される状態信号が変化する。このため、状態信号の変化を監視することによって、入力装置の位置や姿勢を変化させるプレイヤーの操作を、第 1 オブジェクトの位置や姿勢に正確に反映させることが可能である。

【 0 0 1 0 】

50

ゲーム装置では、このようなゲーム処理に際して、第1オブジェクトが判定対象と衝突するか否かを判定する衝突判定が衝突判定手段によって実行される。ここで、判定対象は、仮想ゲーム空間内で第1オブジェクトが接触する可能性がある敵オブジェクトや敵オブジェクトが所持する武器等のいわゆるノンプレイヤオブジェクトに対して設定された判定領域である。また、ゲーム装置では、この衝突判定と共に、入力装置の動作方向（入力装置が動かされた方向）が状態信号に基づいて判別される。第1オブジェクトが判定対象に衝突すると衝突判定手段によって判定された場合、判別された入力装置の動作方向に基づいて、その判定対象に対してヒット処理を行うか否かが衝突処理決定手段によって決定される。ここで、ヒット処理は、第1オブジェクトが判定対象にヒットしたことを示す演出を画面に表示するための処理である。ヒット処理を行うと決定された場合、例えば第1オブジェクトとしての剣オブジェクトが敵オブジェクトにヒットして敵オブジェクトに対してダメージが与えられる様子を示すアニメーションが画面に表示される。一方、ヒット処理を行わないと決定された場合、例えば第1オブジェクトとしての剣オブジェクトが敵オブジェクトに弾き返される様子を示すアニメーションが画面に表示される。

10

【0011】

ゲーム装置のコンピュータを、仮想ゲーム空間内における第1オブジェクトと判定対象との位置関係を検出する位置関係検出手段として更に機能させてもよい。この場合、衝突処理決定手段は、方向判別手段によって判別された動作方向、及び位置関係検出手段によって検出された位置関係に基づいて、判定対象に対してヒット処理を行うか否かを決定する。

20

【0012】

この構成によれば、入力装置の動作方向に加えて、第1オブジェクトと判定対象との位置関係を考慮して、判定対象に対してヒット処理を行うか否かが決定される。このため、第1オブジェクトと判定対象との位置関係が変化したとしても判定対象に対してヒット処理を行うか否かが適切に決定される。

【0013】

ゲーム装置のコンピュータを、第1オブジェクトが複数の判定対象と衝突すると衝突判定手段によって判定された場合に、方向判別手段によって判別された動作方向、及び位置関係検出手段によって検出された位置関係に基づいて、複数の判定対象に対して優先順位を設定する優先順位設定手段として更に機能させてもよい。この場合、衝突処理決定手段は、優先順位が最上位の判定対象に対してヒット処理を行うか否かを決定する。

30

【0014】

この構成によれば、例えば複数の判定対象のうち第1オブジェクトが最初に衝突する判定対象の優先順位が最も高くなるように優先順位を設定して、最優先の判定対象に対してヒット処理を行うか否かを決定することが可能となる。このため、ヒット処理を行うか否かを決定すべき判定対象に対して、ヒット処理を行うか否かが適切に決定される。

【0015】

ゲーム装置のコンピュータを、状態信号に基づいて、入力装置の振り操作が開始されたか否かを判定する開始判定手段として更に機能させてもよい。この場合、方向判別手段は、入力装置の振り操作が開始されたとき開始判定手段によって判定されたときに、入力装置の動作方向を判別する。

40

【0016】

この構成によれば、第1オブジェクトが判定対象に衝突したときに入力装置の動作方向を判別する場合に比べて、判定対象に対してヒット処理を行うか否かをより早いタイミングで決定することが可能になる。その結果、入力装置の振り操作が行われてから衝突時の演出が画面に表示されるまでのタイムラグが短縮される。

【0017】

状態検出手段は、角速度センサを含み、状態信号は、角速度センサから出力される角速度データを含んでいてもよい。

【0018】

50

この構成によれば、例えば状態検出手段として加速度センサを使用する場合に比べて、入力装置の動作方向の判別、及び第1オブジェクトが判定対象と衝突するか否かの判定をより正確に行うことが可能になる。その結果、判定対象に対してヒット処理を行うか否かの判定が、より正確に行われる。

【0019】

本発明は、入力装置の位置及び姿勢の少なくとも一方を検出する状態検出手段から出力された状態信号に基づいてゲーム処理を行うゲーム装置として捉えることもできる。このゲーム装置は、方向判別手段、衝突判定手段、及び衝突処理決定手段を備える。方向判別手段は、状態信号に基づいて、入力装置の動作方向を判別する。衝突判定手段は、状態信号に基づいて仮想ゲーム空間内で動作する第1オブジェクトが判定対象と衝突するか否かを判定する。衝突処理決定手段は、第1オブジェクトが判定対象に衝突すると衝突判定手段によって判定された場合に、方向判別手段によって判別された動作方向に基づいて、その判定対象に対してヒット処理を行うか否かを決定する。

10

【0020】

また、本発明は、入力装置の位置及び姿勢の少なくとも一方を検出する状態検出手段から出力された状態信号に基づいてゲーム処理を行うゲームシステムとして捉えることもできる。このゲームシステムは、方向判別手段、衝突判定手段、及び衝突処理決定手段を備える。方向判別手段は、状態信号に基づいて、入力装置の動作方向を判別する。衝突判定手段は、状態信号に基づいて仮想ゲーム空間内で動作する第1オブジェクトが判定対象と衝突するか否かを判定する。衝突処理決定手段は、第1オブジェクトが判定対象に衝突すると衝突判定手段によって判定された場合に、方向判別手段によって判別された動作方向に基づいて、その判定対象に対してヒット処理を行うか否かを決定する。

20

【0021】

また、本発明は、入力装置の位置及び姿勢の少なくとも一方を検出する状態検出手段から出力された状態信号に基づいて例えばゲーム装置のコンピュータがゲーム処理を行うためのゲーム処理方法として捉えることもできる。このゲーム処理方法では、状態信号に基づいて、入力装置の動作方向が判別される。また、状態信号に基づいて仮想ゲーム空間内で動作する第1オブジェクトが判定対象と衝突するか否かが判定される。そして、第1オブジェクトが判定対象に衝突すると判定された場合に、判別された動作方向に基づいて、その判定対象に対してヒット処理を行うか否かが決定される。

30

【0022】

なお、入力装置の動作方向の判別は、第1オブジェクトが判定対象と衝突するか否かの衝突判定に先立って行われてもよいし、衝突判定の後に行われてもよい。

【発明の効果】

【0023】

この発明によれば、入力装置の動作方向に応じて、判定対象に対してヒット処理を行うか否かが決定されるので、入力装置の動作方向に応じて、被衝突オブジェクトの反応を示す演出を容易に切り替えることができる。

【図面の簡単な説明】

【0024】

【図1】本発明の一実施形態に係るゲーム装置3を含むゲームシステム1の外観斜視図

【図2】第1コントローラ7及びジャイロセンサユニット9を上面後方から見た斜視図

【図3】第1コントローラ7を下面前方から見た斜視図

【図4】第2コントローラ8の斜視図

【図5】ゲーム装置3の構成を示すブロック図

【図6】コントローラ5の内部構成を示すブロック図

【図7】ジャイロセンサユニット9によって角速度が検出されるロール角、ピッチ角、及びヨー角について説明するための第1コントローラ7及びジャイロセンサユニット9の斜視図

【図8】プレイヤーが第1コントローラ7及び第2コントローラ8を持った状態を例示する

40

50

説明図

- 【図 9】液晶テレビ 2 に表示されるゲーム画像の一例を示す画面図
- 【図 10】プレイヤーが第 1 コントローラ 7 を左から右へ振った場合に液晶テレビ 2 に表示されるアニメーションの一例を示す画面図
- 【図 11】プレイヤーが第 1 コントローラ 7 を右から左へ振った場合に液晶テレビ 2 に表示されるアニメーションの一例を示す画面図
- 【図 12】外部メインメモリ 12 のメモリマップを例示する図
- 【図 13】ゲーム装置 3 で実行されるメイン処理の一例を示すフローチャート
- 【図 14】プレイヤーによる第 1 コントローラ 7 の振り操作について説明するための説明図
- 【図 15】図 13 のステップ S 9 における構え処理の詳細フローチャート 10
- 【図 16】図 13 のステップ S 10 における攻撃開始処理の詳細フローチャート
- 【図 17】図 13 のステップ S 12 における当たり判定処理の詳細フローチャート
- 【図 18】図 13 のステップ S 12 における当たり判定処理の詳細フローチャート
- 【図 19】プレイヤーが第 1 コントローラ 7 を左から右へ振った場合に液晶テレビ 2 に表示されるアニメーションの一例を示す画面図
- 【図 20】プレイヤーが第 1 コントローラ 7 を右から左へ振った場合に液晶テレビ 2 に表示されるアニメーションの一例を示す画面図
- 【図 21】プレイヤーが第 1 コントローラ 7 を右から左へ振った場合に液晶テレビ 2 に表示されるアニメーションの一例を示す画面図
- 【図 22】プレイヤーが第 1 コントローラ 7 を右から左へ振った場合に液晶テレビ 2 に表示されるアニメーションの一例を示す画面図 20
- 【図 23】当たり判定処理の変形例を示すフローチャート
- 【図 24】プレイヤーが第 1 コントローラ 7 を右から左へ振った場合に液晶テレビ 2 に表示されるアニメーションの一例を示す画面図
- 【図 25】プレイヤーが第 1 コントローラ 7 を左から右へ振った場合に液晶テレビ 2 に表示されるアニメーションの一例を示す画面図
- 【図 26】プレイヤーが第 1 コントローラ 7 を右から左へ振った場合に液晶テレビ 2 に表示されるアニメーションの一例を示す画面図
- 【発明を実施するための形態】
- 【0025】 30
- 以下、適宜図面を参照しつつ、本発明の一実施形態について説明する。図 1 は、本発明の一実施形態に係るゲーム装置 3 を含むゲームシステム 1 の外観斜視図である。本実施形態では、本発明のゲーム装置が据置型のゲーム装置 3 である場合を例にして、本発明について説明するが、本発明のゲーム装置は、例えばゲーム装置と表示装置とが一体的に構成された携帯型のゲーム装置であってもよい。
- 【0026】
- [ゲームシステム 1 の全体構成]
- まず、ゲームシステム 1 の各構成の概略を説明する。図 1 において、ゲームシステム 1 は、液晶テレビ 2、ゲーム装置 3、光ディスク 4、マーカ部 6、及びコントローラ 5 を含む。ゲームシステム 1 は、コントローラ 5 を用いたプレイヤーのゲーム操作に基づいてゲーム装置 3 でゲーム処理を実行するものである。 40
- 【0027】
- ゲーム装置 3 には、ゲーム装置 3 に対して交換可能に用いられる情報記憶媒体の一例である光ディスク 4 が着脱可能に挿入される。光ディスク 4 には、ゲーム装置 3 において実行されるゲームプログラムが記憶されている。ゲーム装置 3 の前面には光ディスク 4 の挿入口が設けられている。ゲーム装置 3 は、挿入口に挿入された光ディスク 4 からゲームプログラムを読み出して実行することによってゲーム処理を実行する。
- 【0028】
- ゲーム装置 3 には、液晶テレビ 2 が接続コードを介して接続される。後述するが、ゲーム装置 3 は、仮想ゲーム空間に配置された仮想カメラの視点から見たオブジェクトを含む 50

仮想ゲーム空間の画像（以下「ゲーム画像」という。）を生成して液晶テレビ2へ出力する。この一連の処理は、本実施形態では、フレーム単位（例えば1/60秒間隔）で行われる。液晶テレビ2は、このようにしてゲーム装置3から出力されるゲーム画像を受信して画面に表示する。

【0029】

液晶テレビ2の画面の周辺（図1では画面の上側）には、マーカ部6が設置される。マーカ部6は、その両端に2つのマーカ6R及び6Lを備えている。マーカ6R（マーカ6Lも同様）は、具体的には1個以上の赤外LEDであって、液晶テレビ2の前方に向かって赤外光を出力する。マーカ部6はゲーム装置3に接続されており、ゲーム装置3はマーカ部6が備える各赤外LEDの点灯を制御することが可能である。また、マーカ部6には、不図示のマイクロフォンが配設され、該マイクロフォンを介して入力された音声情報は、ゲーム装置3に入力される。

10

【0030】

コントローラ5は、プレイヤーによって操作される入力手段である。コントローラ5は、本実施形態では、それぞれプレイヤーが片手で把持可能な第1コントローラ7及び第2コントローラ8と、第1コントローラ7に着脱可能に装着されるジャイロセンサユニット9とから構成されている。第1コントローラ7は、プレイヤーによって操作される入力装置であって、第1コントローラ7に対して行われた操作の内容を示す操作データを生成する。ジャイロセンサユニット9は、この第1コントローラ7に装着された状態で、第1コントローラ7の位置及び姿勢を検出する状態検出手段として機能する。ジャイロセンサユニット9は、第1コントローラ7の角速度を検出して、検出された角速度を示す角速度データを第1コントローラ7へ出力する。ここで、角速度データは、第1コントローラ7の位置及び姿勢を特定するための第1コントローラ7の状態を示す状態信号である。第2コントローラ8は、第2コントローラ8に対して行われた操作の内容を示す操作データを生成する。この操作データは、ジャイロセンサユニット9を介して第1コントローラ7へ出力される。第1コントローラ7は、第1コントローラ7の操作データ、第2コントローラ8の操作データ、ジャイロセンサユニット9からの角速度データ等を含むコントローラデータをゲーム装置3へ送信する。

20

【0031】

コントローラデータを送信するために、第1コントローラ7とゲーム装置3とは無線通信によって接続される。本実施形態では、第1コントローラ7とゲーム装置3との間の無線通信にBluetooth（ブルートゥース）（登録商標）の技術が用いられる。なお、他の実施形態においては、第1コントローラ7とゲーム装置3とは有線で接続されてもよい。

30

【0032】

[第1コントローラ7の外部構成]

図2は、第1コントローラ7及びジャイロセンサユニット9を上面後方から見た斜視図である。図3は、第1コントローラ7を下面前方から見た斜視図である。

【0033】

図2及び図3に示されるように、第1コントローラ7は、例えばプラスチック成型によって形成されたハウジング71を有している。ハウジング71は、その前後方向（図2、3におけるZ軸方向）を長手方向とした略直方体形状を有しており、全体として大人や子供が片手で把持可能な大きさに構成されている。このハウジング71には、操作部72が配置されている。

40

【0034】

ハウジング71の上面に配置された操作部72は、十字キー72a、操作ボタン72b～72g、及び操作ボタン72hから構成されている。

【0035】

十字キー72aは、十字型の4方向プッシュスイッチであり、4つの方向（前後左右）に対応する操作部分が十字の突出片にそれぞれ90°間隔で配置される。プレイヤーが十字

50

キー 7 2 a のいずれかの操作部分を押下することによって、前後左右いずれかの方向が選択される。プレイヤーが十字キー 7 2 a を操作することによって、例えば複数の選択肢からいずれかの選択肢を選択することができる。

【 0 0 3 6 】

操作ボタン 7 2 b ~ 7 2 g は、プレイヤーがボタン頭部を押下することによって、それぞれの操作ボタン 7 2 b ~ 7 2 g に割り当てられた操作信号を出力する。例えば、操作ボタン 7 2 b ~ 7 2 d には、1 番ボタン、2 番ボタン、及び A ボタン等としての機能が割り当てられる。また、操作ボタン 7 2 e ~ 7 2 g には、マイナスボタン、ホームボタン、及びプラスボタン等としての機能が割り当てられる。これらの操作ボタン 7 2 a ~ 7 2 g は、ゲーム装置 3 が実行するゲームプログラムに応じてそれぞれの操作機能が割り当てられる。なお、操作ボタン 7 2 f は、その上面がハウジング 7 1 の上面に埋没しており、プレイヤーが不意に誤って押下することのないタイプの操作ボタンである。

10

【 0 0 3 7 】

操作ボタン 7 2 h は、遠隔からゲーム装置 3 の電源をオン / オフする電源スイッチである。この操作ボタン 7 2 h は、ハウジング 7 1 の上面に埋没しており、操作ボタン 7 2 f と同じタイプの操作ボタンである。

【 0 0 3 8 】

ハウジング 7 1 の上面には、操作部 7 2 の他に複数の LED 7 0 2 が設けられる。ここで、第 1 コントローラ 7 は、他の第 1 コントローラ 7 と区別するためにコントローラ種別 (番号) が設けられている。例えば、LED 7 0 2 は、第 1 コントローラ 7 に現在設定されている上記コントローラ種別をプレイヤーに通知するために用いられる。具体的には、第 1 コントローラ 7 から無線コントローラモジュール 1 9 (図 5 参照) へコントローラデータを送信する際、上記コントローラ種別に応じて複数の LED 7 0 2 のうち、種別に対応する LED が点灯する。

20

【 0 0 3 9 】

また、ハウジング 7 1 の上面には、操作ボタン 7 2 b 及び操作ボタン 7 2 e ~ 7 2 g の間に、ハウジング 7 1 に内蔵されたスピーカ 7 0 6 (図 6 参照) からの音を外部に放出するための音抜き孔 7 1 1 が形成されている。

【 0 0 4 0 】

図 3 に示されているように、ハウジング 7 1 の下面には、凹部が形成されている。ハウジング 7 1 の下面の凹部は、プレイヤーが第 1 コントローラ 7 の前面をマーク 6 L 及び 6 R に向けて片手で把持したときに、当該プレイヤーの人差し指や中指が位置するような位置に形成される。そして、上記凹部の後面側傾斜面には、操作ボタン 7 2 i が設けられる。操作ボタン 7 2 i は、プレイヤーがボタン頭部を押下することによって、操作ボタン 7 2 i に割り当てられた操作信号を出力するものであり、例えば B ボタンとして機能する。

30

【 0 0 4 1 】

ハウジング 7 1 の前面には、撮像情報演算部 7 4 (図 3 参照) の一部を構成する撮像素子 7 4 3 (図 6 参照) が設けられる。ここで、撮像情報演算部 7 4 は、第 1 コントローラ 7 が撮像した画像データを解析してその中で輝度が高い場所を判別してその場所の重心位置やサイズなどを検出するためのシステムであり、例えば、最大 2 0 0 フレーム / 秒程度のサンプリング周期であるため比較的高速な第 1 コントローラ 7 の動きでも追跡して解析することができる。この撮像情報演算部 7 4 の詳細な構成については、後述する。また、ハウジング 7 1 の後面には、コネクタ 7 3 (図 2 参照) が設けられている。コネクタ 7 3 は、例えばエッジコネクタである。

40

【 0 0 4 2 】

[ジャイロセンサユニット 9 の外部構成]

図 2 には現れていないが、ジャイロセンサユニット 9 の先端面には、第 1 コントローラ 7 のコネクタ 7 3 と接続可能なコネクタ 9 1 (図 6 参照) が設けられている。コネクタ 9 1 がコネクタ 7 3 に接続されることで、ジャイロセンサユニット 9 が第 1 コントローラ 7 と物理的及び電氣的に接続される。ジャイロセンサユニット 9 は、このようにして第 1 コ

50

ントローラ 7 に一体化された状態で、第 1 コントローラ 7 の 3 軸 (X 、 Y 、 Z 軸) 周りの角速度を検出する。第 1 コントローラ 7 の角速度が検出されると、その検出結果を示す角速度データがジャイロセンサユニット 9 から第 1 コントローラ 7 へ出力される。

【 0 0 4 3 】

ジャイロセンサユニット 9 の後端面には、第 2 コントローラ 8 のコネクタ 8 2 (図 1 参照) と接続可能なコネクタ 9 2 (図 6 参照) が設けられている。コネクタ 8 2 がコネクタ 9 2 に接続されることで、第 2 コントローラ 8 がジャイロセンサユニット 9 を介して第 1 コントローラ 7 と接続される。なお、図 2 では、コネクタ 9 2 にコネクタカバー 9 3 が取り付けられた状態が表記されており、コネクタ 9 2 は現れていない。

【 0 0 4 4 】

ジャイロセンサユニット 9 の側面には、リリースボタン 9 4 が設けられている。図には示されていないが、ジャイロセンサユニット 9 には、ジャイロセンサユニット 9 の先端面から突出するフックが設けられている。ジャイロセンサユニット 9 が第 1 コントローラ 7 に装着される際に、このフックがハウジング 7 1 の後面に形成された孔 7 6 (図 2 参照) を介してハウジング 7 1 内に挿入されて、ハウジング 7 1 の内壁と係合する。これにより、ジャイロセンサユニット 9 が第 1 コントローラ 7 に固定される。リリースボタン 9 4 は、このフックと連結されている。プレイヤーは、リリースボタン 9 4 を押下したままフックをハウジング 7 1 から引き抜くことで、ジャイロセンサユニット 9 を第 1 コントローラ 7 から取り外すことができる。

【 0 0 4 5 】

[第 2 コントローラ 8 の外部構成]

第 2 コントローラ 8 のコネクタ 8 2 (図 1 参照) は、第 1 コントローラ 7 のコネクタ 7 3 (図 2 及び図 6 参照) 又はジャイロセンサユニット 9 のコネクタ 9 2 (図 6 参照) と接続可能である。コネクタ 8 2 がコネクタ 7 3 に接続された場合、第 2 コントローラ 8 の操作データがケーブル 8 1 及びコネクタ 8 2 を介して第 1 コントローラ 7 に直接入力される。一方、コネクタ 9 1 がコネクタ 7 3 に接続され、且つコネクタ 8 2 がコネクタ 9 2 に接続された場合、第 2 コントローラ 8 の操作データは、ケーブル 8 1、コネクタ 8 2、及びジャイロセンサユニット 9 を介して第 1 コントローラ 7 に入力される。

【 0 0 4 6 】

図 4 は、第 2 コントローラ 8 の斜視図であり、図 4 (A) は第 2 コントローラ 8 を上面後方から見た斜視図であり、図 4 (B) は第 2 コントローラ 8 を下面前方から見た斜視図である。なお、図 4 では、第 2 コントローラ 8 のケーブル 8 1 及びコネクタ 8 2 の図示が省略されている。

【 0 0 4 7 】

第 2 コントローラ 8 は、例えばプラスチック成型によって形成されたハウジング 8 3 を有している。ハウジング 8 3 は、その前後方向 (図 4 における Z 軸方向) に細長い楕円形状を有し、後端側の左右方向 (X 軸方向) の幅が先端側の左右方向の幅よりも狭くなっている。また、ハウジング 8 3 は、側面視では、全体として湾曲した形状を有しており、先端側の水平部分から後端側に向かって下がるように湾曲している。ハウジング 8 3 は、第 1 コントローラ 7 のハウジング 7 1 と同様に、全体として大人や子供が片手で把持可能な大きさに構成されているが、長手方向 (Z 軸方向) の長さは、ハウジング 7 1 よりも若干短く設定されている。このハウジング 8 3 には、操作部 8 4 が配置されている。

【 0 0 4 8 】

ハウジング 8 3 の上面の先端側には、操作部 8 4 としてのアナログジョイスティック 8 4 a が設けられている。また、ハウジング 8 3 の先端面には、C ボタン 8 4 b 及び Z ボタン 8 4 c が設けられている。アナログジョイスティック 8 4 a、C ボタン 8 4 b、及び Z ボタン 8 4 c は、ゲーム装置 3 が実行するゲームプログラムに応じてそれぞれの操作機能が割り当てられる。プレイヤーは、アナログジョイスティック 8 4 a を傾けたりボタン 8 4 b、8 4 c を押下したりすることによって、ゲーム操作を行うことができる。

【 0 0 4 9 】

[ゲーム装置 3 の内部構成]

次に、図 5 を参照しつつ、ゲーム装置 3 の内部構成について説明する。図 5 は、ゲーム装置 3 の構成を示すブロック図である。ゲーム装置 3 は、CPU 10、システム LSI 11、外部メインメモリ 12、ROM/RTC 13、ディスクドライブ 14、及び AV-IC 15 等を有する。

【 0050 】

CPU 10 は、光ディスク 4 に格納されたゲームプログラムを実行することによってゲーム処理を実行するものであり、ゲームプロセッサとして機能する。CPU 10 は、システム LSI 11 に接続される。システム LSI 11 には、CPU 10 の他、外部メインメモリ 12、ROM/RTC 13、ディスクドライブ 14、及び AV-IC 15 が接続される。システム LSI 11 は、それに接続される各構成要素間のデータ転送の制御、表示すべき画像の生成、外部装置からのデータの取得等の処理を行う。システム LSI 11 の内部構成については後述する。

10

【 0051 】

外部メインメモリ 12 は、揮発性メモリである。外部メインメモリ 12 は、光ディスク 4 から読み出されたゲームプログラム、フラッシュメモリ 17 から読み出されたゲームプログラム等のプログラムを格納したり、各種データを格納したりするものであり、CPU 10 のワーク領域、バッファ領域等として用いられる。

【 0052 】

ROM/RTC 13 は、ゲーム装置 3 の起動用のプログラムが組み込まれる ROM (いわゆる、ブート ROM) と、時間をカウントするクロック回路 (RTC: Real Time Clock) とを有する。

20

【 0053 】

ディスクドライブ 14 は、光ディスク 4 からプログラムデータ、テクスチャデータ等を読み出し、読み出したデータを後述する内部メインメモリ 11e 又は外部メインメモリ 12 に書き込む。

【 0054 】

システム LSI 11 には、入出力プロセッサ (I/O プロセッサ) 11a、GPU (Graphics Processor Unit) 11b、DSP (Digital Signal Processor) 11c、VRAM 11d、及び内部メインメモリ 11e が設けられる。図示は省略するが、これらの構成要素 11a ~ 11e は内部バスによって互いに接続される。

30

【 0055 】

GPU 11b は、描画手段の一部を形成し、CPU 10 からのグラフィクスコマンド (作画命令) に従って画像を生成する。VRAM 11d は、GPU 11b がグラフィクスコマンドを実行するために必要なデータ (ポリゴンデータ、テクスチャデータ等のデータ) を記憶する。画像が生成される際には、GPU 11b は、VRAM 11d に記憶されたデータを用いて画像データを作成する。

【 0056 】

DSP 11c は、オーディオプロセッサとして機能し、内部メインメモリ 11e、外部メインメモリ 12 等に格納されるサウンドデータ、音波形 (音色) データ等を用いて、音声データを生成する。

40

【 0057 】

上述のように生成された画像データ及び音声データは、AV-IC 15 によって読み出される。AV-IC 15 は、読み出した画像データを、AV コネクタ 16 を介して液晶テレビ 2 に出力するとともに、読み出した音声データを、液晶テレビ 2 に内蔵されるスピーカ 2a に出力する。これによって、画像が液晶テレビ 2 に表示されるとともに音がスピーカ 2a から出力される。

【 0058 】

入出力プロセッサ 11a は、それに接続される構成要素との間でデータの送受信を実行

50

したり、外部装置からのデータのダウンロードを実行したりする。入出力プロセッサ 11 a は、フラッシュメモリ 17、無線通信モジュール 18、無線コントローラモジュール 19、拡張コネクタ 20、及びメモリカード用コネクタ 21 に接続される。無線通信モジュール 18 にはアンテナ 22 が接続され、無線コントローラモジュール 19 にはアンテナ 23 が接続される。

【 0059 】

入出力プロセッサ 11 a は、無線通信モジュール 18 及びアンテナ 22 を介してネットワークに接続し、ネットワークに接続される他のゲーム装置や各種サーバと通信することができる。入出力プロセッサ 11 a は、定期的にフラッシュメモリ 17 にアクセスし、ネットワークへ送信する必要があるデータの有無を検出し、当該データがある場合には、無線通信モジュール 18 及びアンテナ 22 を介してネットワークに送信する。また、入出力プロセッサ 11 a は、他のゲーム装置から送信されてくるデータやダウンロードサーバからダウンロードしたデータを、ネットワーク、アンテナ 22 及び無線通信モジュール 18 を介して受信し、受信したデータをフラッシュメモリ 17 に格納する。CPU 10 は、ゲームプログラムを実行することにより、フラッシュメモリ 17 に格納されたデータを読み出してゲームプログラムで利用する。フラッシュメモリ 17 には、ゲーム装置 3 と他のゲーム装置や各種サーバとの間で送受信されるデータの他、ゲーム装置 3 を利用してプレイしたゲームのセーブデータ（ゲームの結果データ又は途中データ）が格納されてもよい。

【 0060 】

また、入出力プロセッサ 11 a は、コントローラ 7 から送信されるコントローラデータをアンテナ 23 及び無線コントローラモジュール 19 を介して受信し、内部メインメモリ 11 e 又は外部メインメモリ 12 のバッファ領域に格納（一時記憶）する。

【 0061 】

更に、入出力プロセッサ 11 a には、拡張コネクタ 20 及びメモリカード用コネクタ 21 が接続される。拡張コネクタ 20 は、USB や SCSI のようなインターフェースのためのコネクタであり、外部記憶媒体のようなメディアを接続したり、他のコントローラのような周辺機器を接続したり、有線の通信用コネクタを接続することによって無線通信モジュール 18 に換えてネットワークとの通信を行ったりすることができる。メモリカード用コネクタ 21 は、メモリカードのような外部記憶媒体を接続するためのコネクタである。例えば、入出力プロセッサ 11 a は、拡張コネクタ 20 やメモリカード用コネクタ 21 を介して、外部記憶媒体にアクセスし、データを保存したり、データを読み出したりすることができる。

【 0062 】

ゲーム装置 3 には、電源ボタン 24、リセットボタン 25、及びイジェクトボタン 26 が設けられる。電源ボタン 24 及びリセットボタン 25 は、システム LSI 11 に接続される。電源ボタン 24 が押下されると、ゲーム装置 3 の各構成要素に対して、図示しない AC アダプタを経て電源が供給される。リセットボタン 25 が押下されると、システム LSI 11 は、ゲーム装置 3 の起動プログラムを再起動する。イジェクトボタン 26 は、ディスクドライブ 14 に接続される。イジェクトボタン 26 が押下されると、ディスクドライブ 14 から光ディスク 4 が排出される。

【 0063 】

[コントローラ 5 の内部構成]

次に、図 6 を参照しつつ、コントローラ 5 の内部構成について説明する。図 6 は、コントローラ 5 の内部構成を示すブロック図である。なお、図 6 には、第 1 コントローラ 7 にジャイロセンサユニット 9 が接続され、更にジャイロセンサユニット 9 に第 2 コントローラ 8 が接続された状態が示されている。

【 0064 】

図 6 に示されるように、第 1 コントローラ 7 は、上述した操作部 72 の他に、撮像情報演算部 74、加速度センサ 701、バイプレータ 704、スピーカ 706、サウンド IC 707、アンプ 708、及び通信部 75 を備えている。

【 0 0 6 5 】

撮像情報演算部 7 4 は、赤外線フィルタ 7 4 1、レンズ 7 4 2、撮像素子 7 4 3、及び画像処理回路 7 4 4 を含んでいる。赤外線フィルタ 7 4 1 は、第 1 コントローラ 7 の前方から入射する光から赤外線のみを通過させる。レンズ 7 4 2 は、赤外線フィルタ 7 4 1 を透過した赤外線を集光して撮像素子 7 4 3 へ入射させる。撮像素子 7 4 3 は、例えば C M O S センサやあるいは C C D センサのような固体撮像素子であり、レンズ 7 4 2 が集光した赤外線を受光して画像信号を出力する。ところで、液晶テレビ 2 に固定されたマーカ部 6 のマーカ 6 R 及び 6 L からは、液晶テレビ 2 の前方に向かって赤外光が出射されている。このため、赤外線フィルタ 7 4 1 を設けることによって、撮像素子 7 4 3 は、赤外線フィルタ 7 4 1 を通過した赤外線だけを受光して画像データを生成するので、マーカ 6 R 及び 6 L の画像を正確に撮像することができる。以下では、撮像素子 7 4 3 によって撮像された画像を撮像画像と呼ぶ。撮像素子 7 4 3 によって生成された画像データは、画像処理回路 7 4 4 で処理される。画像処理回路 7 4 4 は、撮像画像内における撮像対象（マーカ 6 R 及び 6 L ）の位置を算出する。画像処理回路 7 4 4 は、算出された位置を示す座標を通信部 7 5 のマイコン 7 5 1 へ出力する。この座標のデータは、マイコン 7 5 1 によって操作データとしてゲーム装置 3 に送信される。以下では、上記座標を「マーカ座標」と呼ぶ。マーカ座標は第 1 コントローラ 7 自体の向き（傾斜角度）や位置に対応して変化するので、ゲーム装置 3 は、このマーカ座標を用いて第 1 コントローラ 7 の向きや位置を算出することができる。

10

【 0 0 6 6 】

加速度センサ 7 0 1 は、第 1 コントローラ 7 の位置及び姿勢を検出するものである。この加速度センサ 7 0 1 は、3 方向、すなわち、上下方向（図 2 に示す Y 軸）、左右方向（図 2 に示す X 軸）、及び前後方向（図 2 に示す Z 軸）の 3 軸方向のそれぞれに関して直線加速度を検出する。加速度センサ 7 0 1 によって検出された加速度を示す状態信号（加速度データ）は、通信部 7 5 へ出力され、通信部 7 5 によってゲーム装置 3 へ送信される。加速度センサ 7 0 1 が検出した加速度は、第 1 コントローラ 7 の向き（傾斜角度）や動きに応じて変化するので、ゲーム装置 3 は、加速度データに基づいて第 1 コントローラ 7 の位置及び姿勢を算出することが可能である。本実施形態では、ゲーム装置 3 は、後述する角速度データに基づいて第 1 コントローラ 7 の位置及び姿勢を算出するが、加速度データに基づいて第 1 コントローラ 7 の位置及び姿勢を算出することも可能である。

20

30

【 0 0 6 7 】

通信部 7 5 は、マイクロコンピュータ（Micro Computer：マイコン）7 5 1、メモリ 7 5 2、無線モジュール 7 5 3、及びアンテナ 7 5 4 を含んでいる。マイコン 7 5 1 は、処理の際にメモリ 7 5 2 を記憶領域として用いながら、送信データを無線送信する無線モジュール 7 5 3 を制御する。また、マイコン 7 5 1 は、アンテナ 7 5 4 を介して無線モジュール 7 5 3 が受信したゲーム装置 3 からのデータに応じて、サウンド IC 7 0 7 及びバイブレータ 7 0 4 の動作を制御する。無線モジュール 7 5 3 及びアンテナ 7 5 4 を備えていることにより、第 1 コントローラ 7 がワイヤレスコントローラとして機能する。

【 0 0 6 8 】

サウンド IC 7 0 7 は、通信部 7 5 を介してゲーム装置 3 から送信されたサウンドデータ等を処理する。このサウンドデータがアンプ 7 0 8 で増幅されてスピーカ 7 0 6 へ送られることで、スピーカ 7 0 6 から音声が出力される。バイブレータ 7 0 4 は、例えば振動モータやソレノイドである。バイブレータ 7 0 4 は、通信部 7 5 を介してゲーム装置 3 から送信された振動データ（例えば、バイブレータ 7 0 4 を ON 又は OFF する信号）等に応じて作動する。バイブレータ 7 0 4 が作動することによって第 1 コントローラ 7 に振動が発生するので、それを把持しているプレイヤーの手にその振動が伝達され、いわゆる振動対応ゲームが実現できる。

40

【 0 0 6 9 】

第 2 コントローラ 8 は、図 6 に示されるように、操作部 8 4 及び加速度センサ 8 5 を有

50

する。加速度センサ 85 としては、例えば 3 軸加速度センサが適用され、第 2 コントローラ 8 の上下方向（図 4 に示す y 軸方向）、左右方向（図 4 に示す x 軸方向）、及び前後方向（図 4 に示す z 軸方向）の 3 軸方向のそれぞれの方向に関して直線加速度を検出する。加速度センサ 85 が検出した加速度を示す加速度データは、第 1 コントローラ 7 へ送られ、通信部 75 によってゲーム装置 3 へ送信される。加速度センサ 85 が検出した加速度は、第 2 コントローラ 8 の向き（傾斜角度）や動きに応じて変化するので、ゲーム装置 3 は、第 1 コントローラ 7 と同様に、加速度センサ 85 からの加速度データに基づいて、第 2 コントローラ 8 の位置及び姿勢を算出することが可能である。

【0070】

操作部 84 は、上述したアナログジョイスティック 84a、C ボタン 84b、及び Z ボタン 84c から構成されている。操作部 84 が操作されると、その操作内容を示す操作データが生成される。操作データは、ケーブル 81、コネクタ 82、及びコネクタ 92 を介してジャイロセンサユニット 9 へ送られる。また、加速度センサ 85 によって検出された第 2 コントローラ 8 の加速度を示す加速度データも、同様にジャイロセンサユニット 9 へ送られる。これらの操作データ及び加速度データは、ジャイロセンサ 9 によって第 1 コントローラ 7 の通信部 75 へ送られる。

【0071】

ジャイロセンサユニット 9 は、3 軸（本実施形態では X Y Z 軸）周りの角速度を検出し、検出した角速度を示す角速度データを第 1 コントローラ 7 へ出力するものである。このジャイロセンサユニット 9 は、図 6 に示されるように、その内部にマイコン 95、2 軸ジャイロセンサ 96（角速度センサ）、及び 1 軸ジャイロセンサ（角速度センサ）97 を有している。

【0072】

図 7 は、ジャイロセンサユニット 9 によって角速度が検出されるロール角、ピッチ角、及びヨー角について説明するための第 1 コントローラ 7 及びジャイロセンサユニット 9 の斜視図である。2 軸ジャイロセンサ 96 は、ロール角及びピッチ角に関する 2 つの（単位時間当たりの）角速度（Z 軸周りの角速度及び X 軸周りの角速度）を検出する。1 軸ジャイロセンサ 97 は、ヨー角に関する（単位時間当たりの）角速度（Y 軸周りの角速度）を検出する。なお、本明細書では、図 7 に示されるように、第 1 コントローラ 7 の撮像方向である Z 軸正方向を基準として、Z 軸周り、X 軸周り、Y 軸周りの回転方向を、それぞれ、ロール方向、ピッチ方向、ヨー方向と呼ぶ。

【0073】

2 軸ジャイロセンサ 96 及び 1 軸ジャイロセンサ 97 で検出された角速度を示すデータは、マイコン 95 へ出力される。したがって、マイコン 95 には、X Y Z 軸の 3 軸周りの角速度を示すデータが入力されることになる。マイコン 95 は、上記 3 軸周りの角速度を示すデータを角速度データとしてコネクタ 91 を介して第 1 コントローラ 7 へ出力する。なお、マイコン 95 から第 1 コントローラ 7 へのデータ送信は所定の周期毎に逐次行われるが、ゲームの処理は 1 / 60 秒を単位として（1 フレーム時間として）行われることが一般的であるので、この時間以下の周期で送信を行うことが好ましい。後述するが、角速度データは、第 1 コントローラ 7 からゲーム装置 3 へ送信される。ジャイロセンサユニット 9 によって検出される第 1 コントローラ 7 の角速度は、第 1 コントローラ 7 の向き（傾斜角度）や動きに応じて変化するので、ゲーム装置 3 は、角速度データに基づいて、第 1 コントローラ 7 の位置及び姿勢を正確に算出することが可能である。

【0074】

第 1 コントローラ 7 の説明に戻り、メモリ 752 には、操作部 72 からの操作データ、撮像情報演算部 74 からのマーカ座標、加速度センサ 701 からの加速度データが一時的に格納される。また、第 1 コントローラ 7 にジャイロセンサユニット 9 が装着されているので、メモリ 752 には、2 軸ジャイロセンサ 96 及び 1 軸ジャイロセンサ 97 からの角速度データ（第 1 コントローラ 7 の状態信号）が一時的に格納される。さらに、第 1 コントローラ 7 にジャイロセンサユニット 9 を介して第 2 コントローラ 8 が接続されているの

10

20

30

40

50

で、メモリ752には、操作部84からの操作データ、及び加速度センサ85からの加速度データが一時的に格納される。マイコン751は、ゲーム装置3の無線コントローラモジュール19(図5参照)への送信タイミングが到来すると、メモリ752に格納されているデータを上記コントローラデータとして無線モジュール753へ出力する。無線モジュール753は、例えばBluetooth(ブルートゥース)の技術を用いて、所定周波数の搬送波をコントローラデータで変調し、その微弱電波信号をアンテナ754から放射する。つまり、コントローラデータは、無線モジュール753で微弱電波信号に変調されて第1コントローラ7から送信される。微弱電波信号はゲーム装置3側の無線コントローラモジュール19で受信される。受信された微弱電波信号について復調や復号を行うことによって、ゲーム装置3はコントローラデータを取得することができる。そして、ゲーム装置3のCPU10は、取得したコントローラデータとゲームプログラムとに基づいて、ゲーム処理を行う。

10

【0075】

なお、通信部75から無線コントローラモジュール19への無線送信は所定の周期毎に逐次行われるが、ゲームの処理は1/60秒を単位として(1フレーム時間として)行われることが一般的であるので、この時間以下の周期で送信を行うことが好ましい。第1コントローラ7の通信部75は、例えば1/200秒に1回の割合で各コントローラデータをゲーム装置3の無線コントローラモジュール19へ送信する。

【0076】

以上説明したコントローラ5を用いることによって、各操作ボタンを押下する従来のボタン操作だけでなく、コントローラ5自体を動かすことによっても、ゲームなどのアプリケーションに対する入力を行うことができる。ゲームをプレイするに際して、プレイヤーは、例えば図8に示されるように、右手で第1コントローラ7を持ち、左手で第2コントローラ8を持つ。上述のように、第1コントローラ7には、加速度センサ701が内蔵されると共にジャイロセンサユニット9が固定されている。また、第2コントローラ8には、加速度センサ85が内蔵されている。第1コントローラ7がプレイヤーによって動かされると、加速度センサ701によって第1コントローラ7の3軸方向の加速度が検出されると共に、ジャイロセンサユニット9の2軸ジャイロセンサ96及び1軸ジャイロセンサ97によって第1コントローラ7の3軸周りの角速度が検出される。一方、第2コントローラ8がプレイヤーによって動かされると、加速度センサ85によって第2コントローラ8の3軸方向の加速度が検出される。これらの検出結果を示すデータは、上述のようにコントローラデータとしてゲーム装置3へ送信されて、ゲーム処理に反映される。このため、プレイヤーは、第1コントローラ7や第2コントローラ8を振り動かすといったゲーム操作を行うことができる。

20

30

【0077】

なお、上述したハードウェア構成は単なる一例に過ぎず、ゲーム装置3やコントローラ5の構成は適宜変更可能である。

【0078】

[ゲームの概要]

次に、図9~図11を参照しつつ、ゲーム装置3のCPU10がゲームプログラムを実行することによって進行するゲームの概要について説明する。

40

【0079】

本実施形態において実行されるゲームでは、仮想ゲーム空間(仮想3次元空間)内に配置されたプレイヤーオブジェクト101及び剣オブジェクト104を操作して敵を倒すアクションアドベンチャーゲームである。図9は、液晶テレビ2に表示されるゲーム画像の一例を示す画面図である。図9に例示されるように、液晶テレビ2の画面には、プレイヤーによって操作されるプレイヤーオブジェクト101が敵オブジェクト102に遭遇した様子が表示されている。プレイヤーは、第2コントローラ8のアナログジョイスティック84a(図8参照)を操作することによってプレイヤーオブジェクト101を移動させることができる。

50

【 0 0 8 0 】

例えば、プレイヤーオブジェクト101が敵オブジェクト102を攻撃可能な位置まで移動すると、図には示されていないが、プレイヤーオブジェクト101が背負っている剣オブジェクト104（第1オブジェクトの一例）を鞘から抜いて剣オブジェクト104を構える様子が液晶テレビ2の画面に表示される。その際、プレイヤーオブジェクト101が攻撃可能な敵オブジェクト102が複数存在する場合、いずれかの敵オブジェクト102と重なるようにカーソル103が表示される。カーソル103は、プレイヤーオブジェクト101の攻撃対象となる敵オブジェクトを切り替えるためのものである。例えば第2コントローラ8のZボタン84cが押下されると、カーソル103は、ボタン操作前に重なるように表示されていた敵オブジェクト102とは異なる敵オブジェクト102と重なるように表示される。したがって、攻撃可能な敵オブジェクト102が複数存在する場合、プレイヤーは、Zボタン84cを押下することによって、攻撃対象となる敵オブジェクトを切り替えることができる。なお、このようにしていずれかの敵オブジェクトが選択されたとしても、剣オブジェクト104が振られた際に複数の敵オブジェクトと衝突するような場合には、これら複数の敵オブジェクトに対して攻撃を加えることが可能である。

10

【 0 0 8 1 】

プレイヤーは、第1コントローラ7を剣に見立てて、第1コントローラ7を構えることによって、プレイヤーオブジェクト101に剣オブジェクト104を構えさせることができる。また、プレイヤーは、第1コントローラ7を構えた状態から任意の振り方向（動作方向）へ第1コントローラ7を振る動作（ゲーム操作）を行うことで、プレイヤーオブジェクト101に剣オブジェクト104で敵オブジェクト102を斬りつける攻撃動作を行わせることができる。なお、第1コントローラ7の位置及び姿勢がジャイロセンサユニット9によって検出されるので、プレイヤーが第1コントローラ7を振るのと同じような方向に、プレイヤーオブジェクト101に剣オブジェクト104を振らせることができる。

20

【 0 0 8 2 】

図10は、プレイヤーが第1コントローラ7を左から右へ振った場合に液晶テレビ2に表示されるアニメーションの一例を示す画面図である。図11は、プレイヤーが第1コントローラ7を右から左へ振った場合に液晶テレビ2に表示されるアニメーションの一例を示す画面図である。プレイヤーが第1コントローラ7を左から右へ振った場合、図10に例示されるように、剣オブジェクト104が敵オブジェクト102に命中（ヒット）して敵オブジェクト102にダメージが与えられる様子を示すアニメーション（モーションブラー効果を示す画像）が表示される。一方、プレイヤーが第1コントローラ7を右から左へ振った場合、図11に例示されるように、剣オブジェクト104が敵オブジェクト102に命中（ヒット）せずに弾き返されるアニメーションが表示される。このように、本発明は、第1コントローラ7の振り方向（剣オブジェクト104の振り方向）に応じて、剣オブジェクト104が敵オブジェクト（本実施形態では敵オブジェクト102）に衝突した場合の敵オブジェクトの反応を示す演出を容易に切り替えることができることを主な特徴としている。以下、第1コントローラ7の振り方向に応じて、敵オブジェクト102の反応を示す演出を切り替えるためのゲーム装置3の構成、及びゲーム装置3で行われるゲーム処理について詳細に説明する。

30

40

【 0 0 8 3 】

[主要データ]

以下、図12を参照しつつ、ゲーム処理の際に外部メインメモリ12に記憶されるデータについて説明する。ここで、図12は、外部メインメモリ12のメモリマップを例示する図である。図12に例示されるように、外部メインメモリ12は、プログラム記憶領域121及びデータ記憶領域126を有している。プログラム記憶領域121には、CPU10によって実行されるゲームプログラムが記憶される。データ記憶領域126には、ゲーム処理に必要な各種データが記憶される。プログラム記憶領域121内のプログラム、及びデータ記憶領域126内のデータの一部は、光ディスク4に予め記憶されたデータが、ゲーム処理に際して光ディスク4から読み出されたものである。

50

【0084】

プログラム記憶領域121には、メイン処理プログラム122、構え処理プログラム123、攻撃開始処理プログラム124、当たり判定処理プログラム125等が記憶されている。メイン処理プログラム122は、図13に示されている後述するメイン処理をCPU10に実行させるためのプログラムである。構え処理プログラム123は、図15に示されている後述する構え処理をCPU10に実行させるためのプログラムである。攻撃開始処理プログラム124は、図16に示されている後述する攻撃開始処理をCPU10に実行させるためのプログラムである。当たり判定処理プログラム125は、図17及び図18に示されている後述する当たり判定処理をCPU10に実行させるためのプログラムである。

10

【0085】

データ記憶領域126には、コントローラデータ127、推定姿勢データ128、剣姿勢データ129、剣位置データ130、振り方向データ131、攻撃フラグ132、位置関係データ133、移動軌跡データ134、指向性データ135、優先順位データ136、剣振りアニメーションデータ137、アニメーション設定データ138、オブジェクトデータ139等が記憶される。

【0086】

コントローラデータ127は、第1コントローラ7からゲーム装置3へ送信されたコントローラデータである。上述したように、コントローラデータは、第1コントローラ7からゲーム装置3へ1/200秒に1回の割合で送信される。このため、外部メインメモリ12に記憶されるコントローラデータ127は、この割合で更新される。本実施形態においては、データ記憶領域126には、古いコントローラデータが最新のコントローラデータに書き換えられることによって古いコントローラデータは破棄されるが、古いコントローラデータを利用してデータの補正等を行う場合には、例えば数回分のコントローラデータを記憶しておくようにしてもよい。

20

【0087】

コントローラデータ127には、角速度データ1271、加速度データ1272、マーカ座標データ1273、及び操作データ1274が含まれる。角速度データ1271は、ジャイロセンサユニット9(図6参照)によって検出された図7に示すXYZの3軸周りに関する第1コントローラ7の角速度を示すデータである。加速度データ1272は、加速度センサ701(図6参照)によって検出された図2に示すXYZの3軸方向に関する第1コントローラ7の加速度を示すデータ、及び加速度センサ85(図6参照)によって検出された図4に示すxyzの3軸方向に関する第2コントローラ8の加速度を示すデータである。

30

【0088】

マーカ座標データ1273は、撮像情報演算部74の画像処理回路744によって算出されるマーカ座標を示すデータである。マーカ座標は、撮像画像に対応する平面上の位置を表すための2次元座標系で表現される。なお、撮像素子743によって2つのマーカ6R及び6Lが撮像される場合には、2つのマーカ座標が算出される。一方、撮像素子743の撮像可能な範囲内にマーカ6R及び6Lのいずれか一方しか位置しない場合には、撮像素子743によって1つのマーカのみが撮像され、1つのマーカ座標が算出される。また、撮像素子743の撮像可能な範囲内にマーカ6R及び6Lが両方とも位置しない場合には、撮像素子743によってマーカが撮像されず、マーカ座標は算出されない。したがって、マーカ座標データ1273は、2つのマーカ座標を示す場合もあるし、1つのマーカ座標を示す場合もあるし、マーカ座標がないことを示す場合もある。

40

【0089】

操作データ1274は、第1コントローラ7の操作ボタン72a~72iの入力状態を示すデータと、第2コントローラ8のアナログジョイスティック84a、Cボタン84b、及びZボタン84cの入力状態を示すデータとを含む。ゲーム装置3のCPU10は、操作データ1274によって示される機能を実現するための処理を行う。

50

【0090】

推定姿勢データ128は、第1コントローラ7の姿勢を示すデータである。この推定姿勢データ128は、無線コントローラモジュール19によって第1コントローラ7からのコントローラデータが受信されて角速度データ1271が更新される毎に、角速度データ1271に基づいて更新される。

【0091】

剣姿勢データ129は、剣オブジェクト104の姿勢を示すデータである。剣位置データ130は、仮想ゲーム空間における剣オブジェクト104の位置を示すデータである。剣姿勢データ129及び剣位置データ130は、第1コントローラ7の姿勢及び位置を反映したものとなるように、推定姿勢データ128に基づいて適宜更新される。

10

【0092】

振り方向データ131は、プレイヤーによって第1コントローラ7が振られた方向（第1コントローラ7が動いた方向）を示すデータである。この振り方向データ131は、角速度データ1271に基づいて算出される。

【0093】

攻撃フラグ132は、剣オブジェクト104による攻撃動作の開始が指示されたか否かを示すデータである。後に詳述するが、プレイヤーが第1コントローラ7を振るゲーム操作を行うと、CPU10は、第1コントローラ7の角速度（角速度データ1271が示す角速度）が所定値以上であるか否かを判定する。そして、CPU10は、第1コントローラ7の角速度が所定値以上であると判定した場合、剣オブジェクト104による攻撃動作が指示されたものと判断して、攻撃フラグ132をONに設定する。また、この攻撃フラグ132は、プレイヤーオブジェクト101が剣オブジェクト104を振る様子を示す剣振りのアニメーションの内容が設定された後に（後述する図17のステップS126の処理が行われた後に）、OFFに設定される。

20

【0094】

位置関係データ133は、攻撃フラグ132がONに設定されたときのプレイヤーオブジェクト101と敵オブジェクト等のノンプレイヤーオブジェクトとの位置関係を示すデータである。この位置関係データ133には、プレイヤーオブジェクト101の重心座標と敵オブジェクトの重心座標との位置関係を示す情報のみならず、敵オブジェクトの構えや敵オブジェクトが向いている方向といった敵オブジェクト102の姿勢を示す情報も含まれている。

30

【0095】

移動軌跡データ134は、プレイヤーオブジェクト101が剣オブジェクト104を振る剣振り動作を行う際の剣オブジェクト104の移動軌跡を示すデータである。この移動軌跡データ134が示す移動軌跡は、剣姿勢データ129、剣位置データ130、及び振り方向データ131に基づいて計算される。

【0096】

指向性データ135は、一部の敵オブジェクト（本実施形態では敵オブジェクト102）に対して設定されるデータである。この指向性データ135は、敵オブジェクト102に対して剣オブジェクト104がヒット可能なヒット方向を示すデータである。

40

【0097】

プレイヤーオブジェクト101が敵オブジェクト102を剣オブジェクト104で攻撃する際に、振り方向データ131が示す第1コントローラ7の振り方向が攻撃対象である敵オブジェクト102に対応する指向性データ135が示すヒット方向と一致した場合、その敵オブジェクト102に対してヒット処理が行われる。ここで、ヒット処理は、剣オブジェクト104を敵オブジェクト102にヒットさせると共に、剣オブジェクト104がヒットした敵オブジェクト102に対してダメージを与える処理である。

【0098】

一方、振り方向データ131が示す第1コントローラ7の振り方向がヒット方向と一致しなかった場合、攻撃対象である敵オブジェクト102に対して弾き処理が行われる。こ

50

ここで、弾き処理は、敵オブジェクト102に剣オブジェクト104を弾き返す動作を行わせる処理である。この弾き処理が行われた場合には、剣オブジェクト104が敵オブジェクト102に衝突したとしても、敵オブジェクト102にダメージが与えられることはない。

【0099】

優先順位データ136は、移動軌跡データ134が示す剣オブジェクト104の移動軌跡上に攻撃対象となる複数の敵オブジェクトが存在する場合に設定されるデータである。この優先順位データ136は、剣オブジェクト104の移動軌跡上に存在する複数の敵オブジェクトに対する優先順位を示すデータである。本実施形態では、振り方向データ131が示す振り方向に対応する方向に剣オブジェクト104が振られた際に剣オブジェクト104が最初に衝突する敵オブジェクトの優先順位が最も高くなるように、上記複数の敵オブジェクトに対して優先順位が設定される。後述するが、この優先順位データ136が示す優先順位が最も高い敵オブジェクトに対して、ヒット処理を行うか又は弾き処理を行うかを決定する処理が行われ、その処理の結果を示すアニメーションが液晶テレビ2の画面に表示される。

10

【0100】

剣振りアニメーションデータ137は、モーションブラー効果を利用して、プレイヤーオブジェクト101が剣オブジェクト104を振る様子を画面表示するための動画像データである。本実施形態では、剣振りアニメーションデータ137として、ヒット処理用アニメーションデータ、弾き処理用アニメーションデータ、及び空振り処理用アニメーションデータの3種類のアニメーションデータがデータ記憶領域126に記憶されている。ここで、ヒット処理用アニメーションデータは、剣オブジェクト104が敵オブジェクトにヒットする様子を画面表示するためのアニメーションデータである。弾き処理用アニメーションデータは、剣オブジェクト104が敵オブジェクト(例えば敵オブジェクト102)に弾き返される様子を画面表示するためのアニメーションデータである。空振り処理用アニメーションデータは、プレイヤーオブジェクト101が剣オブジェクト104を空振りする様子を画面表示するためのアニメーションデータである。

20

【0101】

なお、剣オブジェクト104の空振りは、剣オブジェクト104が敵オブジェクトにヒットする場合と同様に、プレイヤーオブジェクト101が剣を振り切る動作である。このため、ヒット処理用アニメーションデータを、空振り処理用アニメーションデータとして使用してもよい。すなわち、プレイヤーオブジェクト101が剣を振る様子を示すアニメーションを、ヒット処理用アニメーションデータ、及び弾き処理用アニメーションデータの2種類のアニメーションデータによって実現してもよい。また、本実施形態では、剣オブジェクト104が敵オブジェクト102に弾き返される様子を画面表示するために弾き処理用アニメーションデータが使用される場合について説明するが、弾き処理に応じた画面表示を、ヒット処理用アニメーションデータ、及び弾き処理用アニメーションデータを連続再生することによって実現してもよい。

30

【0102】

アニメーション設定データ138は、剣オブジェクト104による攻撃動作が行われた結果として液晶テレビ2に画面表示すべきアニメーションの内容を示すデータである。剣オブジェクト104による攻撃動作が行われた場合、このアニメーション設定データ138が示す設定情報に応じたアニメーションが再生される。例えば「ヒット処理」を示すアニメーション設定データ138がデータ記憶領域126に記憶されている場合、剣振りアニメーションデータ137のうちのヒット処理用アニメーションデータが再生される。また、例えば「弾き処理」を示すアニメーション設定データ138がデータ記憶領域126に記憶されている場合、剣振りアニメーションデータ137のうちの弾き処理用アニメーションデータが再生される。

40

【0103】

オブジェクトデータ139は、ゲーム処理で使用されるプレイヤーオブジェクト101や

50

敵オブジェクト102等の各オブジェクトに関するデータであり、各オブジェクトの位置座標データ、モデリングデータ、テクスチャデータ（RGB値）等から構成される。

【0104】

なお、図には示されていないが、データ記憶領域126には、ゲーム処理中に用いられる音声データや、仮想ゲーム空間の様子を画面表示するための仮想カメラの制御に関するデータ等が記憶されるが、これらのデータは本発明とは直接関係しないので、ここでの詳細な説明は省略する。

【0105】

[メイン処理]

次に、ゲーム装置3によって実行されるゲーム処理について説明する。ゲーム装置3の電源が投入されると、ゲーム装置3のCPU10は、ROM/RTC13に記憶されている起動プログラムを実行する。これにより、外部メインメモリ12等の各ユニットが初期化される。そして、光ディスク4に記憶されたゲームプログラムが外部メインメモリ12に読み込まれ、CPU10によって当該ゲームプログラムの実行が開始される。

【0106】

図13は、ゲーム装置3で実行されるメイン処理の一例を示すフローチャートである。まず、CPU10は、以降の処理において用いられるデータの初期化処理を実行する（ステップS1）。具体的には、CPU10は、以降の処理に使用される外部メインメモリ12のデータ記憶領域126内の各種変数やフラグ等を初期化する。そして、CPU10は、プレイヤーオブジェクト101、及び敵オブジェクト102等のノンプレイヤーオブジェクトを仮想ゲーム空間に配置する（ステップS2）。具体的には、CPU10は、ゲーム開始時における仮想カメラの初期位置、及び各種オブジェクトの初期配置を示すデータをデータ記憶領域126に格納する。

【0107】

続いて、仮想ゲーム空間が構築されてゲーム画像が液晶テレビ2に表示される。すなわち、CPU10は、3次元の仮想ゲーム空間を構築し、各オブジェクトの初期配置を示すデータに従って各オブジェクトを仮想ゲーム空間内に配置する。そして、CPU10は、この仮想ゲーム空間を仮想カメラの視点から見たゲーム画像をGPU11bに生成させる。このゲーム画像が液晶テレビ2へ出力されることによって、液晶テレビ2にゲーム画像が表示される。以降、ステップS3～S16の処理ループが1フレーム（本実施形態では1/60秒）毎に繰り返されることによって、ゲームが進行していく。

【0108】

ステップS2の処理に続いて、CPU10は、データ記憶領域126に記憶されている情報に基づいて、プレイヤーオブジェクト101が剣オブジェクト104を振る剣振りのアニメーションの再生中であるか否かを判定する（ステップS3）。剣振りアニメーションの再生中であるとCPU10によって判定された場合（ステップS3：YES）、後述するステップS14へ処理が進められる。

【0109】

CPU10は、剣振りアニメーションの再生中ではないと判定した場合（ステップS3：NO）、コントローラデータを取得する（ステップS4）。具体的には、CPU10は、無線コントローラモジュール19によって受信された第1コントローラ7からのコントローラデータをコントローラデータ127としてデータ記憶領域126に格納する。

【0110】

次に、CPU10は、プレイヤーオブジェクト101の移動が指示されたか否かを判定する（ステップS5）。具体的には、CPU10は、コントローラデータ127の一部としてデータ記憶領域126に格納された操作データ1274に、第2コントローラ8のアナログジョイスティック84aが操作されたことを示す操作データが含まれているか否かを判定する。

【0111】

CPU10は、プレイヤーオブジェクト101の移動が指示されたと判定した場合（ステ

10

20

30

40

50

ップS5: YES)、すなわち操作データ1274にアナログジョイスティック84aの操作データが含まれていた場合、その操作データに応じた位置へプレイオブジェクト101を移動させる(ステップS6)。このステップS6の処理が行われることにより、オブジェクトデータ139がプレイオブジェクト101の最新の位置を示すように更新される。一方、プレイオブジェクト101の移動が指示されていないとCPU10によって判定された場合(ステップS5:NO)、すなわち操作データ1274にアナログジョイスティック84aの操作データが含まれていなかった場合、後述するステップS7へ処理が進められる。

【0112】

CPU10は、ステップS6においてプレイオブジェクト101の移動処理を行った場合、又はステップS5において「NO」と判定した場合、攻撃フラグ132がONに設定されているか否かを判定する(ステップS7)。

10

【0113】

ここで、プレイヤによる第1コントローラ7の振り操作について説明する。上述したように角速度データ1271が示す角速度は、図7に示されるように、Z軸周りのロール角に関する角速度、X軸周りのピッチ角に関する角速度、及びY軸周りのヨー角に関する角速度の3方向の角速度である。プレイヤが第1コントローラ7を左から右へ振った場合(図10参照)、又はプレイヤが第1コントローラ7を右から左へ振った場合(図11参照)、第1コントローラ7の振り操作に伴ってY軸周りのヨー角に関する角速度が一時的に大きくなる。また、図14に例示されるように、プレイヤが第1コントローラ7を振り下

20

【0114】

CPU10は、攻撃フラグ132がOFFに設定されていると判定した場合(ステップS7:NO)、角速度データ1271が示す第1コントローラ7の角速度が所定値以上であるか否かを判定する(ステップS8)。上述のように、Y軸周りのヨー角に関する角速度、及びX軸周りのピッチ角に関する角速度に基づいて第1コントローラ7の振り操作が開始されたか否かを判定することができる。このため、このステップS8において、開始判定手段として機能するCPU10は、角速度データ1271が示すY軸周りのヨー角に関する角速度、又はX軸周りのピッチ角に関する角速度が所定値以上であるか否かを判定する。本実施形態では、Y軸周りのヨー角に関する角速度、及びX軸周りのピッチ角に関する角速度のいずれか一方でも所定値上である場合、このステップS8において「YES」と判定されて、後述するステップS10へ処理が進められる。一方、Y軸周りのヨー角に関する角速度、及びX軸周りのピッチ角に関する角速度がいずれも所定値未満である場合、このステップS8において「NO」と判定されて、後述するステップS9へ処理が進められる。

30

【0115】

このように、CPU10は、ジャイロセンサユニット9からの角速度データに基づいて、第1コントローラ7の振り操作が開始されたか否かを判定する。

40

【0116】

なお、Y軸周りのヨー角に関する角速度の判定に使用される所定値と、X軸周りのピッチ角に関する角速度に使用される所定値とは、同じ値であってもよいし、異なる値であってもよい。所定値は、ゲームの内容等に応じて適切な値に設定される。

【0117】

CPU10は、角速度データ1271が示す第1コントローラ7の角速度が所定値未満であると判定した場合(ステップS8:NO)、プレイオブジェクト101に対して攻撃指示がなかったものと判断して、プレイオブジェクト101に剣オブジェクト104

50

を構えさせるための構え処理を実行する（ステップS9）。この構え処理の詳細については、図15に基づいて後述する。

【0118】

CPU10は、角速度データ1271が示す第1コントローラ7の角速度が所定値以上であると判定した場合（ステップS8：YES）、プレイヤーオブジェクト101に対して攻撃指示があったと判断して、プレイヤーオブジェクト101に剣オブジェクト104を振る動作を開始させるための攻撃開始処理を実行する（ステップS10）。この攻撃開始処理の詳細については、図16に基づいて後述する。

【0119】

一方、ステップS7において攻撃フラグ132がONに設定されていると判定した場合（ステップS7：YES）、CPU10は、剣オブジェクト104が敵オブジェクト（例えば敵オブジェクト102）にヒットしたか否かを判定する当たり判定処理を実行する（ステップS12）。この当たり判定処理の詳細については図17及び図18に基づいて後述するが、当たり判定処理が行われることによって、当たり判定処理の結果を示す剣オブジェクト104の剣振りアニメーションが設定される。このため、CPU10は、ステップS12において当たり判定処理を行った後、設定された剣振りアニメーションの再生を開始する（ステップS13）。

10

【0120】

CPU10は、ステップS3で「YES」と判定した場合、ステップS9の処理を実行した場合、ステップS10の処理を実行した場合、又はステップS13の処理を実行した場合、その他の処理を行う（ステップS14）。具体的には、CPU10は、仮想ゲーム空間に登場するプレイヤーオブジェクト101及び剣オブジェクト104以外のノンプレイヤーオブジェクト（例えば敵オブジェクト102）を移動させるといった処理を行う。そして、CPU10は、ステップS3～ステップS14の処理を行った結果を示すゲーム画像をGPU11bに生成させて、生成されたゲーム画像を液晶テレビ2に表示させる（ステップS15）。

20

【0121】

ステップS15の処理に続いて、CPU10は、電源ボタン24、リセットボタン25、又は操作ボタン72hの操作の有無に基づいて、ゲームの終了が指示されたか否かを判定する（ステップS16）。ゲームの終了が指示されていないとCPU10によって判定された場合（ステップS16：NO）、ステップS3へ処理が戻されて、ステップS3以降のゲーム処理が繰り返される。逆に、ゲームの終了が指示されたとCPU10によって判定された場合（ステップS16：YES）、一連のゲーム処理が終了する。

30

【0122】

[構え処理]

図15は、図13のステップS9における構え処理の詳細フローチャートである。上述したステップS8において第1コントローラ7の角速度が所定値未満であると判定した場合（ステップS8：NO）、CPU10は、角速度データ1271に基づいて、第1コントローラ7の姿勢を算出する（ステップS91）。具体的には、CPU10は、推定姿勢データ128が示す第1コントローラ7の姿勢を、角速度データ1271が示す角速度に基づいて更新することによって、第1コントローラ7の現在の姿勢を算出する。

40

【0123】

続いて、CPU10は、推定姿勢データ128が示す第1コントローラ7の最新の姿勢に基づいて、剣オブジェクト104の姿勢及び位置を算出する（ステップS92）。具体的には、CPU10は、推定姿勢データ128が示す第1コントローラ7の姿勢に基づいて、剣オブジェクト104が第1コントローラ7と同じ向きを向くように、剣オブジェクト104の姿勢を算出する。そして、CPU10は、オブジェクトデータ139が示すプレイヤーオブジェクト101の位置やプレイヤーオブジェクト101の腕の長さ等を考慮して、仮想ゲーム空間における剣オブジェクト104の位置を算出する。このステップS92の処理で算出された剣オブジェクト104の姿勢及び位置は、剣姿勢データ129及び剣

50

位置データ130としてデータ記憶領域126に記憶される。

【0124】

このステップS92の処理が行われることにより、第1コントローラ7の姿勢が剣オブジェクト104の姿勢及び位置に反映される。そして、剣姿勢データ129、剣位置データ130、オブジェクトデータ139等に基づいてステップS15の表示処理が行われることで、プレイヤーが第1コントローラ7を構えた姿勢と同じようにプレイヤーオブジェクト101が剣オブジェクト104を構えた状態を示すゲーム画像が液晶テレビ2に表示される。

【0125】

[攻撃開始処理]

図16は、図13のステップS10における攻撃開始処理の詳細フローチャートである。上述したステップS8において第1コントローラ7の角速度が所定値以上であると判定した場合(ステップS8: YES)、CPU10は、仮想ゲーム空間における剣オブジェクト104の姿勢を変化させる(ステップS101)。具体的には、CPU10は、角速度データ1271が示す3軸周りの角速度に基づいて、剣姿勢データ129を更新する。

10

【0126】

このステップS101の処理が行われてからステップS15の表示処理が行われることで、プレイヤーが第1コントローラ7の振り操作を開始した時点の第1コントローラ7の姿勢と同じ姿勢でプレイヤーオブジェクト101が剣オブジェクト104を振っている状態を示すゲーム画像が液晶テレビ2に表示される。

20

【0127】

ステップS101の処理に続いて、方向判別手段として機能するCPU10は、第1コントローラ7の振り方向を判別する(ステップS102)。具体的には、CPU10は、角速度データ1271が示すY軸周りのヨー角に関する角速度、及びX軸周りのピッチ角に関する角速度に基づいて、第1コントローラ7の振り方向(動作方向)を判別する。本実施形態では、第1コントローラ7の振り方向は、下方向、上方向、右方向、及び左方向の4方向に判別される。このステップS102の処理で判別された振り方向は、振り方向データ131としてデータ記憶領域126に記憶される。

【0128】

ここで、下方向は、プレイヤーが第1コントローラ7を上から下へ向けて振る操作を行ったときの第1コントローラ7の動作方向である(図14参照)。上方向は、プレイヤーが第1コントローラ7を下から上へ向けて振る操作を行ったときの第1コントローラ7の動作方向である。右方向は、プレイヤーが第1コントローラ7を左から右へ向けて振る操作を行ったときの第1コントローラ7の動作方向である(図10参照)。左方向は、プレイヤーが第1コントローラ7を右から左へ向けて振る操作を行ったときの第1コントローラ7の動作方向である(図11参照)。

30

【0129】

なお、本実施形態では、第1コントローラ7の振り方向を上述した4方向に判別する場合について説明するが、判別される振り方向は4方向に限定されるものではない。例えば、第1コントローラ7の振り方向は、上記4方向に右斜め下方向、左斜め下方向、右斜め上方向、及び左斜め上方向を加えた8方向に判別されてもよい。

40

【0130】

ところで、このステップS102における振り方向を判別する処理は、ステップS8において角速度が所定値以上であると判定された場合に、ステップS8の処理に続けて実行される。すなわち、第1コントローラ7の振り方向を判別する処理は、本実施形態では、第1コントローラ7の振り操作が開始されたと判定されたときに行われる。

【0131】

CPU10は、第1コントローラ7の振り方向を判別した後、攻撃フラグ132をONに設定する(ステップS103)。攻撃フラグ132がONに設定されることにより、上記ステップS7において「YES」と判定されて、後述する当たり判定処理へ処理が進め

50

られる。

【0132】

[当たり判定処理]

図17及び図18は、図13のステップS12における当たり判定処理の詳細フローチャートである。上記ステップS7において攻撃フラグ132がONに設定されていると判定した場合(ステップS7: YES)、位置関係検出手段として機能するCPU10は、データ記憶領域126に記憶されている剣位置データ130及びオブジェクトデータ139を参照して、剣オブジェクト104と、敵オブジェクトとの位置関係を検出する(ステップS121)。このステップS121の処理による検出結果は、位置関係データ133としてデータ記憶領域126に格納される。

10

【0133】

ステップS121の処理に続いて、CPU10は、剣姿勢データ129、剣位置データ130、及び振り方向データ131に基づいて、剣オブジェクト104の移動軌跡を計算する(ステップS122)。すなわち、振り方向データ131が示す振り方向と対応する方向に剣オブジェクト104が振られた場合に、剣姿勢データ129及び剣位置データ130が示す位置及び姿勢に構えられた剣オブジェクト104がどのような軌跡を描きながら動作するかを計算する。そして、衝突判定手段として機能するCPU10は、剣オブジェクト104が判定対象としての敵オブジェクトと衝突するか否かを判定する(ステップS123)。具体的には、CPU10は、ステップS121の処理によって得られた位置関係データ133を参照して、ステップS122の処理で計算した移動軌跡上に敵オブジェクト(例えば敵オブジェクト102)が存在するか否かに基づいて、剣オブジェクト104が敵オブジェクトに衝突するか否かを判定する。

20

【0134】

CPU10は、ステップS123において衝突判定を行った後、その判定結果に基づいて、剣オブジェクト104が衝突する判定対象(以下「衝突対象」という。)があるか否かを判定する(ステップS124)。衝突対象がないと判定した場合(ステップS124: NO)、CPU10は、剣オブジェクト104を空振りさせる空振り処理を行う(ステップS125)。

【0135】

CPU10は、ステップS125において空振り処理を行った後、剣振りアニメーションを設定する(ステップS126)。具体的には、CPU10は、空振り処理を示すアニメーション設定データ138をデータ記憶領域126に格納する。そして、CPU10は、ステップS103の処理によってONに設定した攻撃フラグ132をOFFに設定する(ステップS127)。

30

【0136】

このステップS127の処理が行われた後、処理がステップS13へ進められる。すなわち、CPU10は、ステップS126の処理で設定した内容の剣振りアニメーションの再生を開始する(ステップS13)。ここでは、ステップS125の処理が行われてからステップS126の処理が行われることによって、「空振り処理」を示すアニメーション設定データ138がデータ記憶領域126に記憶されている。このため、CPU10は、剣振りアニメーションデータ137の中から空振り処理用アニメーションデータを選択してその再生を開始する。このように、空振り処理用アニメーションデータの再生が開始された場合、空振り処理用アニメーションの再生が完了するまで上記ステップS3において「YES」と判定され続けて、ステップS15の表示処理が繰り返される。その結果、プレイヤーオブジェクト101が剣オブジェクト104を空振りする様子が液晶テレビ2に表示される。

40

【0137】

一方、CPU10は、剣オブジェクト104が衝突する衝突対象があると判定した場合(ステップS124: YES)、衝突対象が複数あるか否かを判定する(ステップS129)。衝突対象が複数あるとCPU10によって判定された場合(ステップS129: Y

50

YES)、後述するステップS136(図18参照)へ処理が進められる。

【0138】

CPU10は、衝突対象が1つであると判定した場合(ステップS129:NO)、その衝突対象が指向性を有しているか否かを判定する(ステップS130)。具体的には、CPU10は、衝突対象に対応する指向性データ135がデータ記憶領域126に記憶されているか否かを判定する。衝突対象に対応する指向性データ135がデータ記憶領域126に記憶されている場合、その衝突対象が指向性を有していると判定することができる。逆に、衝突対象に対応する指向性データ135がデータ記憶領域126に記憶されていない場合、その衝突対象が指向性を有していないと判定することができる。

【0139】

図19は、プレイヤーが第1コントローラ7を左から右へ振った場合に液晶テレビ2に表示されるアニメーションの一例を示す画面図である。図19に例示される敵オブジェクト106には、指向性データ135が設定されていない。この敵オブジェクト106のように指向性を有していない衝突対象と剣オブジェクト104が衝突する場合、第1コントローラ7の振り方向とは無関係に、衝突対象に対してヒット処理が行われる。

【0140】

衝突処理決定手段として機能するCPU10は、衝突対象が指向性を有していないと判定した場合(ステップS130:NO)、衝突対象に対してヒット処理を行う(ステップS131)。具体的には、CPU10は、剣オブジェクト104を衝突対象(ここでは敵オブジェクト106)にヒットさせると共に、剣オブジェクト104がヒットした衝突対象に対してダメージを与える処理を実行する。

【0141】

このステップS131の処理が行われた場合にも、上述したステップS126以降の処理が行われる。ステップS131の処理に続いてステップS126の処理を行う場合、CPU10は、ステップS126の処理において、「ヒット処理」を示すアニメーション設定データ138をデータ記憶領域126に格納する。この場合、CPU10は、ステップS13において、剣振りアニメーションデータ137の中からヒット処理用アニメーションデータを選択してその再生を開始する。このように、ヒット処理用アニメーションデータの再生が開始された場合、ヒット処理用アニメーションの再生が完了するまで上記ステップS3において「YES」と判定され続けて、ステップS15の表示処理が繰り返される。その結果、剣オブジェクト104が衝突対象にヒットしてその衝突対象にダメージが与えられる様子が液晶テレビ2に表示される。図19には、衝突対象としての敵オブジェクト106に対してステップS131の処理でヒット処理が行われた場合に液晶テレビ2に表示されるゲーム画像が例示されている。なお、敵オブジェクト106は指向性を有していないので、敵オブジェクト106に対する剣オブジェクト104の振り方向(第1コントローラ7の振り方向)がどの方向であったとしても、敵オブジェクト106に対してヒット処理が行われる。

【0142】

一方、CPU10は、衝突対象が指向性を有していると判定した場合(ステップS130:YES)、上述したステップS121の処理で検出した位置関係において、上述したステップS102(図16参照)の処理で判別した第1コントローラ7の振り方向が、剣オブジェクト104が衝突する衝突対象に対して設定されている指向性データ135が示すヒット方向と一致するか否かを判定する(ステップS132)。第1コントローラ7の振り方向がヒット方向と一致するとCPU10によって判定された場合(ステップS132:YES)、処理がステップS131へ進められる。すなわち、指向性を有する衝突対象(例えば敵オブジェクト102)に対して、ヒット処理が行われる。このように、指向性を有する衝突対象に対して振り方向とヒット方向とが一致すると判定された場合、例えば図10に例示されるようなアニメーションが液晶テレビ2に表示される。

【0143】

CPU10は、第1コントローラ7の振り方向がヒット方向と一致しないと判定した場

10

20

30

40

50

合（ステップS 1 3 2：NO）、衝突対象に対して弾き処理を行う（ステップS 1 3 3）。具体的には、CPU 1 0は、指向性を有する衝突対象（例えば敵オブジェクト1 0 2）に剣オブジェクト1 0 4を弾き返す動作を行わせる。

【0 1 4 4】

このステップS 1 3 3の処理が行われた場合にも、上述したステップS 1 2 6以降の処理が行われる。ステップS 1 3 3の処理に続いてステップS 1 2 6の処理を行う場合、CPU 1 0は、ステップS 1 2 6の処理において、「弾き処理」を示すアニメーション設定データ1 3 8をデータ記憶領域1 2 6に格納する。この場合、CPU 1 0は、ステップS 1 3において、剣振りアニメーションデータ1 3 7の中から弾き処理用アニメーションデータを選択してその再生を開始する。このように、弾き処理用アニメーションデータの再生が開始された場合、弾き処理用アニメーションの再生が完了するまで上記ステップS 3において「YES」と判定され続けて、ステップS 1 5の表示処理が繰り返される。その結果、剣オブジェクト1 0 4が指向性を有する衝突対象（例えば敵オブジェクト1 0 2）に弾き返される様子が液晶テレビ2に表示される（図1 1参照）。

10

【0 1 4 5】

このように、CPU 1 0は、ステップS 1 3 0～ステップS 1 3 3の処理を実行することによって、ステップS 1 0 2の処理で判別した第1コントローラ7の振り方向に基づいて、剣オブジェクト1 0 4が衝突する衝突対象に対してヒット処理を行うか否か（本実施形態ではヒット処理を行うか或いは弾き処理を行うか）を決定する。

20

【0 1 4 6】

例えば指向性を有する敵オブジェクト1 0 2（図1 0及び図1 1参照）に対しては、ヒット方向として右方向を示す指向性データ1 3 5がデータ記憶領域1 2 6に記憶されている。このため、ステップS 1 0 2の処理で判別された第1コントローラ7の振り方向が右方向であった場合、振り方向とヒット方向とが一致するので、敵オブジェクト1 0 2に対してヒット処理が行われてその様子を示すアニメーションが液晶テレビ2に表示される（図1 0参照）。一方、ステップS 1 0 2の処理で判別された第1コントローラ7の振り方向が左方向であった場合に、振り方向とヒット方向とが一致しないので、敵オブジェクト1 0 2に対して弾き処理が行われてその様子を示すアニメーションが液晶テレビ2に表示される（図1 1参照）。

30

【0 1 4 7】

なお、ステップS 1 3 2の判定処理は、上記ステップS 1 2 1の処理で検出された剣オブジェクト1 0 4とノンプレイヤーオブジェクトとの位置関係を考慮して行われる。例えば、敵オブジェクト1 0 2がプレイヤーオブジェクト1 0 1の方を向いている状態（図1 0及び図1 1参照）と、プレイヤーオブジェクト1 0 1に対して背を向いている状態（不図示）とでは、敵オブジェクト1 0 2に対する指向性データ1 3 5が示すヒット方向が相異なる。つまり、敵オブジェクト1 0 2がプレイヤーオブジェクト1 0 1の方を向いているときには、敵オブジェクト1 0 2に対するヒット方向が右方向であるが、敵オブジェクト1 0 2がプレイヤーオブジェクト1 0 1に背を向いているときには、敵オブジェクト1 0 2に対するヒット方向が左方向である。このように、剣オブジェクト1 0 4（プレイヤーオブジェクト1 0 1）とノンプレイヤーオブジェクトとの位置関係によって、指向性データ1 3 5が示すヒット方向が変化する。このため、剣オブジェクト1 0 4とノンプレイヤーオブジェクトとの位置関係を考慮してヒット処理を行うか否かを決定することにより、ヒット処理を行うか否かを正確に決定することができる。

40

【0 1 4 8】

また、例えば敵オブジェクト1 0 2が盾を上に向けることによって敵オブジェクト1 0 2のヒット方向が変化する場合や、例えば剣オブジェクト1 0 4と敵オブジェクト1 0 2との間に障害物となるオブジェクトが位置している場合にも、剣オブジェクト1 0 4とノンプレイヤーオブジェクトとの位置関係を考慮してヒット処理を行うか否かを決定することは有効である。

【0 1 4 9】

50

上述したステップS 1 2 9において、優先順位設定手段として機能するCPU 1 0は、衝突対象が複数あると判定した場合（ステップS 1 2 9：YES）、ステップS 1 2 1の処理で検出したオブジェクト同士の位置関係、及びステップS 1 0 2の処理で判別した第1コントローラ7の振り方向に基づいて、複数の衝突対象に対して優先順位を設定する（ステップS 1 3 6）。具体的には、CPU 1 0は、ステップS 1 2 1において検出したオブジェクト同士の位置関係において、ステップS 1 0 2において判別した振り方向で第1コントローラ7が振られた場合に剣オブジェクト1 0 4が最初に衝突する衝突対象の優先順位が最も高くなるように、複数の衝突対象のそれぞれに対して優先順位を設定する。

【0 1 5 0】

例えば図2 0に例示されるように衝突対象としての敵オブジェクト1 0 2と敵オブジェクト1 0 6とが並んでいる状態において第1コントローラ7が右から左へ振られた場合、剣オブジェクト1 0 4が最初に衝突する衝突対象は、敵オブジェクト1 0 6である。このため、CPU 1 0は、敵オブジェクト1 0 6の優先順位が敵オブジェクト1 0 2の優先順位よりも高くなるように、敵オブジェクト1 0 2及び敵オブジェクト1 0 6に対して優先順位を設定する。

10

【0 1 5 1】

また、例えば図2 1に例示されるように衝突対象としての敵オブジェクト1 0 6と敵オブジェクト1 0 2とが並んでいる状態において第1コントローラ7が右から左へ振られた場合、剣オブジェクト1 0 4が最初に衝突する衝突対象は、敵オブジェクト1 0 2である。このため、CPU 1 0は、敵オブジェクト1 0 2の優先順位が敵オブジェクト1 0 6の優先順位よりも高くなるように、敵オブジェクト1 0 2及び敵オブジェクト1 0 6に対して優先順位を設定する。

20

【0 1 5 2】

このように、CPU 1 0は、剣オブジェクト1 0 4が複数の衝突対象に衝突すると判定した場合に、第1コントローラ7の振り方向、及び剣オブジェクト1 0 4とノンプレイヤーオブジェクトとの位置関係に基づいて、複数の衝突対象に対して優先順位を設定する。このステップS 1 3 6の処理によって設定された優先順位は、優先順位データ1 3 6としてデータ記憶領域1 2 6に格納される。

【0 1 5 3】

ステップS 1 3 6の処理に続いて、CPU 1 0は、最優先の衝突対象に指向性があるか否かを判定する（ステップS 1 3 7）。具体的には、CPU 1 0は、データ記憶領域1 2 6に記憶されている優先順位データ1 3 6を参照して、剣オブジェクト7が衝突する可能性がある複数の衝突対象の中から優先順位が最も高い衝突対象を特定する。そして、CPU 1 0は、その衝突対象に対する指向性データ1 3 5がデータ記憶領域1 2 6に記憶されているか否かに基づいて、その衝突対象が指向性を有しているか否かを判定する。

30

【0 1 5 4】

CPU 1 0は、最優先の衝突対象に指向性がないと判定した場合（ステップS 1 3 7：NO）、最優先の衝突対象に対してヒット処理を行う（ステップS 1 3 8）。このステップS 1 3 8のヒット処理は、ステップS 1 3 0で「NO」と判定されてステップS 1 3 1で行われるヒット処理と同様に行われる。このステップS 1 3 8の処理が行われると、処理がステップS 1 2 6（図1 7参照）へ進められる。指向性がない衝突対象としては、本実施形態では敵オブジェクト1 0 6が挙げられる。敵オブジェクト1 0 6が最優先の衝突対象に設定されてヒット処理が行われた場合、図2 0に例示されるように、剣オブジェクト1 0 4が敵オブジェクト1 0 6にヒットする様子を示すヒット処理用アニメーションが再生されることになる。

40

【0 1 5 5】

CPU 1 0は、最優先の衝突対象に指向性があると判定した場合（ステップS 1 3 7：YES）、第1コントローラ7の振り方向が、最優先の衝突対象に対して設定されているヒット方向と一致するか否かを判定する（ステップS 1 3 9）。具体的には、CPU 1 0は、最優先の衝突対象に対する指向性データ1 3 5を参照して、最優先の衝突対象に対し

50

て予め設定されているヒット方向を判別する。そして、CPU10は、ステップS102の処理で判別した振り方向（振り方向データ131が示す振り方向）が、最優先の衝突対象のヒット方向と一致するか否かを判定する。

【0156】

CPU10は、振り方向とヒット方向とが一致すると判定した場合（ステップS139：YES）、最優先の衝突対象に対して上述したステップS138のヒット処理を行う。指向性がある衝突対象としては、本実施形態では敵オブジェクト102が挙げられる。敵オブジェクト102が最優先の衝突対象に設定されてヒット処理が行われた場合、図22に例示されるように、剣オブジェクト104が敵オブジェクト102にヒットする様子を示すヒット処理用アニメーションが再生されることになる。

10

【0157】

逆に、振り方向とヒット方向とが一致しないと判定した場合（ステップS139：NO）、CPU10は、最優先の衝突対象に対して弾き処理を行う（ステップS140）。このステップS140の弾き処理は、ステップS133で行われる弾き処理と同様に行われる。このステップS140の処理が行われると、処理がステップS126（図17参照）へ進められる。例えば敵オブジェクト102が最優先の衝突対象に設定されて弾き処理が行われた場合、図21に例示されるように、剣オブジェクト104が敵オブジェクト102に弾き返される様子を示す弾き処理用アニメーションが再生されることになる。

【0158】

このように、CPU10は、複数の衝突対象に対して優先順位を設定した場合、優先順位が最上位の衝突対象に対してヒット処理を行うか弾き処理を行うかを決定する。

20

【0159】

[本実施形態の作用効果]

以上説明したように、本実施形態によれば、剣オブジェクト104が敵オブジェクトに衝突すると判定された場合に、第1コントローラ7の振り方向に基づいて、敵オブジェクトに対してヒット処理を行うか否かが決定される。すなわち、剣オブジェクト104が敵オブジェクトに衝突した場合に無条件にヒット処理が行われるのではなく、第1コントローラ7の振り方向に基づいて、ヒット処理を行うか否かが決定される。したがって、第1コントローラ7の振り方向（すなわち剣オブジェクト104が敵オブジェクトに衝突する方向）に応じて、衝突時の敵オブジェクトの反応を示す演出を容易に切り替えることができる。

30

【0160】

また、本実施形態では、第1コントローラ7の振り方向だけでなく、剣オブジェクト104と敵オブジェクトとの位置関係を考慮して、敵オブジェクトに対してヒット処理を行うか否かが決定される。このため、剣オブジェクト104と、剣オブジェクト104による攻撃を受ける敵オブジェクトとの位置関係（例えば敵オブジェクト102の向きや姿勢）が変化したとしても、敵オブジェクトに対してヒット処理を行うか否かを適切に決定することができる。

【0161】

また、本実施形態では、剣オブジェクト104が複数の敵オブジェクトと衝突する可能性がある場合に、複数の敵オブジェクトに対して優先順位が設定され、最上位の敵オブジェクトに対してヒット処理を行うか否かが決定される。このため、ヒット処理を行うか否かを決定すべき敵オブジェクトに対して、ヒット処理を行うか否かを適切に決定することができる。

40

【0162】

また、本実施形態では、剣オブジェクト104による攻撃動作を指示する第1コントローラ7の振り操作が開始されたと判定されたタイミングで、第1コントローラ7の振り方向が判別される。言い換えれば、剣オブジェクト104が敵オブジェクトに衝突するよりも前に、第1コントローラ7の振り方向が判別される。このため、剣オブジェクト104が敵オブジェクトに衝突したときに第1コントローラ7の振り方向を判別する場合に比べ

50

て、より早いタイミングで敵オブジェクトに対してヒット処理を行うか否かを決定することができる。その結果、第1コントローラ7の振り操作が行われてから剣振りのアニメーションの再生が開始されるまでのタイムラグを短縮することができる。

【0163】

また、本実施形態では、第1コントローラ7の振り方向を判別する判別処理や、剣オブジェクト104が敵オブジェクトと衝突するか否かの衝突判定が、ジャイロセンサユニット9から出力される角速度データに基づいて行われる。このため、例えば加速度センサ701から出力される加速度データを使用する場合に比べて、上記判別処理及び衝突処理をより正確に行うことができる。

【0164】

[当たり判定処理の変形例]

上述した本実施形態では、剣オブジェクト104が複数の衝突対象と衝突する可能性がある場合に、最優先の衝突対象のみにヒット処理を行うか否かを決定する場合について説明した。これに加えて、最優先の衝突対象に対してヒット処理を行うと決定した場合に、次優先の衝突対象に対してヒット処理を行うか否かを更に決定するようにしてもよい。

【0165】

以下、図23を参照しつつ、当たり判定処理の変形例について説明する。ここで、図23は、当たり判定処理の変形例を示すフローチャートである。本変形例では、上述したステップS129(図17参照)において衝突対象が複数あるとCPU10によって判定された場合(ステップS129: YES)、図18におけるステップS136~ステップS140の処理に代えて、以下に説明するステップS151以降の処理が行われる。

【0166】

CPU10は、衝突対象が複数あると判定した場合(ステップS129: YES)、ステップS136の処理と同様に、オブジェクト同士の位置関係、及び第1コントローラ7の振り方向に基づいて、複数の判定対象に対して優先順位を設定する(ステップS151)。

【0167】

そして、CPU10は、ステップS137の処理と同様に、最優先の衝突対象に指向性があるか否かを判定する(ステップS152)。最優先の衝突対象に指向性があると判定した場合(ステップS152: YES)、CPU10は、ステップS139の処理と同様に、第1コントローラ7の振り方向が、最優先の衝突対象に対して予め設定されたヒット方向と一致するか否かを判定する(ステップS153)。ここで、振り方向とヒット方向とが一致しないと判定した場合(ステップS153: NO)、CPU10は、ステップS140の処理と同様に、最優先の衝突対象に対して弾き処理を行う(ステップS154)。このステップS154の処理が行われた後、処理が上述したステップS126へ進められる。この場合、図21に例示されるように、剣オブジェクト104が最優先の衝突対象である敵オブジェクト102に弾き返される様子を示す弾き処理用アニメーションが再生されることになる。

【0168】

一方、CPU10は、最優先の衝突対象に指向性がないと判定した場合(ステップS152: NO)、又は振り方向がヒット方向と一致すると判定した場合(ステップS153: YES)、ステップS138の処理と同様に、最優先の衝突対象に対してヒット処理を行う(ステップS155)。

【0169】

最優先の衝突対象にヒット処理が行われる場合、最優先の衝突対象を斬った剣オブジェクト104が、次に優先順位が高い衝突対象と衝突することになる。そこで、本変形例においては、ステップS155の処理が行われた場合に、最優先の衝突対象よりも優先順位が低い衝突対象に対して、以下に説明するステップS156以降の処理が行われる。

【0170】

ステップS155の処理を行った場合、CPU10は、次に優先順位が高い衝突対象に

10

20

30

40

50

指向性があるか否かを判定する（ステップS156）。具体的には、CPU10は、次優先の衝突対象である敵オブジェクトに対する指向性データ135がデータ記憶領域126に記憶されているか否かを判定する。

【0171】

CPU10は、次優先の衝突対象に指向性があると判定した場合（ステップS156：YES）、第1コントローラ7の振り方向が、次優先の衝突対象に対して予め設定されているヒット方向と一致するか否かを判定する（ステップS157）。具体的には、CPU10は、ステップS156の処理で判定の対象となった次優先の衝突対象に対応する指向性データ135を参照して、次優先の衝突対象のヒット方向を特定する。そして、CPU10は、振り方向データ131が示す振り方向が、特定した次優先の衝突対象のヒット方向と一致するか否かを判定する。

10

【0172】

CPU10は、第1コントローラ7の振り方向と次優先の衝突対象のヒット方向とが一致しないと判定した場合（ステップS157：NO）、次優先の衝突対象に対して弾き処理を行う（ステップS158）。このステップS158における弾き処理は、弾き処理の対象が最優先の衝突対象ではなく次優先の衝突対象であることを除いて、ステップS140の処理と同様に行われる。

【0173】

このステップS158の処理が行われた後、処理が上述したステップS126へ進められる。この場合、最優先の衝突対象に対してヒット処理が行われ、且つ次優先の衝突対象に対して弾き処理が行われている。このため、図24に例示されるようなアニメーションが再生されることになる。ここで、図24は、敵オブジェクト106が最優先の衝突対象に設定され、敵オブジェクト106の隣に位置する敵オブジェクト102が次優先の衝突対象に設定され、更に第1コントローラ7の振り方向がその敵オブジェクト102に対するヒット方向と一致しない場合に液晶テレビ2に表示されるアニメーションの一例を示す画面図である。図24に例示されるように、敵オブジェクト106に対してヒット処理が行われ、且つ敵オブジェクト106の隣に位置する敵オブジェクト102に対して弾き処理が行われた様子が示されている。

20

【0174】

CPU10は、次優先の衝突対象に指向性がないと判定した場合（ステップS156：NO）、又は振り方向がヒット方向と一致すると判定した場合（ステップS157：YES）、次優先の衝突対象に対してヒット処理を行う（ステップS159）。このステップS159におけるヒット処理は、ヒット処理の対象が最優先の衝突対象ではなく次優先の衝突対象であることを除いて、ステップS138の処理と同様に行われる。

30

【0175】

ステップS159の処理を行った後、CPU10は、ヒット処理を行うか或いは弾き処理を行うかが決定されていない衝突対象があるか否かを判定する（ステップS160）。処理が未定である衝突対象があるとCPU10によって判定された場合（ステップS160：YES）、処理がステップS156へ戻される。すなわち、ステップS155の処理でヒット処理された衝突対象、及びステップS159の処理でヒット処理された衝突対象以外の衝突対象が複数の衝突対象の中に含まれている場合に、処理がステップS156へ戻される。

40

【0176】

処理が未定の衝突対象がないとCPU10によって判定された場合（ステップS160：NO）、処理が上述したステップS126へ進められる。この場合、最優先の衝突対象及び次優先の衝突対象に対してヒット処理が行われる。このため、図25に例示されるようなアニメーションが再生されることになる。ここで、図25は、敵オブジェクト106が最優先の衝突対象に設定され、敵オブジェクト106の隣に位置する敵オブジェクト106が次に優先順位が高い衝突対象に設定され、更にその隣に位置する敵オブジェクト102が最も優先順位が低い衝突対象に設定された場合に液晶テレビ2に表示されるアニメ

50

ーションの一例を示す画面図である。ここで、2体の敵オブジェクト106は指向性を有しておらず、また、第1コントローラ7の振り方向が敵オブジェクト102のヒット方向と一致している。このため、図25に例示されるように、剣オブジェクト104が振り抜かれて、2体の敵オブジェクト106、及び敵オブジェクト102に対してヒット処理が行われた様子が表示される。

【0177】

一方、ステップS160の処理で「YES」と判定されてステップS156へ処理が戻され、更に優先順位が最も低い衝突対象に対してステップS158の弾き処理が行われた場合、図26に例示されるようなアニメーションが再生されることになる。ここで、図26は、敵オブジェクト106が最優先の衝突対象に設定され、その敵オブジェクト106の隣に位置する敵オブジェクト106が次に優先順位が高い衝突対象に設定され、更にその隣に位置する敵オブジェクト102が最も優先順位が低い衝突対象に設定された場合に液晶テレビ2に表示されるアニメーションの一例を示す画面図である。ここで、2体の敵オブジェクト106は指向性を有しておらず、また、第1コントローラ7の振り方向が敵オブジェクト102のヒット方向と一致していない。このため、図26に例示されるように、剣オブジェクト104が2体の敵オブジェクト106にヒットした後に敵オブジェクト102によって弾き返される様子が表示される。

【0178】

なお、以上説明した当たり処理の変形例は、例えば剣オブジェクト104が最後まで振り切られる様子を表示するためのアニメーションデータ、振り動作を開始した剣オブジェクト104が最初の衝突対象の所で止まる様子を表示するためのアニメーションデータ、振り動作を開始した剣オブジェクト104が2番目の衝突対象の所で止まる様子を表示するためのアニメーションデータ、振り動作を開始した剣オブジェクト104が3番目の衝突対象の所で止まる様子を表示するためのアニメーションデータ、及び剣オブジェクト104が弾き返される様子を表示するためのアニメーションデータの5種類のアニメーションデータを用意することによって実現可能である。

【0179】

[その他の変形例]

本発明は上記実施形態に限定されるものではなく、例えば以下の形態であってもよい。

すなわち、上記実施形態では、剣オブジェクト104とノンプレイヤーオブジェクトとの位置関係を考慮して衝突対象に対してヒット処理を行うか否かを決定する場合について説明したが、オブジェクト同士の位置関係を考慮せず、第1コントローラ7の振り方向のみに基づいて、衝突対象に対してヒット処理を行うか否かを決定するようにしてもよい。

【0180】

また、上記実施形態では、剣オブジェクト104が複数の衝突対象と衝突する可能性がある場合に優先順位を設定して優先順位が最も高い衝突対象に対してのみヒット処理を行うか否かを決定する場合について説明した。また、上記変形例では、複数の衝突対象に優先順位を設定して、優先順位の高い衝突対象にヒット処理を行うと決定した場合に、次に優先順位が高い衝突対象にヒット処理を行うか否かを決定する場合について説明した。これらに代えて、優先順位を設定せずに、複数の衝突対象の全てに対して、ヒット処理を行うか否かを決定するようにしてもよい。

【0181】

また、上記実施形態では、第1コントローラ7の振り操作が開始されたとき(角速度データ1271が示す角速度が所定値以上になったとき)に、第1コントローラ7の振り方向を判別する場合について説明した。これに代えて、剣オブジェクト104が衝突対象としての敵オブジェクトと衝突したと判定したタイミングで、第1コントローラ7の振り方向を判別するようにしてもよい。具体的には、図16におけるステップS102の処理を、例えばステップS124(図17参照)で「YES」と判定されたときに行うようにしてもよい。

【0182】

10

20

30

40

50

また、上記実施形態では、第1コントローラ7の振り操作が開始されたとき（角速度データ1271が示す角速度が所定値以上になったとき）に、剣オブジェクト104の移動軌跡を算出して、敵オブジェクトとの衝突を判定する場合について説明した。これに代えて、第1コントローラ7の振り操作が開始された後も角速度データを毎フレーム取得し、取得した角速度データに基づいてリアルタイムに剣オブジェクト104の移動（動作）を制御するようにしてもよい。この場合、取得した角速度データに基づいて算出された剣オブジェクト104の姿勢及び位置と敵オブジェクトの位置とによって衝突が判定される。

【0183】

また、上記実施形態では、衝突対象が複数あると判定された後に優先順位を設定する場合について説明したが、優先順位の設定は、ステップS123（図17参照）の衝突判定に先立って行ってもよい。

10

【0184】

また、指向性データ135の他の例として、衝突判定用の判定領域のサイズを示すデータを用いるようにしてもよい。具体的には、構え処理（図15参照）のときや、ステップS123の衝突判定前の任意のタイミングで第1コントローラ7の振り方向を判別して、判別した振り方向に応じて、衝突対象である敵オブジェクトに対して設定されている衝突判定用の判定領域のサイズを変化させる。例えば、プレイヤーが第1コントローラ7を右から左へ振ったときには衝突対象の判定領域のサイズを相対的に大きくし、逆にプレイヤーが第1コントローラ7を左から右へ振ったときには衝突対象の判定領域のサイズを相対的に小さくする。このようにすれば、プレイヤーが第1コントローラ7を右から左へ振ったときには衝突対象の判定領域のサイズが相対的に大きいため衝突しやすく、逆にプレイヤーが第1コントローラ7を左から右へ振ったときには衝突対象の判定領域のサイズが相対的に小さいために衝突しにくくなり、ヒット可能な方向を示すデータを用いた場合と同様の効果が得られる。

20

【0185】

また、上記実施形態では、3軸周りの角速度を検出するために、2軸ジャイロセンサ96と1軸ジャイロセンサ97とを用いる構成としたが、3軸周りの角速度を検出することができればよく、用いるジャイロセンサの数及び組み合わせはどのようなものであっても構わない。

【0186】

また、上記実施形態では、各ジャイロセンサ96及び97が角速度を検出する3つの軸は、加速度センサ701が加速度を検出する3つの軸（XYZ軸）と一致するように設定されていたが、各ジャイロセンサ96及び97が角速度を検出する3つの軸と、加速度センサ701が加速度を検出する3つの軸とは一致していなくてもよい。

30

【0187】

また、第1コントローラ7にジャイロセンサユニット9を装着せずに、第1コントローラ7に第2コントローラ8を直接装着してもよい。この場合、ジャイロセンサユニット9から出力される角速度データに代えて、加速度センサ701から出力される加速度データに基づいて第1コントローラ7の位置及び姿勢を算出するようにすればよい。

【0188】

また、上記実施形態では、剣オブジェクト104を持つプレイヤーオブジェクト101を含まないアニメーションが表示される場合について説明したが、剣オブジェクト104及びプレイヤーオブジェクト101を含むアニメーションを表示してもよい。また、剣オブジェクト104及びプレイヤーオブジェクト101を含まず、剣オブジェクト104の移動軌跡のみを示すアニメーションを表示するようにしてもよい。すなわち、第1オブジェクト（本実施形態では剣オブジェクト104）を、液晶テレビ2に表示しないようにしてもよい。

40

【0189】

また、上記実施形態では、衝突判定用の判定領域としての衝突対象が1体の敵オブジェクトに対して1つ設定されている場合について説明したが、衝突対象は、1体の敵オブジ

50

ェクトに対して相異なる位置に複数設定されてもよい。例えば、敵オブジェクトの中心から上下左右の異なる位置にそれぞれ4つの衝突対象を設定してもよい。

【0190】

また、上記実施形態では、上記のようなゲーム処理を1台のゲーム装置3で実現しているが、これは必須ではなく、複数の情報処理装置が協働することによって上記のようなゲーム処理を実現してもよい。すなわち、方向判別手段、衝突判定手段、及び衝突処理決定手段のうち少なくとも1つの手段の機能を、ゲーム装置3以外の例えばネットワーク上のサーバ装置が実現するようにしてもよい。この場合、ゲーム装置3及びサーバ装置からなるゲームシステムが、上述したゲーム装置3と同様に機能する。

【産業上の利用可能性】

10

【0191】

本発明は、仮想ゲーム空間内のオブジェクト同士の衝突判定を行い、その判定結果に応じてゲーム処理を行うゲーム装置のコンピュータに実行させるゲームプログラム、ゲーム装置、ゲームシステム、及びゲーム処理方法等に適用可能である。

【符号の説明】

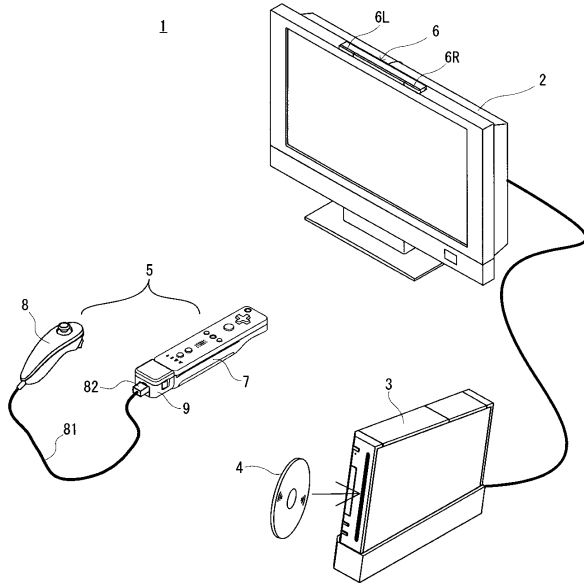
【0192】

- 1 ゲームシステム
- 3 ゲーム装置
- 4 光ディスク
- 5 コントローラ
- 7 第1コントローラ(入力装置)
- 8 第2コントローラ
- 9 ジャイロセンサユニット
- 10 CPU
- 12 外部メインメモリ
- 96 2軸ジャイロセンサ(角速度センサ)
- 97 1軸ジャイロセンサ(角速度センサ)
- 104 剣オブジェクト(第1オブジェクト)
- 102 敵オブジェクト(判定対象)
- 106 敵オブジェクト(判定対象)

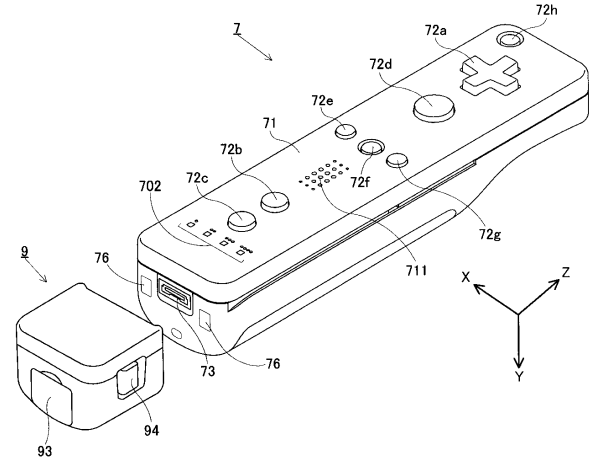
20

30

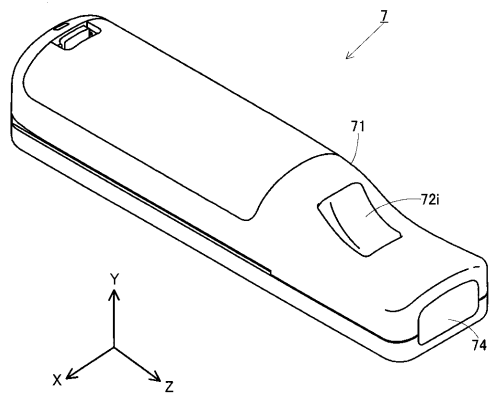
【図1】



【図2】

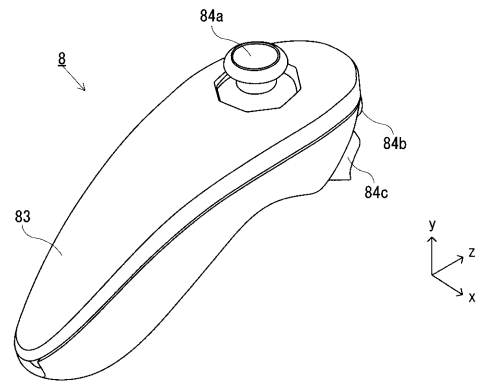


【図3】

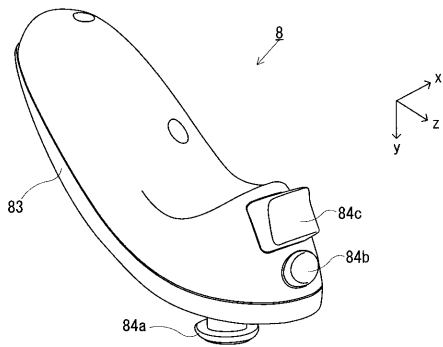


【図4】

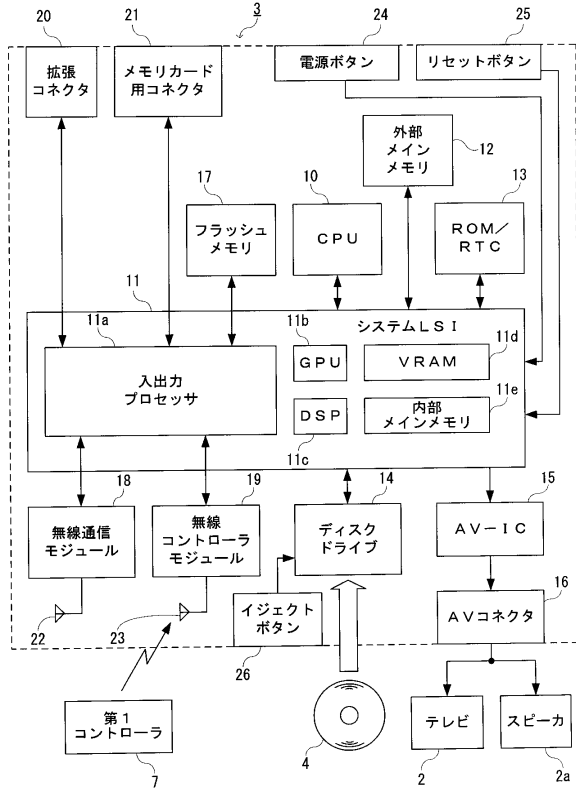
(A)



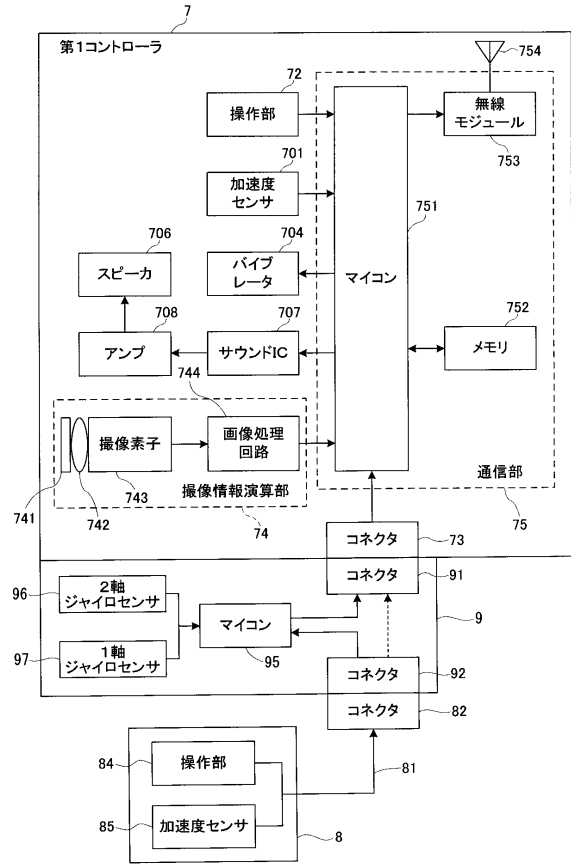
(B)



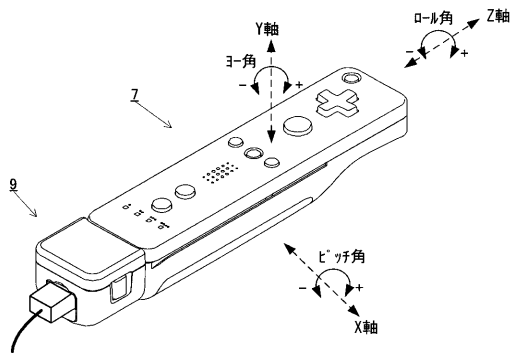
【図5】



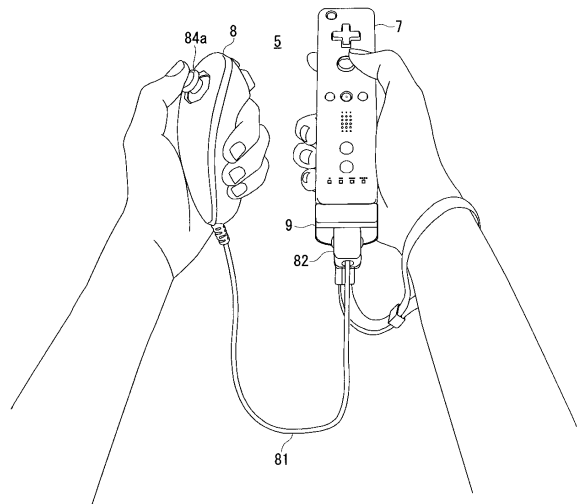
【図6】



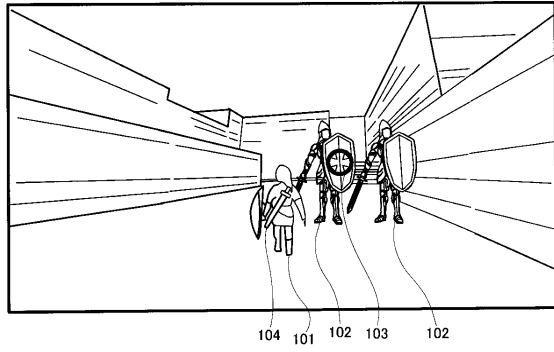
【図7】



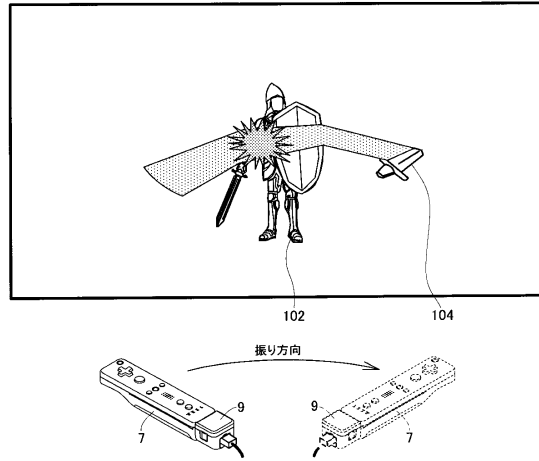
【図8】



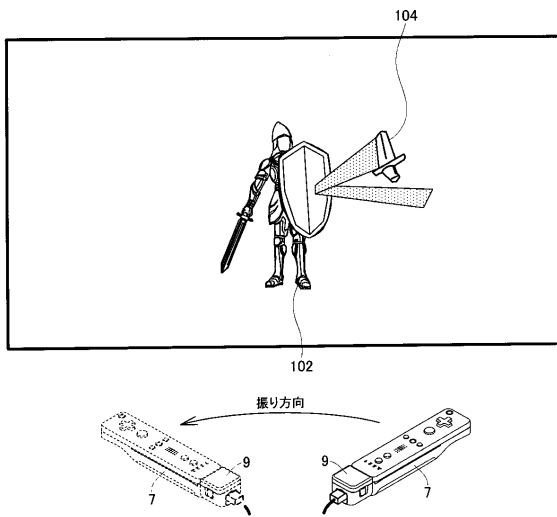
【図 9】



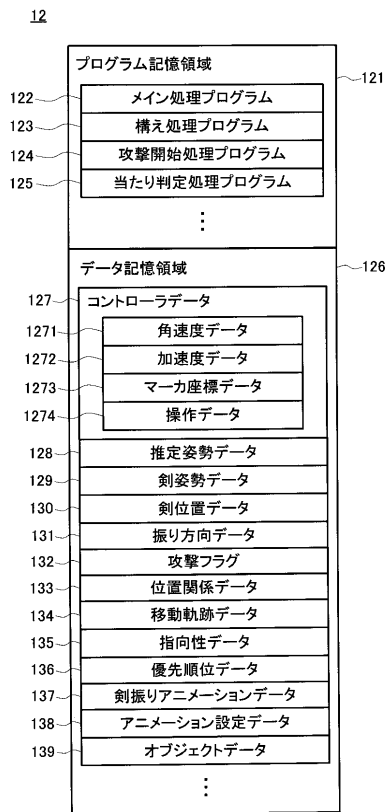
【図 10】



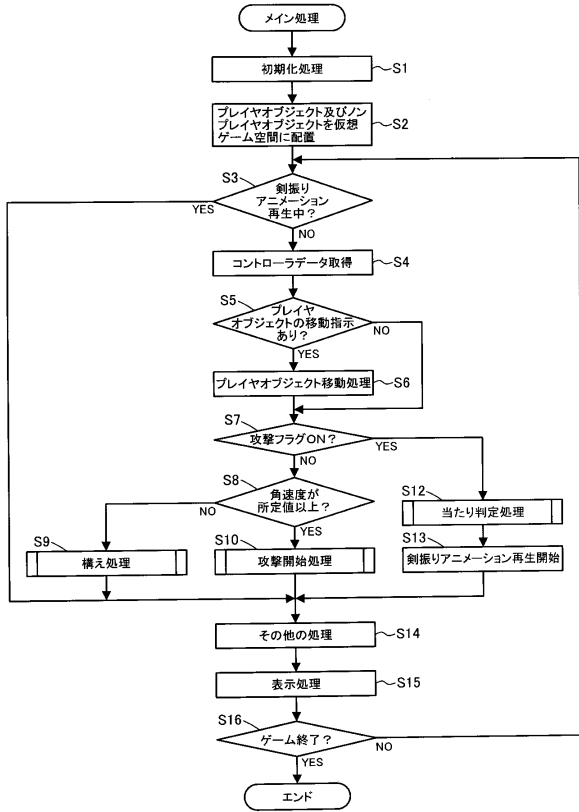
【図 11】



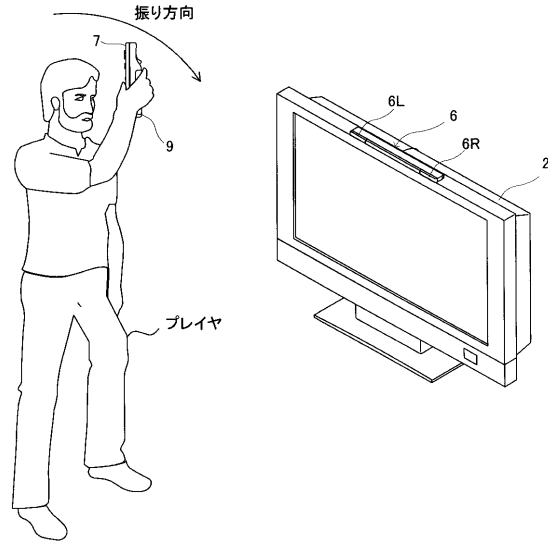
【図 12】



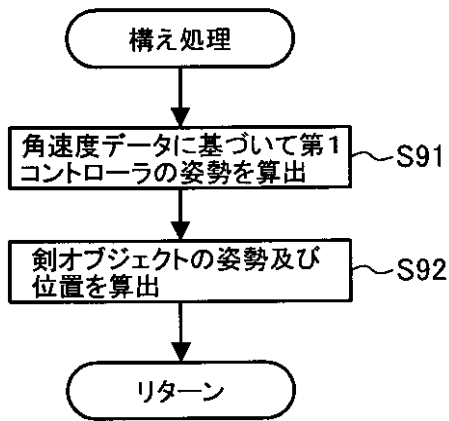
【図13】



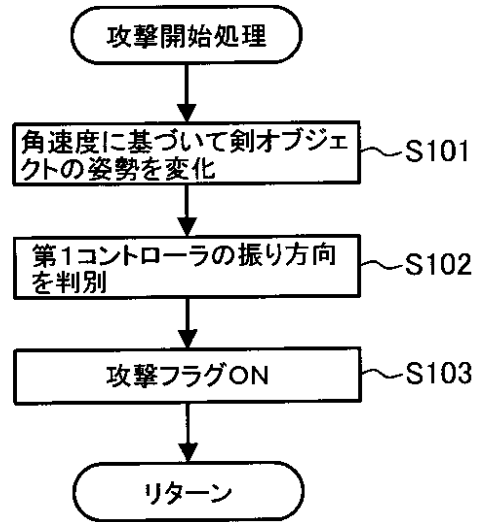
【図14】



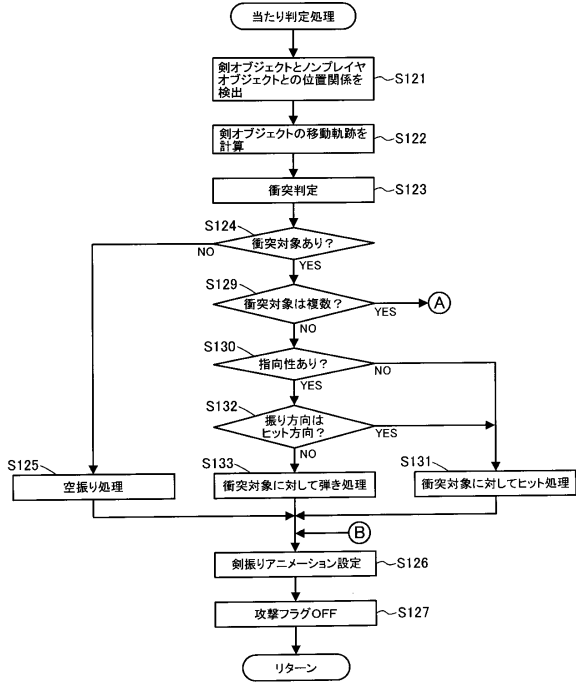
【図15】



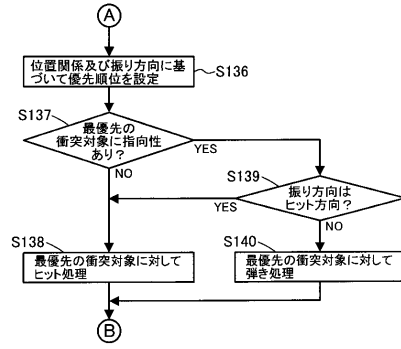
【図16】



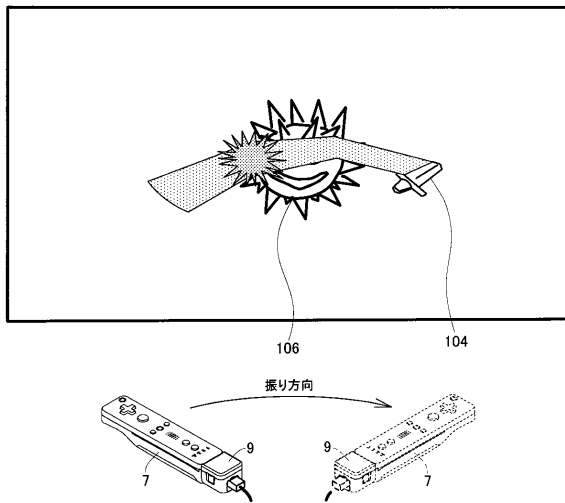
【図17】



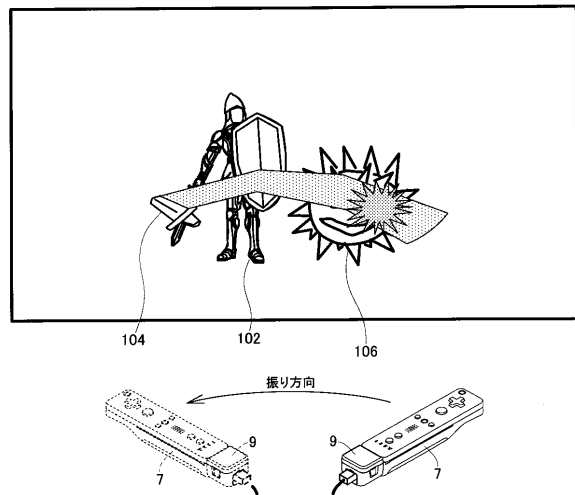
【図18】



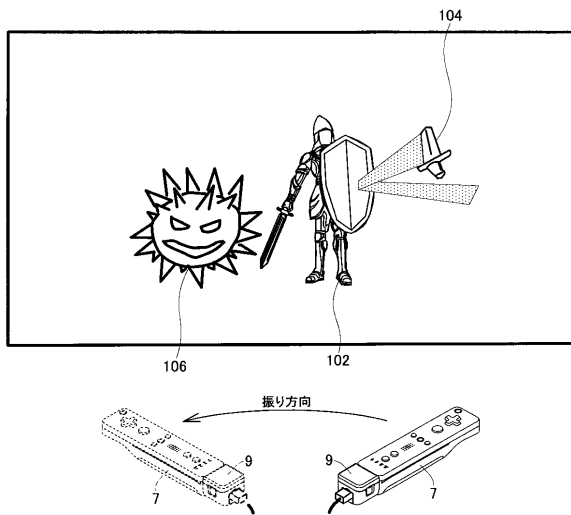
【図19】



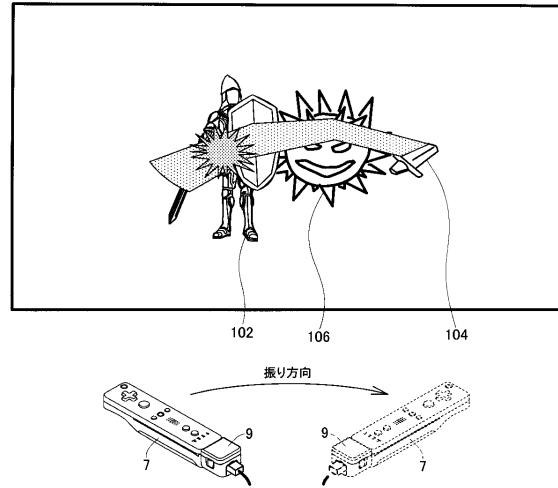
【図20】



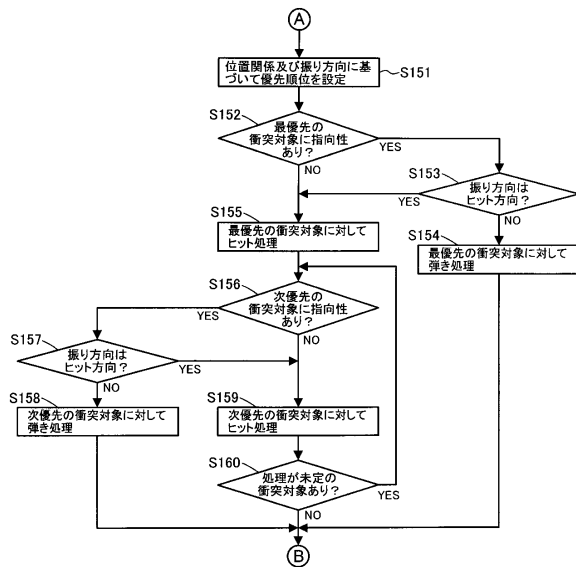
【図 2 1】



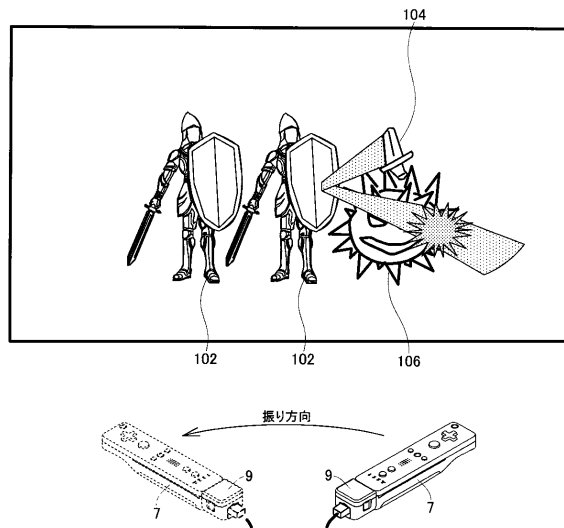
【図 2 2】



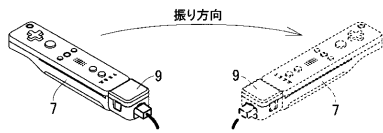
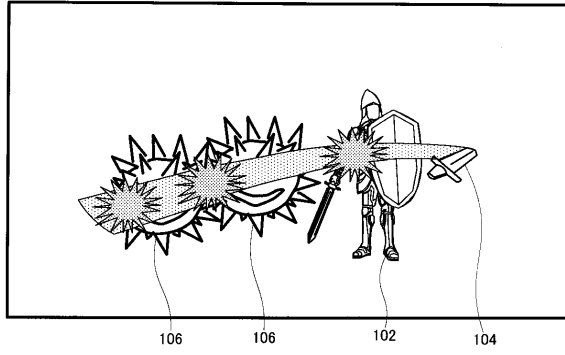
【図 2 3】



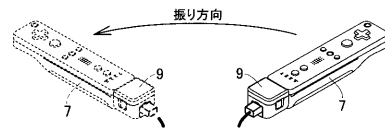
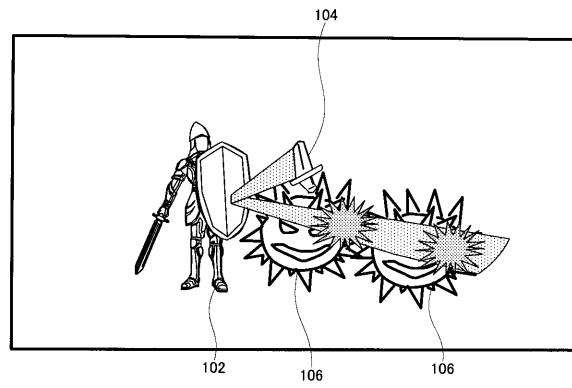
【図 2 4】



【図 25】



【図 26】



フロントページの続き

- (72)発明者 山田 洋一
京都府京都市南区上鳥羽鉾立町11番地1 任天堂株式会社内
- (72)発明者 藤林 秀磨
京都府京都市南区上鳥羽鉾立町11番地1 任天堂株式会社内

審査官 坪内 優佳

- (56)参考文献 特開2008-136695(JP,A)
特開2005-211242(JP,A)
特開2010-017389(JP,A)
特開2003-085585(JP,A)
特開2007-190069(JP,A)
特開2010-167223(JP,A)
ギルティギア 易鳥, 月刊アルカディア, 株式会社エンターブレイン, 2004年 2月 1日
 , 第5巻/第2号, p.22-31, 特に、p.30「攻撃の当たり方」の欄を参照。

- (58)調査した分野(Int.Cl., DB名)
A63F 13/00 - 13/98
A63F 9/24