

(19) 日本国特許庁(JP)

## (12) 公開特許公報(A)

(11) 特許出願公開番号

特開2018-181306

(P2018-181306A)

(43) 公開日 平成30年11月15日(2018.11.15)

(51) Int.Cl.	F 1	テーマコード (参考)
<b>G06T 19/00</b> (2011.01)	G06T 19/00	C 5B050
<b>G06F 3/01</b> (2006.01)	G06F 3/01	570 5E555
<b>G06F 3/0484</b> (2013.01)	G06F 3/0484	170
<b>A63F 13/25</b> (2014.01)	G06T 19/00	300B
<b>A63F 13/428</b> (2014.01)	A63F 13/25	

審査請求 未請求 請求項の数 9 O L (全 36 頁) 最終頁に続く

(21) 出願番号	特願2018-20528 (P2018-20528)	(71) 出願人	509070463
(22) 出願日	平成30年2月7日 (2018.2.7)	株式会社コロプラ	
(62) 分割の表示	特願2017-80894 (P2017-80894)	東京都渋谷区恵比寿四丁目20番3号	
原出願日	の分割 平成29年4月14日 (2017.4.14)	(72) 発明者	柏原 崇生 東京都渋谷区恵比寿四丁目20番3号 株式会社コロプラ内

F ターム (参考) 5B050 AA10 BA08 BA13 CA07 EA07  
 EA24 EA27 FA02  
 5E555 AA76 BA20 BA83 BB20 BC08  
 BE17 CB66 CC26 DA08 DB53  
 DC19 EA14 FA00

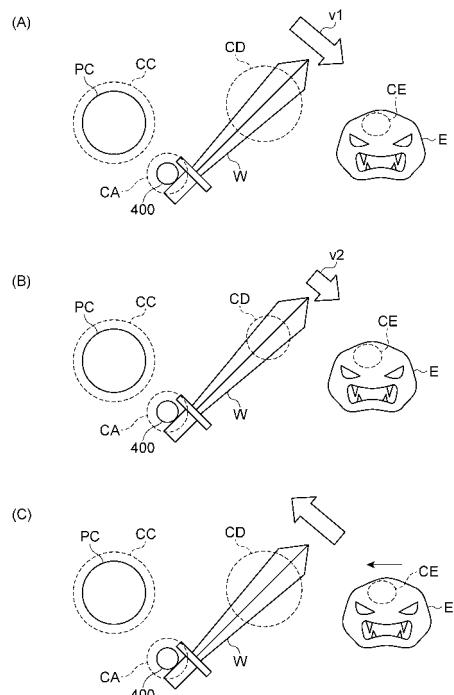
(54) 【発明の名称】情報処理方法、装置、および当該情報処理方法をコンピュータに実行させるためのプログラム

## (57) 【要約】

【課題】ユーザの仮想体験のエンタテイメント性向上させ得る。

【解決手段】一実施形態の情報処理方法は、ユーザの手の動きに応じて、操作オブジェクトを動かすステップと、操作オブジェクトの移動速度に基づいて、操作オブジェクトに関連付けられるコリジョン効果を決定するステップと、操作オブジェクトと対象オブジェクトとが衝突した場合に、操作オブジェクトと対象オブジェクトとの関係に応じた処理を実行するステップと、ヘッドマウントデバイスの動きと仮想空間データとに基づいて視界画像を生成し、表示部に視界画像を表示させるステップと、を含む。

【選択図】図13



**【特許請求の範囲】****【請求項 1】**

表示部を備えるヘッドマウントデバイスを介してユーザに仮想体験を提供するためにコンピュータによって実行される情報処理方法であって、

前記ユーザの手に関連付けられた手オブジェクトおよび前記手オブジェクトにより操作されるオブジェクトの少なくとも一方を含む操作オブジェクトと、前記操作オブジェクトと相互に作用する対象オブジェクトと、を含む仮想空間を規定する仮想空間データを生成するステップと、

前記ヘッドマウントデバイスの動きと前記ユーザの手の動きとを検出するステップと、前記ユーザの手の動きに応じて、前記操作オブジェクトを動かすステップと、

前記操作オブジェクトの移動速度に基づいて、前記操作オブジェクトに関連付けられるコリジョン効果を決定するステップと、

前記操作オブジェクトと前記対象オブジェクトとが衝突した場合に、前記操作オブジェクトと前記対象オブジェクトとの関係に応じた処理を実行するステップと、

前記ヘッドマウントデバイスの動きと前記仮想空間データとに基づいて視界画像を生成し、前記表示部に前記視界画像を表示させるステップと、  
を含む、情報処理方法。

**【請求項 2】**

前記コリジョン効果は、前記操作オブジェクトに関連付けられるコリジョンエリアのサイズ、または、前記コリジョンエリアに基づいて前記操作オブジェクトと前記対象オブジェクトとが衝突したと判定された場合に生じさせる所定のパラメータの変動量である、  
請求項 1 に記載の情報処理方法。

**【請求項 3】**

前記操作オブジェクトの移動速度が大きい程、前記コリジョンエリアのサイズを大きくまたは小さくする、  
請求項 2 に記載の情報処理方法。

**【請求項 4】**

前記所定のパラメータは、前記対象オブジェクトに関連付けられたパラメータであり、  
前記操作オブジェクトの移動速度が大きい程、前記所定のパラメータの変動量を大きくまたは小さくする、  
請求項 2 に記載の情報処理方法。

**【請求項 5】**

前記所定のパラメータは、前記ユーザに関連付けられたパラメータであり、  
前記操作オブジェクトの移動速度が大きい程、前記所定のパラメータの変動量を小さくまたは大きくする、  
請求項 2 に記載の情報処理方法。

**【請求項 6】**

前記操作オブジェクトの移動速度が予め定められた閾値以下の場合には、前記コリジョン効果を無効化する、  
請求項 1 ~ 5 のいずれか一項に記載の情報処理方法。

**【請求項 7】**

前記対象オブジェクトに対する前記操作オブジェクトの移動方向に基づいて、前記コリジョン効果を決定するステップをさらに含む、  
請求項 1 ~ 5 のいずれか一項に記載の情報処理方法。

**【請求項 8】**

請求項 1 ~ 7 のいずれか一項に記載の情報処理方法をコンピュータに実行させる、プログラム。

**【請求項 9】**

少なくともメモリと、前記メモリに結合されたプロセッサとを備え、前記プロセッサの制御により請求項 1 ~ 7 のいずれか一項に記載の情報処理方法を実行する、装置。

10

20

30

40

50

**【発明の詳細な説明】****【技術分野】****【0001】**

本開示は、情報処理方法、装置、および当該情報処理方法をコンピュータに実行させるためのプログラムに関する。

**【背景技術】****【0002】**

特許文献1は、仮想空間において、ユーザの身体的な動きに連動させて剣等のオブジェクトを操作して敵と戦うバトルゲームを提供するための技術を開示している。

**【先行技術文献】**

10

**【特許文献】****【0003】****【特許文献1】特許第5996138号公報****【発明の概要】****【発明が解決しようとする課題】****【0004】**

しかしながら、特許文献1に開示された技術には、より直感的な仮想体験をユーザに提供する上で改善の余地がある。例えば、仮想空間においてユーザが操作する操作オブジェクトと他のオブジェクト(例えば敵キャラクタ)との間の相互作用に関して、ユーザの仮想体験のエンタテイメント性をさらに向上させる余地がある。

20

**【0005】**

そこで、本開示は、ユーザの仮想体験のエンタテイメント性を向上させ得る情報処理方法、装置、および当該情報処理方法をコンピュータに実行させるためのプログラムを提供することを目的とする。

**【課題を解決するための手段】****【0006】**

本開示が示す一態様によれば、表示部を備えるヘッドマウントデバイスを介してユーザに仮想体験を提供するためにコンピュータによって実行される情報処理方法が提供される。この情報処理方法は、ユーザの手に関連付けられた手オブジェクトおよび手オブジェクトにより操作されるオブジェクトの少なくとも一方を含む操作オブジェクトと、操作オブジェクトと相互に作用する対象オブジェクトと、を含む仮想空間を規定する仮想空間データを生成するステップと、ヘッドマウントデバイスの動きとユーザの手の動きとを検出するステップと、ユーザの手の動きに応じて、操作オブジェクトを動かすステップと、操作オブジェクトの移動速度に基づいて、操作オブジェクトに関連付けられるコリジョン効果を決定するステップと、操作オブジェクトと対象オブジェクトとが衝突した場合に、操作オブジェクトと対象オブジェクトとの関係に応じた処理を実行するステップと、ヘッドマウントデバイスの動きと仮想空間データとに基づいて視界画像を生成し、表示部に視界画像を表示させるステップと、を含む。

30

**【発明の効果】****【0007】**

本開示によれば、ユーザの仮想体験のエンタテイメント性を向上させ得る情報処理方法、装置、および当該情報処理方法をコンピュータに実行させるためのプログラムを提供することが可能となる。

40

**【図面の簡単な説明】****【0008】****【図1】ある実施の形態に従うHMDシステム100の構成の概略を表す図である。**

【図2】一局面に従うコンピュータ200のハードウェア構成の一例を表すブロック図である。

【図3】ある実施の形態に従うHMD装置110に設定されるuvw視野座標系を概念的に表す図である。

50

【図4】ある実施の形態に従う仮想空間2を表現する一態様を概念的に表す図である。

【図5】ある実施の形態に従うHMD装置110を装着するユーザ190の頭部を上から表した図である。

【図6】仮想空間2において視界領域23をX方向から見たYZ断面を表す図である。

【図7】仮想空間2において視界領域23をY方向から見たXZ断面を表す図である。

【図8】ある実施の形態に従うコントローラ160の概略構成を表す図である。

【図9】ある実施の形態に従うコンピュータ200をモジュール構成として表すブロック図である。

【図10】状態(A)は、HMD装置110とコントローラ160とを装着したユーザ190を示す図である。状態(B)は、仮想カメラ1と手オブジェクト400と対象オブジェクト500とを含む仮想空間2を示す図である。 10

【図11】HMDシステム100が実行する処理を表すフローチャートである。

【図12】図11のステップS9の処理の例を表すフローチャートである。

【図13】武器オブジェクトWと敵オブジェクトEとの間のコリジョン制御を説明するための図である。

【図14】図11のステップS9の処理の例を表すフローチャートである。

【図15】武器オブジェクトWと攻撃オブジェクトAとの間のコリジョン制御を説明するための図である。

【図16】ユーザ190の仮想空間2におけるアクションを評価するための処理の例を表すフローチャートである。 20

【図17】状態(A)は、第3の判定例を説明するための図である。状態(B)は、第7の判定例を説明するための図である。

【図18】第5の判定例を説明するための図である。

【図19】第8の判定例を説明するための図である。

【発明を実施するための形態】

【0009】

以下、図面を参照しつつ、本開示の実施の形態について説明する。以下の説明では、同一の部品には同一の符号を付してある。それらの名称および機能も同じである。したがって、それらについての詳細な説明は繰り返さない。

【0010】 30

[HMDシステムの構成]

図1を参照して、HMD(Head Mount Device)システム100の構成について説明する。図1は、ある実施の形態に従うHMDシステム100の構成の概略を表す図である。ある局面において、HMDシステム100は、家庭用のシステムとしてあるいは業務用のシステムとして提供される。

【0011】

HMDシステム100は、HMD装置110(ヘッドマウントデバイス)と、HMDセンサ120と、コントローラ160と、コンピュータ200とを備える。HMD装置110は、ディスプレイ112(表示部)と、注視センサ140とを含む。コントローラ160は、モーションセンサ130を含み得る。 40

【0012】

ある局面において、コンピュータ200は、インターネットその他のネットワーク19に接続可能であり、ネットワーク19に接続されているサーバ150その他のコンピュータと通信可能である。別の局面において、HMD装置110は、HMDセンサ120の代わりに、センサ114を含み得る。

【0013】

HMD装置110は、ユーザの頭部に装着され、動作中に仮想空間をユーザに提供し得る。より具体的には、HMD装置110は、右目用の画像および左目用の画像をディスプレイ112にそれぞれ表示する。ユーザの各目がそれぞれの画像を視認すると、ユーザは、両目の視差に基づき当該画像を3次元の画像として認識し得る。 50

## 【0014】

ディスプレイ112は、例えば、非透過型の表示装置として実現される。ある局面において、ディスプレイ112は、ユーザの両目の前方に位置するようにHMD装置110の本体に配置されている。したがって、ユーザは、ディスプレイ112に表示される3次元画像を視認すると、仮想空間に没入することができる。ある実施の形態において、仮想空間は、例えば、背景、ユーザが操作可能なオブジェクト、およびユーザが選択可能なメニューの画像等を含む。ある実施の形態において、ディスプレイ112は、所謂スマートフォンその他の情報表示端末が備える液晶ディスプレイまたは有機EL(Electro Luminescence)ディスプレイとして実現され得る。ディスプレイ112は、HMD装置110の本体と一体に構成されてもよいし、別体として構成されてもよい。

10

## 【0015】

ある局面において、ディスプレイ112は、右目用の画像を表示するためのサブディスプレイと、左目用の画像を表示するためのサブディスプレイとを含み得る。別の局面において、ディスプレイ112は、右目用の画像と左目用の画像とを一体として表示する構成であってもよい。この場合、ディスプレイ112は、高速シャッタを含む。高速シャッタは、画像がいずれか一方の目にのみ認識されるように、右目用の画像と左目用の画像とを交互に表示可能に作動する。

## 【0016】

HMDセンサ120は、複数の光源(図示しない)を含む。各光源は例えば、赤外線を発するLED(Light Emitting Diode)により実現される。HMDセンサ120は、HMD装置110の動きを検出するためのポジショントラッキング機能を有する。HMDセンサ120は、この機能を用いて、現実空間内におけるHMD装置110の位置および傾きを検出する。

20

## 【0017】

なお、別の局面において、HMDセンサ120は、カメラにより実現されてもよい。この場合、HMDセンサ120は、カメラから出力されるHMD装置110の画像情報を用いて、画像解析処理を実行することにより、HMD装置110の位置および傾きを検出することができる。

## 【0018】

別の局面において、HMD装置110は、位置検出器として、HMDセンサ120の代わりに、センサ114を備えてもよい。HMD装置110は、センサ114を用いて、HMD装置110自身の位置および傾きを検出し得る。例えば、センサ114が角速度センサ、地磁気センサ、加速度センサ、あるいはジャイロセンサ等である場合、HMD装置110は、HMDセンサ120の代わりに、これらの各センサのいずれかを用いて、自身の位置および傾きを検出し得る。一例として、センサ114が角速度センサである場合、角速度センサは、現実空間におけるHMD装置110の3軸周りの角速度を経時的に検出する。HMD装置110は、各角速度に基づいて、HMD装置110の3軸周りの角度の時間的变化を算出し、さらに、角度の時間的变化に基づいて、HMD装置110の傾きを算出する。また、HMD装置110は、透過型表示装置を備えていてもよい。この場合、当該透過型表示装置は、その透過率を調整することにより、一時的に非透過型の表示装置として構成可能であってもよい。また、視界画像は仮想空間を構成する画像の一部に、現実空間を提示する構成を含んでいてもよい。例えば、HMD装置110に搭載されたカメラで撮影した画像を視界画像の一部に重畳して表示させてもよいし、当該透過型表示装置の一部の透過率を高く設定することにより、視界画像の一部から現実空間を視認可能にしてよい。

30

## 【0019】

注視センサ140は、ユーザ190の右目および左目の視線が向けられる方向(視線方向)を検出する。当該方向の検出は、例えば、公知のアイトラッキング機能によって実現される。注視センサ140は、当該アイトラッキング機能を有するセンサにより実現される。ある局面において、注視センサ140は、右目用のセンサおよび左目用のセンサを含

40

50

むことが好ましい。注視センサ140は、例えば、ユーザ190の右目および左目に赤外光を照射するとともに、照射光に対する角膜および虹彩からの反射光を受けることにより各眼球の回転角を検出するセンサであってもよい。注視センサ140は、検出した各回転角に基づいて、ユーザ190の視線方向を検知することができる。

#### 【0020】

サーバ150は、コンピュータ200にプログラムを送信し得る。別の局面において、サーバ150は、他のユーザによって使用されるHMD装置に仮想現実を提供するための他のコンピュータ200と通信し得る。例えば、アミューズメント施設において、複数のユーザが参加型のゲームを行う場合、各コンピュータ200は、各ユーザの動作に基づく信号を他のコンピュータ200と通信して、同じ仮想空間において複数のユーザが共通のゲームを楽しむことを可能にする。サーバ150は、一または複数のコンピュータ装置により構成され得る。サーバ150は、後述するコンピュータ200のハードウェア構成と同様のハードウェア構成（プロセッサ、メモリ、ストレージ等）を備え得る。

10

#### 【0021】

コントローラ160は、ユーザ190からコンピュータ200への命令の入力を受け付ける。ある局面において、コントローラ160は、ユーザ190によって把持可能に構成される。別の局面において、コントローラ160は、ユーザ190の身体あるいは衣類の一部に装着可能に構成される。別の局面において、コントローラ160は、コンピュータ200から送られる信号に基づいて、振動、音、光のうちの少なくともいずれかを出力するように構成されてもよい。別の局面において、コントローラ160は、仮想現実を提供する仮想空間に配置されるオブジェクトの位置および動き等を制御するためにユーザ190によって与えられる操作を受け付ける。

20

#### 【0022】

モーションセンサ130は、ある局面において、ユーザの手に取り付けられて、ユーザの手の動きを検出する。例えば、モーションセンサ130は、手の回転速度、回転数等を検出する。検出された信号は、コンピュータ200に送られる。モーションセンサ130は、例えば、手袋型のコントローラ160に設けられている。ある実施の形態において、現実空間における安全のため、コントローラ160は、手袋型のようによくユーザ190の手に装着されることにより容易に飛んで行かないものに装着されるのが望ましい。別の局面において、ユーザ190に装着されないセンサがユーザ190の手の動きを検出してもよい。例えば、ユーザ190を撮影するカメラの信号が、ユーザ190の動作を表す信号として、コンピュータ200に入力されてもよい。モーションセンサ130とコンピュータ200とは、有線により、または無線により互いに接続される。無線の場合、通信形態は特に限られず、例えば、Blueooth（登録商標）その他の公知の通信手法が用いられる。

30

#### 【0023】

##### 【ハードウェア構成】

図2を参照して、本実施の形態に係るコンピュータ200について説明する。図2は、一局面に従うコンピュータ200のハードウェア構成の一例を表すブロック図である。コンピュータ200は、主たる構成要素として、プロセッサ10と、メモリ11と、ストレージ12と、出入力インターフェース13と、通信インターフェース14とを備える。各構成要素は、それぞれ、バス15に接続されている。

40

#### 【0024】

プロセッサ10は、コンピュータ200に与えられる信号に基づいて、あるいは、予め定められた条件が成立したことに基づいて、メモリ11またはストレージ12に格納されているプログラムに含まれる一連の命令を実行する。ある局面において、プロセッサ10は、CPU(Central Processing Unit)、MPU(Micro Processor Unit)、FPGA(Field-Programmable Gate Array)その他のデバイスとして実現される。

#### 【0025】

メモリ11は、プログラムおよびデータを一時的に保存する。プログラムは、例えば、

50

ストレージ12からロードされる。メモリ11に保存されるデータは、コンピュータ200に入力されたデータと、プロセッサ10によって生成されたデータとを含む。ある局面において、メモリ11は、RAM(Random Access Memory)その他の揮発性メモリとして実現される。

【0026】

ストレージ12は、プログラムおよびデータを永続的に保持する。ストレージ12は、例えば、ROM(Read-Only Memory)、ハードディスク装置、フラッシュメモリ、その他の不揮発性記憶装置として実現される。ストレージ12に格納されるプログラムは、HMDシステム100において仮想空間を提供するためのプログラム、シミュレーションプログラム、ゲームプログラム、ユーザ認証プログラム、および他のコンピュータ200との通信を実現するためのプログラム等を含む。ストレージ12に格納されるデータは、仮想空間を規定するためのデータおよびオブジェクト等を含む。

10

【0027】

なお、別の局面において、ストレージ12は、メモリカードのように着脱可能な記憶装置として実現されてもよい。さらに別の局面において、コンピュータ200に内蔵されたストレージ12の代わりに、外部の記憶装置に保存されているプログラムおよびデータを使用する構成が使用されてもよい。このような構成によれば、例えば、アミューズメント施設のように複数のHMDシステム100が使用される場面において、プログラムおよびデータ等の更新を一括して行うことが可能になる。

20

【0028】

ある実施の形態において、入出力インターフェース13は、HMD装置110、HMDセンサ120またはモーションセンサ130との間で信号を通信する。ある局面において、入出力インターフェース13は、USB(Universal Serial Bus)インターフェース、DVI(Digital Visual Interface)、HDMI(登録商標)(High-Definition Multimedia Interface)その他の端子を用いて実現される。なお、入出力インターフェース13は上述のものに限られない。

20

【0029】

ある実施の形態において、入出力インターフェース13は、さらに、コントローラ160と通信し得る。例えば、入出力インターフェース13は、モーションセンサ130から出力された信号の入力を受ける。別の局面において、入出力インターフェース13は、プロセッサ10から出力された命令を、コントローラ160に送る。当該命令は、振動、音声出力、発光等をコントローラ160に指示する。コントローラ160は、当該命令を受信すると、その命令に応じて、振動、音声出力または発光のいずれかを実行する。

30

【0030】

通信インターフェース14は、ネットワーク19に接続されて、ネットワーク19に接続されている他のコンピュータ(例えば、サーバ150)と通信する。ある局面において、通信インターフェース14は、例えば、LAN(Local Area Network)その他の有線通信インターフェース、あるいは、Wi-Fi(Wireless Fidelity)、Bluetooth(登録商標)、NFC(Near Field Communication)その他の無線通信インターフェースとして実現される。なお、通信インターフェース14は上述のものに限られない。例えば、入出力インターフェース13は、Bluetooth(登録商標)等の無線通信インターフェースを含み得る。

40

【0031】

ある局面において、プロセッサ10は、ストレージ12にアクセスし、ストレージ12に格納されている1つ以上のプログラムをメモリ11にロードし、当該プログラムに含まれる一連の命令を実行する。当該1つ以上のプログラムは、コンピュータ200のオペレーティングシステム、仮想空間を提供するためのアプリケーションプログラム、コントローラ160を用いて仮想空間で実行可能なゲームソフトウェア等を含み得る。プロセッサ10は、入出力インターフェース13を介して、仮想空間を提供するための信号をHMD装置110に送る。HMD装置110は、その信号に基づいてディスプレイ112に映像

50

を表示する。

【0032】

なお、図2に示される例では、コンピュータ200がHMD装置110の外部に設けられる構成が示されているが、別の局面において、コンピュータ200は、HMD装置110に内蔵されてもよい。一例として、ディスプレイ112を含む携帯型の情報通信端末（例えば、スマートフォン）がコンピュータ200として機能してもよい。

【0033】

また、コンピュータ200は、複数のHMD装置110に共通して用いられる構成であってもよい。このような構成によれば、例えば、複数のユーザに同一の仮想空間を提供することもできるので、各ユーザは同一の仮想空間で他のユーザと同一のアプリケーションを楽しむことができる。

10

【0034】

ある実施の形態において、HMDシステム100では、グローバル座標系が予め設定されている。グローバル座標系は、現実空間における鉛直方向、鉛直方向に直交する水平方向、ならびに、鉛直方向および水平方向の双方に直交する前後方向にそれぞれ平行な、3つの基準方向（軸）を有する。本実施の形態では、グローバル座標系は視点座標系の一つである。そこで、グローバル座標系における水平方向、鉛直方向（上下方向）、および前後方向は、それぞれ、x軸、y軸、z軸と規定される。より具体的には、グローバル座標系において、x軸は現実空間の水平方向に平行である。y軸は、現実空間の鉛直方向に平行である。z軸は現実空間の前後方向に平行である。

20

【0035】

ある局面において、HMDセンサ120は、赤外線センサを含む。赤外線センサが、HMD装置110の各光源から発せられた赤外線をそれぞれ検出すると、HMD装置110の存在を検出する。HMDセンサ120は、さらに、各点の値（グローバル座標系における各座標値）に基づいて、HMD装置110を装着したユーザ190の動きに応じた、現実空間内におけるHMD装置110の位置および傾きを検出する。より詳しくは、HMDセンサ120は、経時的に検出された各値を用いて、HMD装置110の位置および傾きの時間的变化を検出できる。

【0036】

グローバル座標系は現実空間の座標系と平行である。したがって、HMDセンサ120によって検出されたHMD装置110の各傾きは、グローバル座標系におけるHMD装置110の3軸周りの各傾きに相当する。HMDセンサ120は、グローバル座標系におけるHMD装置110の傾きに基づき、uvw視野座標系をHMD装置110に設定する。HMD装置110に設定されるuvw視野座標系は、HMD装置110を装着したユーザ190が仮想空間において物体を見る際の視点座標系に対応する。

30

【0037】

[uvw視野座標系]

図3を参照して、uvw視野座標系について説明する。図3は、ある実施の形態に従うHMD装置110に設定されるuvw視野座標系を概念的に表す図である。HMDセンサ120は、HMD装置110の起動時に、グローバル座標系におけるHMD装置110の位置および傾きを検出する。プロセッサ10は、検出された値に基づいて、uvw視野座標系をHMD装置110に設定する。

40

【0038】

図3に示されるように、HMD装置110は、HMD装置110を装着したユーザの頭部を中心（原点）とした3次元のuvw視野座標系を設定する。より具体的には、HMD装置110は、グローバル座標系を規定する水平方向、鉛直方向、および前後方向（x軸、y軸、z軸）を、グローバル座標系内においてHMD装置110の各軸周りの傾きだけ各軸周りにそれぞれ傾けることによって新たに得られる3つの方向を、HMD装置110におけるuvw視野座標系のピッチ方向（u軸）、ヨー方向（v軸）、およびロール方向（w軸）として設定する。

50

## 【0039】

ある局面において、HMD装置110を装着したユーザ190が直立し、かつ、正面を視認している場合、プロセッサ10は、グローバル座標系に平行なuvw視野座標系をHMD装置110に設定する。この場合、グローバル座標系における水平方向(x軸)、鉛直方向(y軸)、および前後方向(z軸)は、HMD装置110におけるuvw視野座標系のピッチ方向(u軸)、ヨー方向(v軸)、およびロール方向(w軸)に一致する。

## 【0040】

uvw視野座標系がHMD装置110に設定された後、HMDセンサ120は、HMD装置110の動きに基づいて、設定されたuvw視野座標系におけるHMD装置110の傾き(傾きの変化量)を検出できる。この場合、HMDセンサ120は、HMD装置110の傾きとして、uvw視野座標系におけるHMD装置110のピッチ角( $u$ )、ヨー角( $v$ )、およびロール角( $w$ )をそれぞれ検出する。ピッチ角( $u$ )は、uvw視野座標系におけるピッチ方向周りのHMD装置110の傾き角度を表す。ヨー角( $v$ )は、uvw視野座標系におけるヨー方向周りのHMD装置110の傾き角度を表す。ロール角( $w$ )は、uvw視野座標系におけるロール方向周りのHMD装置110の傾き角度を表す。

10

## 【0041】

HMDセンサ120は、検出されたHMD装置110の傾き角度に基づいて、HMD装置110が動いた後のHMD装置110におけるuvw視野座標系を、HMD装置110に設定する。HMD装置110と、HMD装置110のuvw視野座標系との関係は、HMD装置110の位置および傾きに関わらず、常に一定である。HMD装置110の位置および傾きが変わると、当該位置および傾きの変化に連動して、グローバル座標系におけるHMD装置110のuvw視野座標系の位置および傾きが変化する。

20

## 【0042】

ある局面において、HMDセンサ120は、赤外線センサからの出力に基づいて取得される赤外線の光強度および複数の点間の相対的な位置関係(例えば、各点間の距離など)に基づいて、HMD装置110の現実空間内における位置を、HMDセンサ120に対する相対位置として特定してもよい。また、プロセッサ10は、特定された相対位置に基づいて、現実空間内(グローバル座標系)におけるHMD装置110のuvw視野座標系の原点を決定してもよい。

30

## 【0043】

## [仮想空間]

図4を参照して、仮想空間についてさらに説明する。図4は、ある実施の形態に従う仮想空間2を表現する一態様を概念的に表す図である。仮想空間2は、中心21の360度方向の全体を覆う全天球状の構造を有する。図4では、説明を複雑にしないために、仮想空間2のうちの上半分の天球が例示されている。仮想空間2では各メッシュが規定される。各メッシュの位置は、仮想空間2に規定されるXYZ座標系における座標値として予め規定されている。コンピュータ200は、仮想空間2に展開可能なコンテンツ(静止画、動画等)を構成する各部分画像を、仮想空間2において対応する各メッシュにそれぞれ対応付けて、ユーザによって視認可能な仮想空間画像22が展開される仮想空間2をユーザに提供する。

40

## 【0044】

ある局面において、仮想空間2では、中心21を原点とするXYZ座標系が規定される。XYZ座標系は、例えば、グローバル座標系に平行である。XYZ座標系は視点座標系の一種であるため、XYZ座標系における水平方向、鉛直方向(上下方向)、および前後方向は、それぞれX軸、Y軸、Z軸として規定される。したがって、XYZ座標系のX軸(水平方向)がグローバル座標系のx軸と平行であり、XYZ座標系のY軸(鉛直方向)がグローバル座標系のy軸と平行であり、XYZ座標系のZ軸(前後方向)がグローバル座標系のz軸と平行である。仮想空間2内の各位置は、XYZ座標系における座標値によって一意に特定される。

50

## 【0045】

HMD装置110の起動時、すなわちHMD装置110の初期状態において、仮想カメラ1は、例えば仮想空間2の中心21に配置される。仮想カメラ1は、現実空間におけるHMD装置110の動きに連動して、仮想空間2を同様に移動する。これにより、現実空間におけるHMD装置110の位置および傾きの変化が、仮想空間2において同様に再現される。

## 【0046】

仮想カメラ1には、HMD装置110の場合と同様に、uvw視野座標系が規定される。仮想空間2における仮想カメラ1のuvw視野座標系は、現実空間（グローバル座標系）におけるHMD装置110のuvw視野座標系に連動するように規定されている。したがって、HMD装置110の傾きが変化すると、それに応じて、仮想カメラ1の傾きも変化する。また、仮想カメラ1は、HMD装置110を装着したユーザの現実空間における移動に連動して、仮想空間2において移動することもできる。

10

## 【0047】

仮想カメラ1の向きは、仮想カメラ1の位置および傾きに応じて決まるので、ユーザが仮想空間画像22を視認する際に基準となる視線（基準視線5）は、仮想カメラ1の向きに応じて決まる。コンピュータ200のプロセッサ10は、基準視線5に基づいて、仮想空間2における視界領域23を規定する。視界領域23は、仮想空間2のうち、HMD装置110を装着したユーザの視界に対応する。

20

## 【0048】

注視センサ140によって検出されるユーザ190の視線方向は、ユーザ190が物体を視認する際の視点座標系における方向である。HMD装置110のuvw視野座標系は、ユーザ190がディスプレイ112を視認する際の視点座標系に等しい。また、仮想カメラ1のuvw視野座標系は、HMD装置110のuvw視野座標系に連動している。したがって、ある局面に従うHMDシステム100は、注視センサ140によって検出されたユーザ190の視線方向を、仮想カメラ1のuvw視野座標系におけるユーザの視線方向とみなすことができる。

## 【0049】

## [ユーザの視線]

図5を参照して、ユーザの視線方向の決定について説明する。図5は、ある実施の形態に従うHMD装置110を装着するユーザ190の頭部を上から表した図である。

30

## 【0050】

ある局面において、注視センサ140は、ユーザ190の右目および左目の各視線を検出する。ある局面において、ユーザ190が近くを見ている場合、注視センサ140は、視線R1およびL1を検出する。別の局面において、ユーザ190が遠くを見ている場合、注視センサ140は、視線R2およびL2を検出する。この場合、ロール方向wに対して視線R2およびL2がなす角度は、ロール方向wに対して視線R1およびL1がなす角度よりも小さい。注視センサ140は、検出結果をコンピュータ200に送信する。

## 【0051】

コンピュータ200が、視線の検出結果として、視線R1およびL1の検出値を注視センサ140から受信した場合には、その検出値に基づいて、視線R1およびL1の交点である注視点N1を特定する。一方、コンピュータ200は、視線R2およびL2の検出値を注視センサ140から受信した場合には、視線R2およびL2の交点を注視点として特定する。コンピュータ200は、特定した注視点N1の位置に基づき、ユーザ190の視線方向N0を特定する。コンピュータ200は、例えば、ユーザ190の右目Rと左目Lとを結ぶ直線の中点と、注視点N1とを通る直線の延びる方向を、視線方向N0として検出する。視線方向N0は、ユーザ190が両目により実際に視線を向いている方向である。また、視線方向N0は、視界領域23に対してユーザ190が実際に視線を向いている方向に相当する。

40

## 【0052】

50

別の局面において、HMDシステム100は、HMDシステム100を構成するいずれかのパーツに、マイクおよびスピーカを備えてもよい。ユーザは、マイクに発話することにより、仮想空間2に対して、音声による指示を与えることができる。

#### 【0053】

また、別の局面において、HMDシステム100は、テレビジョン放送受信チューナを備えてもよい。このような構成によれば、HMDシステム100は、仮想空間2においてテレビ番組を表示することができる。

#### 【0054】

さらに別の局面において、HMDシステム100は、インターネットに接続するための通信回路、あるいは、電話回線に接続するための通話機能を備えていてもよい。

10

#### 【0055】

##### [視界領域]

図6および図7を参照して、視界領域23について説明する。図6は、仮想空間2において視界領域23をX方向から見たYZ断面を表す図である。図7は、仮想空間2において視界領域23をY方向から見たXZ断面を表す図である。

20

#### 【0056】

図6に示されるように、YZ断面における視界領域23は、領域24を含む。領域24は、仮想カメラ1の基準視線5と仮想空間2のYZ断面とによって定義される。プロセッサ10は、仮想空間2における基準視線5を中心として極角 $\theta$ を含む範囲を、領域24として規定する。

20

#### 【0057】

図7に示されるように、XZ断面における視界領域23は、領域25を含む。領域25は、基準視線5と仮想空間2のXZ断面とによって定義される。プロセッサ10は、仮想空間2における基準視線5を中心とした方位角 $\phi$ を含む範囲を、領域25として規定する。

30

#### 【0058】

ある局面において、HMDシステム100は、コンピュータ200からの信号に基づいて、視界画像をディスプレイ112に表示させることにより、ユーザ190に仮想空間を提供する。視界画像は、仮想空間画像22のうち視界領域23に重畳する部分に相当する。視界領域23内において仮想カメラ1と仮想空間画像22との間に後述する仮想オブジェクトが配置されている場合、視界画像には当該仮想オブジェクトが含まれる。すなわち、視界画像において、仮想空間画像22よりも手前側にある仮想オブジェクトが仮想空間画像22に重畳して表示される。ユーザ190が、頭に装着したHMD装置110を動かすと、その動きに連動して仮想カメラ1も動く。その結果、仮想空間2における視界領域23の位置が変化する。これにより、ディスプレイ112に表示される視界画像は、仮想空間画像22のうち、仮想空間2においてユーザが向いた方向の視界領域23に重畳する画像に更新される。ユーザは、仮想空間2における所望の方向を視認することができる。

30

#### 【0059】

ユーザ190は、HMD装置110を装着している間、現実世界を視認することなく、仮想空間2に展開される仮想空間画像22のみを視認できる。そのため、HMDシステム100は、仮想空間2への高い没入感覚をユーザに与えることができる。

40

#### 【0060】

ある局面において、プロセッサ10は、HMD装置110を装着したユーザ190の現実空間における移動に連動して、仮想空間2において仮想カメラ1を移動し得る。この場合、プロセッサ10は、仮想空間2における仮想カメラ1の位置および傾きに基づいて、HMD装置110のディスプレイ112に投影される画像領域（すなわち、仮想空間2における視界領域23）を特定する。すなわち、仮想カメラ1によって、仮想空間2におけるユーザ190の視野（視界）が定義される。

#### 【0061】

ある実施の形態に従うと、仮想カメラ1は、二つの仮想カメラ、すなわち、右目用の画

50

像を提供するための仮想カメラと、左目用の画像を提供するための仮想カメラとを含むことが望ましい。また、ユーザ190が3次元の仮想空間2を認識できるように、適切な視差が、二つの仮想カメラに設定されていることが好ましい。本実施の形態においては、仮想カメラ1が二つの仮想カメラを含み、二つの仮想カメラのロール方向が合成されることによって生成されるロール方向(w)がHMD装置110のロール方向(w)に適合されるように構成されているものとして、本開示に係る技術思想を例示する。

#### 【0062】

##### [コントローラ]

図8を参照して、コントローラ160の一例について説明する。図8は、ある実施の形態に従うコントローラ160の概略構成を表す図である。

10

#### 【0063】

図8の状態(A)に示されるように、ある局面において、コントローラ160は、右コントローラ160Rと左コントローラ160L(図10参照)とを含み得る。右コントローラ160Rは、ユーザ190の右手で操作される。左コントローラ160Lは、ユーザ190の左手で操作される。ある局面において、右コントローラ160Rと左コントローラ160Lとは、別個の装置として対称に構成される。したがって、ユーザ190は、右コントローラ160Rを持った右手と、左コントローラ160Lを持った左手とをそれぞれ自由に動かすことができる。別の局面において、コントローラ160は両手の操作を受け付ける一体型のコントローラであってもよい。以下、右コントローラ160Rについて説明する。

20

#### 【0064】

右コントローラ160Rは、グリップ30と、フレーム31と、天面32とを備える。グリップ30は、ユーザ190の右手によって把持されるように構成されている。例えば、グリップ30は、ユーザ190の右手の掌と3本の指(中指、薬指、小指)とによって保持され得る。

#### 【0065】

グリップ30は、ボタン33, 34と、モーションセンサ130とを含む。ボタン33は、グリップ30の側面に配置され、右手の中指による操作を受け付ける。ボタン34は、グリップ30の前面に配置され、右手の人差し指による操作を受け付ける。ある局面において、ボタン33, 34は、トリガー式のボタンとして構成される。モーションセンサ130は、グリップ30の筐体に内蔵されている。なお、ユーザ190の動作がカメラその他の装置によってユーザ190の周りから検出可能である場合には、グリップ30は、モーションセンサ130を備えなくてもよい。

30

#### 【0066】

フレーム31は、その円周方向に沿って配置された複数の赤外線LED35を含む。赤外線LED35は、コントローラ160を使用するプログラムの実行中に、当該プログラムの進行に合わせて赤外線を発光する。赤外線LED35から発せられた赤外線は、右コントローラ160Rと左コントローラ160Lとの各位置および姿勢(傾き、向き)等を検出するために使用され得る。図8に示される例では、二列に配置された赤外線LED35が示されているが、配列の数は図8に示されるものに限られない。一列あるいは3列以上の配列が使用されてもよい。

40

#### 【0067】

天面32は、ボタン36, 37と、アナログスティック38とを備える。ボタン36, 37は、プッシュ式ボタンとして構成される。ボタン36, 37は、ユーザ190の右手の親指による操作を受け付ける。アナログスティック38は、ある局面において、初期位置(ニュートラルの位置)から360度任意の方向への操作を受け付ける。当該操作は、例えば、仮想空間2に配置されるオブジェクトを移動させるための操作を含む。

#### 【0068】

ある局面において、右コントローラ160Rおよび左コントローラ160Lは、赤外線LED35その他の部材を駆動するための電池を含む。電池は、充電式、ボタン型、乾電

50

池型等を含むが、これらに限定されない。別の局面において、右コントローラ 160R および左コントローラ 160L は、例えば、コンピュータ 200 の USB インターフェースに接続され得る。この場合、右コントローラ 160R および左コントローラ 160L は、電池を必要としない。

【0069】

図 8 の状態 (A) および状態 (B) に示されるように、例えば、ユーザ 190 の右手 810 に対して、ヨー、ロール、ピッチの各方向が規定される。ユーザ 190 が親指と人差し指とを伸ばした場合に、親指の伸びる方向がヨー方向、人差し指の伸びる方向がロール方向、ヨー方向の軸およびロール方向の軸によって規定される平面に垂直な方向がピッチ方向として規定される。

10

【0070】

[HMD 装置の制御装置]

図 9 を参照して、HMD 装置 110 の制御装置について説明する。ある実施の形態において、制御装置は周知の構成を有するコンピュータ 200 によって実現される。図 9 は、ある実施の形態に従うコンピュータ 200 をモジュール構成として表すブロック図である。

【0071】

図 9 に示されるように、コンピュータ 200 は、表示制御モジュール 220 と、仮想空間制御モジュール 230 と、メモリモジュール 240 と、通信制御モジュール 250 を備える。表示制御モジュール 220 は、サブモジュールとして、仮想カメラ制御モジュール 221 と、視界領域決定モジュール 222 と、視界画像生成モジュール 223 と、基準視線特定モジュール 224 を含む。仮想空間制御モジュール 230 は、サブモジュールとして、仮想空間定義モジュール 231 と、仮想オブジェクト制御モジュール 232 と、操作オブジェクト制御モジュール 233 と、コリジョン制御モジュール 234 と、アクション評価モジュール 235 を含む。

20

【0072】

ある実施の形態において、表示制御モジュール 220 と仮想空間制御モジュール 230 とは、プロセッサ 10 によって実現される。別の実施の形態において、複数のプロセッサ 10 が表示制御モジュール 220 と仮想空間制御モジュール 230 として作動してもよい。メモリモジュール 240 は、メモリ 11 またはストレージ 12 によって実現される。通信制御モジュール 250 は、通信インターフェース 14 によって実現される。

30

【0073】

ある局面において、表示制御モジュール 220 は、HMD 装置 110 のディスプレイ 112 における画像表示を制御する。仮想カメラ制御モジュール 221 は、仮想空間 2 に仮想カメラ 1 を配置し、仮想カメラ 1 の拳動、向き等を制御する。視界領域決定モジュール 222 は、HMD 装置 110 を装着したユーザの頭の向きに応じて、視界領域 23 を規定する。視界画像生成モジュール 223 は、決定された視界領域 23 に基づいて、ディスプレイ 112 に表示される視界画像を生成する。基準視線特定モジュール 224 は、注視センサ 140 からの信号に基づいて、ユーザ 190 の視線を特定する。

40

【0074】

仮想空間制御モジュール 230 は、ユーザ 190 に提供される仮想空間 2 を制御する。仮想空間定義モジュール 231 は、仮想空間 2 を表す仮想空間データを生成することにより、HMD システム 100 における仮想空間 2 を規定する。

【0075】

仮想オブジェクト制御モジュール 232 は、後述するコンテンツ情報 241 およびオブジェクト情報 242 に基づいて、仮想空間 2 に配置される仮想オブジェクトを生成する。また、仮想オブジェクト制御モジュール 232 は、仮想空間 2 における仮想オブジェクトの動作（移動および状態変化等）も制御する。

【0076】

仮想オブジェクトは、仮想空間 2 に配置されるオブジェクト全般である。仮想オブジェ

50

クトは、例えば、ゲームのストーリーの進行に従って配置される森、山その他を含む風景、動物等を含み得る。また、仮想オブジェクトは、仮想空間におけるユーザの分身であるアバターおよびユーザにより操作されるゲームのキャラクタ（プレイヤキャラクタ）等のキャラクタオブジェクトを含み得る。以下の説明において、誤解が生じない場合には、仮想オブジェクトのことを単に「オブジェクト」と表記する。

#### 【0077】

操作オブジェクト制御モジュール233は、ユーザ190の手の動きに応じて動くオブジェクトである操作オブジェクトの仮想空間2内における動作を制御する。ある局面において、操作オブジェクトは、例えば、HMD装置110を装着したユーザ190の手に相当する手オブジェクト、ユーザ190の指に相当する指オブジェクト等を含み得る。また、手オブジェクトにより操作されるオブジェクトも、ユーザ190の手の動きに応じて動く操作オブジェクトとして機能し得る。本実施形態では、手オブジェクトに装着され、手オブジェクトと連動して動く武器オブジェクト（例えば剣を模したオブジェクト）が、操作オブジェクトとして機能する。

10

#### 【0078】

コリジョン制御モジュール234は、仮想空間2に配置されるオブジェクトのそれぞれが、他のオブジェクトと衝突した場合に、当該衝突を検出する。コリジョン制御モジュール234は、例えば、あるオブジェクトと、別のオブジェクトとが触れたタイミングを検出することができ、当該検出がされたときに、予め定められた処理を行う。コリジョン制御モジュール234は、オブジェクトとオブジェクトとが触れている状態から離れたタイミングを検出することができ、当該検出がされたときに、予め定められた処理を行う。コリジョン制御モジュール234は、オブジェクトとオブジェクトとが触れている状態であることを検出することができる。具体的には、コリジョン制御モジュール234は、操作オブジェクトと、他のオブジェクトとが触れた時に、これら操作オブジェクトと他のオブジェクトとが触れたことを検出して、予め定められた処理を行う。

20

#### 【0079】

コリジョン制御モジュール234は、ユーザ190の手の動きの速さに基づいて、コリジョン効果を決定する。コリジョン効果は、例えば操作オブジェクトと他のオブジェクトが衝突する範囲を定義するコリジョンエリア（詳しくは後述）のサイズである。また、コリジョン効果は、例えばコリジョン制御モジュール234が、ユーザ190の手の動きの速さに基づいて、操作オブジェクトと他のオブジェクトとが衝突したと判定された場合に生じさせる効果（所定のパラメータの変動量等）を含む。

30

#### 【0080】

アクション評価モジュール235は、HMD装置110の動き（すなわち、ユーザ190の頭部の動き）またはユーザ190の手の動きに基づいて、仮想空間2におけるユーザ190のアクションを評価する。具体的には、アクション評価モジュール235は、現実空間におけるユーザ190の身体を動きが反映された仮想空間2内におけるプレイヤキャラクタPCおよび操作オブジェクトの動作の格好の良さを評価する。アクション評価モジュール235は、評価結果をディスプレイ112に出力される視界画像に表示したり、図示しないスピーカ等に音声として出力したりすることにより、ユーザ190に通知する。また、アクション評価モジュール235は、評価結果に基づいて、ユーザ190のゲームランク（ユーザ190のアクションの格好の良さを表すレベル）を決定する。

40

#### 【0081】

メモリモジュール240は、コンピュータ200が仮想空間2をユーザ190に提供するためには使用されるデータを保持している。ある局面において、メモリモジュール240は、コンテンツ情報241と、オブジェクト情報242と、ユーザ情報243とを保持している。

#### 【0082】

コンテンツ情報241には、例えば、仮想空間2において再生されるコンテンツ、当該コンテンツで使用されるオブジェクトを配置するための情報等が含まれている。当該コン

50

テソツは、例えば、ゲーム、現実社会と同様の風景を表したコンテンツ等を含み得る。具体的には、コンテンツ情報241は、仮想空間2の背景を規定する仮想空間画像データ(仮想空間画像22)と、仮想空間2に配置されるオブジェクトの定義情報とを含み得る。オブジェクトの定義情報は、オブジェクトを描画するための描画情報(例えば、オブジェクトの形状および色等のデザインを表す情報)、およびオブジェクトの初期配置を示す情報等を含み得る。また、予め設定された動作パターンに基づいて自律的に動作するオブジェクトの定義情報は、当該動作パターンを示す情報(プログラム等)を含み得る。予め定められた動作パターンに基づく動作の例としては、草を模したオブジェクトが一定のパターンで揺れる動作のような単純な繰り返し動作が挙げられる。

## 【0083】

10

オブジェクト情報242には、仮想空間2に配置される各オブジェクトの状態(ゲームの進行およびユーザ190の操作等に応じて変化し得る状態)を示す情報が含まれている。具体的には、オブジェクト情報242は、各オブジェクトの位置を示す位置情報を含み得る。また、オブジェクト情報242は、変形可能なオブジェクトの動作を示す動き情報(すなわち、オブジェクトの形状を特定するための情報)をさらに含み得る。変形可能なオブジェクトの例としては、上述したアバターのように、頭部、胴体、および手等のパートを有し、ユーザ190の動きに応じて各パートを独立して動かすことが可能なオブジェクト等が挙げられる。

## 【0084】

20

ユーザ情報243には、例えば、HMDシステム100の制御装置としてコンピュータ200を機能させるためのプログラム、コンテンツ情報241に保持される各コンテンツを使用するアプリケーションプログラム等が含まれている。

## 【0085】

メモリモジュール240に格納されているデータおよびプログラムは、HMD装置110のユーザによって入力される。あるいは、プロセッサ10が、当該コンテンツを提供する事業者が運営するコンピュータ(例えば、サーバ150)からプログラム(ゲームプログラム等)あるいはデータをダウンロードして、ダウンロードされたプログラムあるいはデータをメモリモジュール240に格納する。

## 【0086】

30

通信制御モジュール250は、ネットワーク19を介して、サーバ150その他の情報通信装置と通信し得る。

## 【0087】

ある局面において、表示制御モジュール220および仮想空間制御モジュール230は、例えば、ユニティテクノロジーズ社によって提供されるUnity(登録商標)を用いて実現され得る。別の局面において、表示制御モジュール220および仮想空間制御モジュール230は、各処理を実現する回路素子の組み合わせとしても実現され得る。

## 【0088】

40

コンピュータ200における処理は、ハードウェアと、プロセッサ10により実行されるソフトウェアとによって実現される。このようなソフトウェアは、ハードディスクその他のメモリモジュール240に予め格納されている場合がある。また、ソフトウェアは、CD-ROMその他のコンピュータ読み取り可能な不揮発性のデータ記録媒体に格納されて、プログラム製品として流通している場合もある。あるいは、当該ソフトウェアは、インターネットその他のネットワークに接続されている情報提供事業者によってダウンロード可能なプログラム製品として提供される場合もある。このようなソフトウェアは、光ディスク駆動装置その他のデータ読み取り装置によってデータ記録媒体から読み取られて、あるいは、通信制御モジュール250を介してサーバ150その他のコンピュータからダウンロードされた後、メモリモジュール240に一旦格納される。そのソフトウェアは、プロセッサ10によってメモリモジュール240から読み出され、実行可能なプログラムの形式でRAMに格納される。プロセッサ10は、そのプログラムを実行する。

## 【0089】

50

図9に示されるコンピュータ200を構成するハードウェアは、一般的なものである。したがって、本実施の形態に係る最も本質的な部分は、コンピュータ200に格納されたプログラムであるともいえる。なお、コンピュータ200のハードウェアの動作は周知であるので、詳細な説明は繰り返さない。

【0090】

なお、データ記録媒体としては、CD-ROM、FD(Flexible Disk)、ハードディスクに限らず、磁気テープ、カセットテープ、光ディスク(MO(Magnetic Optical Disc) / MD(Mini Disc) / DVD(Digital Versatile Disc))、IC(Integrated Circuit)カード(メモリカードを含む)、光カード、マスクROM、EPROM(Erasable Programmable Read-Only Memory)、EEPROM(Electrically Erasable Programmable Read-Only Memory)、フラッシュROMなどの半導体メモリ等の固定的にプログラムを担持する不揮発性のデータ記録媒体でもよい。

10

【0091】

ここでいうプログラムとは、プロセッサ10により直接実行可能なプログラムだけでなく、ソースプログラム形式のプログラム、圧縮処理されたプログラム、暗号化されたプログラム等を含み得る。

【0092】

図10を参照して、コリジョン制御モジュール234が操作オブジェクトと他のオブジェクトとの衝突を判定する処理の詳細について説明する。図10の状態(A)は、HMD装置110とコントローラ160とを装着したユーザ190を示す図である。図10の状態(B)は、仮想カメラ1と手オブジェクト400と対象オブジェクト500とを含む仮想空間2を示す図である。

20

【0093】

図10に示されるように、仮想空間2は、仮想カメラ1と、プレイヤキャラクタPCと、左手オブジェクト400Lと、右手オブジェクト400Rと、対象オブジェクト500とを含む。なお、本実施形態では、プレイヤキャラクタPCの視野は、仮想カメラ1の視野と一致している。これにより、1人称視点における視界画像がユーザに提供される。上述したように、仮想空間制御モジュール230の仮想空間定義モジュール231は、このようなオブジェクトを含む仮想空間2を規定する仮想空間データを生成する。また、上述したように、仮想カメラ1は、ユーザ190に装着されたHMD装置110の動きに連動する。すなわち、仮想カメラ1の視野は、HMD装置110の動きに応じて更新される。右手オブジェクト400Rは、ユーザ190の右手に装着される右コントローラ160Rの動きに応じて移動する操作オブジェクトである。左手オブジェクト400Lは、ユーザ190の左手に装着される左コントローラ160Lの動きに応じて移動する操作オブジェクトである。以降では、説明の便宜上、左手オブジェクト400Lおよび右手オブジェクト400Rのそれぞれを単に手オブジェクト400と総称する場合がある。

30

【0094】

左手オブジェクト400Lおよび右手オブジェクト400Rは、それぞれコリジョンエリアCAを有する。対象オブジェクト500は、コリジョンエリアCBを有する。プレイヤキャラクタPCは、コリジョンエリアCCを有する。コリジョンエリアCA, CB, CCは、各オブジェクト間におけるコリジョン判定(当たり判定)のために利用される。例えば、手オブジェクト400のコリジョンエリアCAと対象オブジェクト500のコリジョンエリアCBとが接触している場合(互いに重なる領域を有する場合を含む)に、手オブジェクト400と対象オブジェクト500とが衝突していると判定される。図10に示されるように、コリジョンエリアCA, CB, CCは、各オブジェクトに設定された座標位置を中心とし、所定の半径を有する球により規定されてもよい。

40

【0095】

[制御構造]

図11を参照して、本実施の形態に係るコンピュータ200の制御構造について説明する。図11は、HMDシステム100が実行する処理を表すフローチャートである。

50

## 【0096】

ステップS1において、コンピュータ200のプロセッサ10は、仮想空間定義モジュール231として、仮想空間画像データを特定し、仮想空間を定義する。すなわち、プロセッサ10は、仮想空間2を規定する仮想空間データを生成する。

## 【0097】

ステップS2において、プロセッサ10は、仮想カメラ制御モジュール221として、仮想カメラ1を初期化する。例えば、プロセッサ10は、メモリのワーク領域において、仮想カメラ1を仮想空間2において予め規定された中心点に配置し、仮想カメラ1の視線をユーザ190が向いている方向に向ける。

## 【0098】

ステップS3において、プロセッサ10は、視界画像生成モジュール223として、初期の視界画像を表示するための視界画像データを生成する。生成された視界画像データは、視界画像生成モジュール223を介して通信制御モジュール250によってHMD装置110に送られる。

## 【0099】

ステップS4において、HMD装置110のディスプレイ112は、コンピュータ200から受信した信号に基づいて、視界画像を表示する。HMD装置110を装着したユーザ190は、視界画像を視認すると仮想空間2を認識し得る。

## 【0100】

ステップS5において、HMDセンサ120は、HMD装置110から発信される複数の赤外線光に基づいて、HMD装置110の位置と傾きを検知する。検知結果は、動き検知データとして、コンピュータ200に送られる。

## 【0101】

ステップS6において、プロセッサ10は、視界領域決定モジュール222として、HMD装置110の位置と傾きとにに基づいて、HMD装置110を装着したユーザ190の視界方向を特定する。プロセッサ10は、アプリケーションプログラムを実行し、アプリケーションプログラムに含まれる命令に基づいて、仮想空間2にオブジェクトを配置する。

## 【0102】

ステップS7において、コントローラ160は、現実空間におけるユーザ190の操作を検出する。例えば、ある局面において、コントローラ160は、ユーザ190によってボタンが押下されたことを検出する。別の局面において、コントローラ160は、ユーザ190の手の動き（たとえば、手を振る動き等）を検出する。具体的には、コントローラ160は、ユーザ190の手が動いた方向および速さ等を検出する。検出内容を示す信号は、コンピュータ200に送られる。

## 【0103】

ステップS8において、プロセッサ10は、仮想オブジェクト制御モジュール232として、ユーザ190の仮想空間2におけるアクションを決定する。例えば、プロセッサ10は、ステップS5で検知されたHMD装置110の動きに基づいて、プレイヤキャラクタP C（図10に示されるように、本実施形態ではユーザ190の頭部に相当するオブジェクト）を動かす。また、プロセッサ10は、ステップS7で検知されたユーザ190の手の動きに基づいて、手オブジェクト400を動かす。プロセッサ10は、手オブジェクト400に他のオブジェクトが装着されている場合には、当該他のオブジェクトを手オブジェクト400と連動させる。

## 【0104】

ステップS9において、プロセッサ10は、コリジョン制御モジュール234として、オブジェクト同士のコリジョンの制御を実行する。ステップS9の処理の詳細については後述する。

## 【0105】

ステップS10において、プロセッサ10は、視界領域決定モジュール222および視

10

20

30

40

50

界画像生成モジュール 223 として、処理の結果に基づく視界画像を表示するための視界画像データを生成し、生成した視界画像データを HMD 装置 110 に出力する。

【0106】

ステップ S11において、HMD 装置 110 のディスプレイ 112 は、受信した視界画像データに基づいて視界画像を更新し、更新後の視界画像を表示する。

【0107】

ステップ S5～S11 の処理は、定期的に繰り返し実行される。

【0108】

[コリジョン制御(攻撃動作)]

図 12 および図 13 を参照して、攻撃動作に着目したコリジョン制御(ステップ S9)の処理手順を説明する。具体的には、武器オブジェクト W と敵オブジェクト E との間のコリジョン制御の処理手順について説明する。図 13 に示されるように、武器オブジェクト W は、手オブジェクト 400 に装着され、手オブジェクト 400 と連動して動く剣を模した操作オブジェクトである。敵オブジェクト E は、武器オブジェクト W の攻撃対象となる対象オブジェクトである。武器オブジェクト W にはコリジョンエリア CD が設定され、敵オブジェクト E にはコリジョンエリア CE が設定されている。武器オブジェクト W と敵オブジェクト E との間には、武器オブジェクト W が敵オブジェクト E に衝突した場合(すなわち、コリジョンエリア CD とコリジョンエリア CE との衝突が検知された場合)に敵オブジェクト E にダメージを与えることができるという関係が予め定められている。

10

【0109】

ステップ S901において、プロセッサ 10 は、武器オブジェクト W の移動速度(すなわち、検出されたユーザ 190 の手の動きの速さに対応する移動速度)に基づいて、武器オブジェクト W に関連付けられるコリジョンエリア CD のサイズを決定する。

20

【0110】

プロセッサ 10 は、武器オブジェクト W の移動速度が大きい程、コリジョンエリア CD のサイズを大きくしてもよい。ユーザ 190 が手を速く動かす場合には、武器オブジェクト W を敵オブジェクト E に当てることが一層困難になると予想される。その場合に、武器オブジェクト W の移動速度に応じて敵オブジェクト E への攻撃の当り易さを調整することで、ユーザ 190 に対して、より直感的な仮想体験を提供し得る。

30

【0111】

プロセッサ 10 は、武器オブジェクト W の移動速度に応じてコリジョンエリア CD のサイズを連続的に変化させてもよい。例えば、プロセッサ 10 は、武器オブジェクト W の移動速度  $x$  に対応するコリジョンエリア CD のサイズ  $F(x)$  を求めるために予め定められた関数  $F$  を用いることにより、コリジョンエリア CD のサイズを決定してもよい。関数  $F$  は、任意の移動速度  $x_1, x_2$  について、 $x_1 > x_2$  の場合に  $F(x_1) > F(x_2)$  が成立する単調増加関数であってもよい。この場合、プロセッサ 10 は、関数  $F$  を用いた演算により、武器オブジェクト W の移動速度が大きい程、コリジョンエリア CD のサイズを大きくすることができる。

30

【0112】

あるいは、プロセッサ 10 は、武器オブジェクト W の移動速度に応じてコリジョンエリア CD のサイズを離散的に変化させてもよい。例えば、プロセッサ 10 は、武器オブジェクト W の移動速度と予め定められた閾値とを比較し、閾値以上の場合に閾値未満の場合よりも大きくなるように、コリジョンエリア CD のサイズを決定してもよい。このような閾値を複数設けることにより、武器オブジェクト W の移動速度に応じてコリジョンエリア CD のサイズを階段状に変化させてもよい。

40

【0113】

図 13 の状態(A)は、武器オブジェクト W の移動速度が  $v_1$  であるときのコリジョンエリア CD を表しており、図 13 の状態(B)は、武器オブジェクト W の移動速度が  $v_2$  ( $v_2 < v_1$ ) であるときのコリジョンエリア CD を表している。本実施形態では、状態(A)および状態(B)に示されるように、武器オブジェクト W の移動速度が大きい程、コリ

50

ジョンエリアCDのサイズが大きくなる。これにより、ユーザ190が手を動かす速さに応じて、敵オブジェクトEに対する武器オブジェクトWの当たり易さを適切に調整できる。

#### 【0114】

ステップS902において、プロセッサ10は、武器オブジェクトWの移動速度に基づいて、コリジョンエリアCDに基づいて武器オブジェクトWと敵オブジェクトEとが衝突したと判定された場合に生じさせる所定のパラメータの変動量を決定する。所定のパラメータは、例えば、敵オブジェクトEに関連付けられた体力値（所謂ヒットポイント）である。この場合、所定のパラメータの変動量は、敵オブジェクトEに与えるダメージ量（すなわち、敵オブジェクトEの体力値の減少量）である。

10

#### 【0115】

プロセッサ10は、武器オブジェクトWの移動速度が大きい程、敵オブジェクトEに与えるダメージ量を大きくしてもよい。この場合、ユーザ190が手を速く動かして攻撃動作を行った場合に、敵オブジェクトEへの攻撃効果が高くなる。これにより、ダイナミックに身体を動かしてゲームを楽しむことをユーザ190に促すことができる。

#### 【0116】

ステップS903において、プロセッサ10は、コリジョンエリアCDおよびコリジョンエリアCEに基づいて、武器オブジェクトWと敵オブジェクトEとが衝突したか否かを判定する。プロセッサ10は、衝突を検知しなかった場合（ステップS903：NO）、処理を終了する。一方、プロセッサ10は、衝突を検知した場合（ステップS903：YES）、ステップS904の処理に進む。

20

#### 【0117】

ステップS904およびS905の処理は、敵オブジェクトEに対する武器オブジェクトWの移動方向に基づいて、敵オブジェクトEに与えるダメージ量を決定（補正）する処理である。

#### 【0118】

ステップS904において、プロセッサ10は、武器オブジェクトWの移動方向（すなわち、検出されたユーザ190の手の動きに基づいて特定される仮想空間2における移動方向）に基づいて、武器オブジェクトWが敵オブジェクトEに向かって動いた結果コリジョンが発生したか否かを判定する。

30

#### 【0119】

武器オブジェクトWが敵オブジェクトEに向かって動いた結果コリジョンが発生したと判定された場合（ステップS904：YES）には、ステップS905において、プロセッサ10は、敵オブジェクトEに与えるダメージ量を上方修正する。一方、武器オブジェクトWが敵オブジェクトEに向かって動いた結果コリジョンが発生したと判定されなかった場合（ステップS904：NO）には、プロセッサ10は、敵オブジェクトEに与えるダメージ量の上方修正を実行しない。図13の状態（C）は、ユーザ190が武器オブジェクトWを振り上げた際に、武器オブジェクトWに向かってくる敵オブジェクトEが偶々武器オブジェクトWに衝突する直前の状態を表している。このように、武器オブジェクトWが敵オブジェクトEから遠ざかる方向に移動している最中に武器オブジェクトWと敵オブジェクトEとが衝突した場合には、上述したステップS904における判定結果は「NO」となる。

40

#### 【0120】

このような処理により、武器オブジェクトWが偶々敵オブジェクトEに衝突した場合よりも、ユーザ190が攻撃動作を意図して動いた結果武器オブジェクトWが敵オブジェクトEに衝突した場合の攻撃効果を高くすることができる。その結果、ユーザ190に積極的な攻撃動作を促すことができ、ゲーム性をより一層向上させることができる。なお、プロセッサ10は、武器オブジェクトWが敵オブジェクトEに向かって動いた結果コリジョンが発生したと判定されなかった場合（ステップS904：NO）に、敵オブジェクトEに与えるダメージ量を下方修正してもよい。この場合にも、上述同様の効果が得られる。

50

## 【0121】

ステップS906において、プロセッサ10は、武器オブジェクトWと敵オブジェクトEとの関係に応じた処理を実行する。上述したように、武器オブジェクトWと敵オブジェクトEとの間には、武器オブジェクトWが敵オブジェクトEに衝突した場合に敵オブジェクトEにダメージを与えることができるという関係が予め定められている。したがって、プロセッサ10は、ステップS902およびS905において決定されたダメージ量に基づいて、敵オブジェクトEの体力値を減少させる。

## 【0122】

なお、上記例では、所定のパラメータが敵オブジェクトEに関連付けられた体力値である場合について説明したが、所定のパラメータは、これに限られない。例えば、所定のパラメータは、敵オブジェクトEの怒り値（例えば一定以上蓄積された場合に敵オブジェクトEの攻撃パターンを変化させるためのパラメータ）等であってもよい。また、所定のパラメータは、武器オブジェクトWで敵オブジェクトEを攻撃することにより得られるスコアおよびゲーム内通貨の量等であってもよい。

10

## 【0123】

## [コリジョン制御（防御動作）]

図14および図15を参照して、防御動作に着目したコリジョン制御（ステップS9）の処理手順を説明する。具体的には、武器オブジェクトWと攻撃オブジェクトAとの間のコリジョン制御の処理手順について説明する。図15に示されるように、攻撃オブジェクトAは、プレイヤキャラクタPCを攻撃する対象オブジェクトである。この例では、攻撃オブジェクトAは、例えば敵オブジェクトEから放たれた飛び道具（ここでは矢）を模したオブジェクトである。ただし、攻撃オブジェクトAは、この例に限らず、例えば敵オブジェクトEの身体の一部（例えば敵オブジェクトEが振り回す腕等）であってもよいし、敵オブジェクトEが手に持った武器等であってもよい。攻撃オブジェクトAには、コリジョンエリアCFが設定されている。プレイヤキャラクタPCと攻撃オブジェクトAとの間には、プレイヤキャラクタPCが攻撃オブジェクトAに衝突した場合（すなわち、コリジョンエリアCCとコリジョンエリアCFとの衝突が検知された場合）にプレイヤキャラクタPCがダメージを受けるという関係が予め定められている。一方、武器オブジェクトWと攻撃オブジェクトAとの間には、武器オブジェクトWが攻撃オブジェクトAに衝突した場合にプレイヤキャラクタPCがダメージを受け得るという関係が予め定められている。ただし、武器オブジェクトWが攻撃オブジェクトAに衝突した場合のプレイヤキャラクタPCの被ダメージ量は、プレイヤキャラクタPCと攻撃オブジェクトAとが衝突する場合の被ダメージ量よりも小さい値（あるいは0）に設定されている。すなわち、ユーザ190は、武器オブジェクトWを攻撃オブジェクトAに衝突させ、攻撃オブジェクトAを破壊したり弾き飛ばしたりすることにより、プレイヤキャラクタPCの被ダメージ量を軽減する（あるいは0にする）ことができる。

20

30

## 【0124】

ステップS911において、プロセッサ10は、武器オブジェクトWの移動速度が予め定められた閾値以下であるか否かを判定する。移動速度が閾値以下と判定されなかった場合（ステップS911：NO）には、プロセッサ10は、ステップS912の処理に進む。

40

## 【0125】

ステップS912において、プロセッサ10は、上述したステップS901の処理と同様に、武器オブジェクトWの移動速度に基づいて、武器オブジェクトWに関連付けられるコリジョンエリアCDのサイズを決定する。具体的には、プロセッサ10は、武器オブジェクトWの移動速度が大きい程、コリジョンエリアCDのサイズを大きくする。

## 【0126】

図15の状態（A）は、武器オブジェクトWの移動速度がv1であるときのコリジョンエリアCDを表しており、図15の状態（B）は、武器オブジェクトWの移動速度がv2（< v1）であるときのコリジョンエリアCDを表している。本実施形態では、状態（A

50

) および状態 (B) に示されるように、武器オブジェクトWの移動速度が大きい程、コリジョンエリアCDのサイズが大きくなる。これにより、ユーザ190が手を動かす速さに応じて、攻撃オブジェクトAに対する武器オブジェクトWの当たり易さを適切に調整できる。

#### 【0127】

ステップS913において、プロセッサ10は、武器オブジェクトWの移動速度に基づいて、コリジョンエリアCDに基づいて武器オブジェクトWと攻撃オブジェクトAとが衝突したと判定された場合に生じさせる所定のパラメータの変動量を決定する。所定のパラメータは、例えば、ブレイヤキャラクタPCに関連付けられた体力値（所謂ヒットポイント）である。この場合、所定のパラメータの変動量は、ブレイヤキャラクタPCの被ダメージ量（すなわち、ブレイヤキャラクタPCの体力値の減少量）である。10

#### 【0128】

プロセッサ10は、武器オブジェクトWの移動速度が大きい程、ブレイヤキャラクタPCの被ダメージ量を小さくしてもよい。この場合、ユーザ190が手を速く動かして防御動作（例えば攻撃オブジェクトAを武器オブジェクトWで弾き飛ばしたり破壊したりする動作）を行った場合に、攻撃オブジェクトAに対する防御効果が高くなる。これにより、ダイナミックに身体を動かしてゲームを楽しむことをユーザ190に促すことができる。

#### 【0129】

ステップS914において、プロセッサ10は、コリジョンエリアCDおよびコリジョンエリアCFに基づいて、武器オブジェクトWと攻撃オブジェクトAとが衝突したか否かを判定する。プロセッサ10は、衝突を検知した場合（ステップS914：YES）、ステップS915の処理に進む。20

#### 【0130】

ステップS915およびS916の処理は、攻撃オブジェクトAに対する武器オブジェクトWの移動方向に基づいて、ブレイヤキャラクタPCの被ダメージ量を決定（補正）する処理である。

#### 【0131】

ステップS915において、プロセッサ10は、武器オブジェクトWの移動方向に基づいて、武器オブジェクトWが攻撃オブジェクトAに向かって動いた結果コリジョンが発生したか否かを判定する。30

#### 【0132】

武器オブジェクトWが攻撃オブジェクトAに向かって動いた結果コリジョンが発生したと判定された場合（ステップS915：YES）には、ステップS916において、プロセッサ10は、ブレイヤキャラクタPCの被ダメージ量を下方修正する。一方、武器オブジェクトWが攻撃オブジェクトAに向かって動いた結果コリジョンが発生したと判定されなかった場合（ステップS915：NO）には、プロセッサ10は、ブレイヤキャラクタPCの被ダメージ量の下方修正を実行しない。

#### 【0133】

このような処理により、武器オブジェクトWが偶々攻撃オブジェクトAに衝突した場合よりも、ユーザ190が防御動作を意図して動いた結果武器オブジェクトWが攻撃オブジェクトAに衝突した場合の防御効果を高くすることができる。その結果、ユーザ190に積極的な防御動作を促すことができ、ゲーム性をより一層向上させることができる。なお、プロセッサ10は、武器オブジェクトWが攻撃オブジェクトAに向かって動いた結果コリジョンが発生したと判定されなかった場合（ステップS915：NO）に、攻撃オブジェクトAから受ける被ダメージ量を上方修正してもよい。この場合にも、上述同様の効果が得られる。40

#### 【0134】

ステップS917において、プロセッサ10は、武器オブジェクトWと攻撃オブジェクトAとの関係に応じた処理を実行する。上述したように、武器オブジェクトWと攻撃オブジェクトAとの間には、武器オブジェクトWが攻撃オブジェクトAに衝突した場合にブレ

10

20

30

40

50

イヤキャラクタＰＣがダメージを受け得るという関係が予め定められている。したがって、プロセッサ10は、ステップS913およびS916において決定された被ダメージ量が0より大きい場合には、当該被ダメージ量に基づいてプレイヤキャラクタＰＣの体力値を減少させる。

#### 【0135】

なお、プロセッサ10は、武器オブジェクトWの移動速度に応じて、武器オブジェクトWを攻撃オブジェクトAに衝突させた場合に攻撃オブジェクトAに与える効果値を決定してもよい。そして、プロセッサ10は、決定された効果値に応じて攻撃オブジェクトAに對して武器オブジェクトWが与える作用を変化させてもよい。例えば、プロセッサ10は、効果値が予め定められた閾値以上の場合には攻撃オブジェクトAを真っ二つに破壊するようにもよいし、効果値が閾値未満の場合には攻撃オブジェクトAを弾き飛ばすようにしてもよい。

10

#### 【0136】

続いて、ステップS911の判定結果が「Y E S」の場合の処理について説明する。この場合、ステップS918において、プロセッサ10は、武器オブジェクトWのコリジョンエリアCDを無効化する。プロセッサ10は、コリジョンエリアCD自体を消滅させてもよいし、コリジョンエリアCDと他のオブジェクトのコリジョンエリアとが重なった場合にコリジョンを検出しないことを示すフラグ情報等をコリジョンエリアCDに設定してもよい。図15の状態(C)は、武器オブジェクトWのコリジョンエリアCD自体を消滅させる例を表している。このように、武器オブジェクトWの移動速度が閾値未満である場合に、武器オブジェクトWと他のオブジェクトとの当たり判定を無効化することにより、武器オブジェクトWによる防御を不可能にすることができる。その結果、ユーザ190が武器オブジェクトWを構えてひたすら防御するといった消極的なプレイをすることを抑制することができる。

20

#### 【0137】

ステップS918において武器オブジェクトWのコリジョンエリアCDが無効化された場合、または、ステップS914において武器オブジェクトWと攻撃オブジェクトAとの衝突が検知されなかった場合には、武器オブジェクトWによる攻撃オブジェクトAに対する防御は失敗する。この場合、ステップS919において、プロセッサ10は、コリジョンエリアCDおよびコリジョンエリアCFに基づいて、プレイヤキャラクタＰＣと攻撃オブジェクトAとが衝突したか否かを判定する。プロセッサ10は、衝突を検知しなかった場合(ステップS919: N O)、処理を終了する。一方、プロセッサ10は、衝突を検知した場合(ステップS919: Y E S)、ステップS920において、予め定められた被ダメージ量(例えば攻撃オブジェクトAに関連付けられた攻撃力およびプレイヤキャラクタＰＣに関連付けられた防御力等に基づいて決定される被ダメージ量)に基づいて、プレイヤキャラクタＰＣの体力値を減少させる。

30

#### 【0138】

なお、上記例では、所定のパラメータがプレイヤキャラクタＰＣに関連付けられた体力値である場合について説明したが、所定のパラメータは、これに限られない。例えば、所定のパラメータは、プレイヤキャラクタＰＣの怒り値(例えば一定以上蓄積された場合に必殺技を利用可能な状態に変化させるためのパラメータ)等であってもよい。

40

#### 【0139】

##### [アクション評価]

図16～図19を参照して、仮想空間2におけるユーザ190のアクションを評価するための処理手順を説明する。本実施形態では、プロセッサ10は、仮想空間2に展開されるゲームコンテンツが提供するクエスト(あるいはステージ)中のユーザ190の個々のアクションに基づいて評価値を変化させる。そして、プロセッサ10は、最終的な評価値に基づいて、ユーザ190のゲームランクまたはスコア等を決定する。1つのクエストは、例えば当該クエストに設定された目的(例えば、特定の敵キャラクタを倒すこと)を達成することにより終了する。また、クエストに設定された目的達成に失敗した場合(例え

50

ばプレイヤキャラクタＰＣの体力値が0になった場合、および予め定められた制限時間を超えた場合等)にも、クエストは終了する。

#### 【0140】

ステップS101において、プロセッサ10は、ユーザ190により選択されたクエストを開始する。例えば、プロセッサ10は、仮想空間2内にクエストを選択するためのメニュー画面を表示してもよい。そして、プロセッサ10は、ユーザ190が、手オブジェクト400(または手オブジェクト400により操作されるタッチペン等のオブジェクト)によりメニュー画面をタッチ操作してクエストを選択できるようにしてよい。

#### 【0141】

ステップS102において、プロセッサ10は、ユーザ190の仮想空間2におけるアクションを決定する。具体的には、プロセッサ10は、検出されたHMD装置110の動きに基づいてプレイヤキャラクタＰＣを動かすとともに、検出されたユーザ190の手の動きに基づいて手オブジェクト400を動かす。手オブジェクト400に他のオブジェクトが装着されている場合には、プロセッサ10は、当該他のオブジェクトも手オブジェクト400に連動させる。当該処理は、図11のステップS8の処理に相当する。

10

#### 【0142】

ステップS103において、プロセッサ10は、アクション評価モジュール235として、ステップS102で決定されたアクションを判定する。具体的には、プロセッサ10は、当該アクションが予め定められた積極的な動作であるか、あるいは、当該アクションが予め定められた消極的な動作であるかを判定する。積極的な動作は、仮想空間2におけるゲームをプレイする上で格好の良い動作として予め定められた動作であり、評価値を増大させる動作である。消極的な動作は、格好の良くない動作として予め定められた動作であり、評価値を減少させる動作である。

20

#### 【0143】

##### (第1の判定例)

プロセッサ10は、HMD装置110またはユーザ190の手(身体の一部)について予め定められた動きが予め定められた時間以上検出されなかった場合に、ユーザ190のアクションを消極的な動作であると判定してもよい。予め定められた動きとは、例えばHMD装置110またはユーザ190の手が予め定められた距離以上移動した場合に生じる動きである。つまり、小刻みに揺れている状態のように実質的に同じ位置に静止しているに等しい動きは、上記予め定められた動きには該当しない。上記判定によれば、ユーザ190が実質的にゲーム内で意味のあるアクションを行っておらず、敵オブジェクトEに対する攻撃動作等を積極的に行っていない場合に、ユーザ190のアクションを消極的であると判定することができる。

30

#### 【0144】

##### (第2の判定例)

プロセッサ10は、HMD装置110またはユーザ190の手の検出された位置に対応する仮想空間2における位置に基づいて、ユーザ190のアクションを判定してもよい。例えば、プロセッサ10は、上記仮想空間2における位置が敵オブジェクトEからの攻撃(攻撃オブジェクトA)が当たらないエリアとして予め設定された安全地帯に含まれており、その状態が予め定められた時間以上続いた場合に、ユーザ190のアクションを消極的な動作であると判定してもよい。

40

#### 【0145】

##### (第3の判定例)

プロセッサ10は、HMD装置110またはユーザ190の手の検出された位置に対応する仮想空間2における高さに基づいて、ユーザ190のアクションを判定してもよい。図17の状態(A)に示されるように、例えばユーザ190がしゃがんだ状態であり、HMD装置110の検出された位置に対応する仮想空間2における高さ(仮想空間2内に設定された仮想地面Gに対するプレイヤキャラクタＰＣの高さh)が所定高さより低い場合、攻撃オブジェクトAの攻撃が当たらない(あるいは当たり難い)状態になる場合がある

50

。そこで、例えば、プロセッサ10は、仮想空間2におけるプレイヤキャラクタPCの高さhが所定の高さより低い状態が予め定められた時間以上続いた場合に、ユーザ190のアクションを消極的な動作であると判定してもよい。

#### 【0146】

##### (第4の判定例)

上述したコリジョン制御において、武器オブジェクトWと敵オブジェクトEまたは攻撃オブジェクトAとの衝突が検知された場合には、ユーザ190が積極的に攻撃動作または防御動作を行っているといえる。そこで、プロセッサ10は、武器オブジェクトWと敵オブジェクトEまたは攻撃オブジェクトAとの衝突が検知された場合に、ユーザ190のアクションを積極的な動作であると判定してもよい。

10

#### 【0147】

##### (第5の判定例)

一方、攻撃オブジェクトAからの攻撃を防御するための操作オブジェクト(例えば盾を模した防具オブジェクトD)が手オブジェクト400に関連付けられており、防具オブジェクトDによる防御動作が予め定められた時間以上続いた場合には、ユーザ190が専ら防御動作のみを行っており、積極的に攻撃動作を行っていないといえる。このような場合には、プロセッサ10は、ユーザ190のアクションを消極的な動作であると判定してもよい。

20

#### 【0148】

防具オブジェクトDによる防御動作が行われているか否かは、視界画像Mにおいて防具オブジェクトDが占める割合に基づいて判定され得る。図18は、ユーザ190が防具オブジェクトDに隠れて攻撃オブジェクトAからの攻撃を防御するための防御動作を行っている状態の視界画像Mの例を表している。視界画像Mには、盾を模した防具オブジェクトDの裏側(手で掴む側)部分が表示されている。このような防御動作が行われているときには、視界画像Mにおける防具オブジェクトDの表示面積が大きくなる傾向がある。そこで、プロセッサ10は、視界画像Mにおける防具オブジェクトDの表示面積が予め定められた閾値以上である場合に、防具オブジェクトDによる防御動作が行われている状態であると判定してもよい。

20

#### 【0149】

##### (第6の判定例)

30

ユーザ190が身体を動かすことによって攻撃オブジェクトAを回避する動作は、格好の良い動作といえる。そこで、プロセッサ10は、ユーザ190のアクションが攻撃オブジェクトAを回避する回避動作に該当する場合には、当該アクションを積極的な動作であると判定してもよい。

#### 【0150】

回避動作であるか否かについては、以下のように判定され得る。すなわち、プロセッサ10は、攻撃オブジェクトAの一連の攻撃動作が実行される間に、HMD装置110の動きが検知され、かつ、プレイヤキャラクタPCのコリジョンエリアCC(第1コリジョンエリア)と攻撃オブジェクトAのコリジョンエリアCF(第2コリジョンエリア)との衝突が検知されなかった場合、ユーザ190のアクションを回避動作であると判定してもよい。攻撃オブジェクトAの一連の攻撃動作の開始時点は、例えば攻撃オブジェクトAがプレイヤキャラクタPCから予め定められた閾値距離内まで近づいたタイミングである。あるいは、敵オブジェクトEの動作を規定するプログラムにおいて予め設定された攻撃動作が実行されるタイミング(例えば攻撃オブジェクトAを放つタイミング)が、攻撃オブジェクトAの一連の攻撃動作の開始時点とみなされてもよい。

40

#### 【0151】

##### (第7の判定例)

上述した回避動作のうち、特に攻撃オブジェクトAからの攻撃をぎりぎりで回避する動作(以下「特定回避動作」)は、特に格好の良い動作といえる。そこで、プロセッサ10は、ユーザ190のアクションが特定回避動作に該当する場合に、当該アクションを積極

50

的な動作（あるいは、積極的な動作のうち評価値の増大量が特に大きい動作）であると判定してもよい。

#### 【0152】

例えば、プロセッサ10は、攻撃オブジェクトAに対して、プレイヤキャラクタPCに攻撃を与えることができる範囲を規定するコリジョンエリアCF1（上述したコリジョンエリアCFと同様のコリジョンエリア）とともに、当該コリジョンエリアCF1を包含するコリジョンエリアCF2（第3コリジョンエリア）を設定する。そして、プロセッサ10は、コリジョンエリアCCとコリジョンエリアCF1との衝突が検知されず、かつ、コリジョンエリアCCとコリジョンエリアCF2との衝突が検知された場合に、ユーザ190のアクションを特定回避動作であると判定してもよい。図17の状態（B）は、このような特定回避動作の一例を表している。このように二重のコリジョンエリアを設定し、外側のコリジョンエリアにのみ衝突したか否かを判定することにより、特定回避動作を少ない計算負荷で容易に判定することができる。また、ユーザ190に敵からの攻撃が当たるか当たらないかぎりぎりの行動を促すことにより、ユーザ190のゲームに対する高揚感をより高めることができる。

10

#### 【0153】

##### （第8の判定例）

ユーザ190のアクションが回避動作に該当するか否かに基づいてアクションを判定する具体的な処理として、プロセッサ10は以下の処理を実行してもよい。すなわち、プロセッサ10は、攻撃オブジェクトAの一連の攻撃動作の開始時以前に当該一連の攻撃動作におけるコリジョンエリアCFの軌道（すなわち、コリジョンエリアCFの将来の動きを表す仮モーション）を算出する。そして、プロセッサ10は、当該一連の攻撃動作の開始時におけるコリジョンエリアCCとコリジョンエリアCFの軌道とが重なっており、かつ、当該一連の攻撃動作が実行された際にコリジョンエリアCCとコリジョンエリアCFとの衝突が実際には検知されなかった場合に、ユーザ190のアクションを積極的な動作であると判定する。

20

#### 【0154】

図19の状態（A）は、攻撃オブジェクトAの一連の攻撃動作におけるコリジョンエリアCFの軌道Tを表している。この例では、軌道Tは、プレイヤキャラクタPCに向かって直進するように円柱状に延びる軌跡を表す。また、状態（A）は、一連の攻撃動作の開始時におけるコリジョンエリアCCと軌道Tとが重なっている状態を表している。図19の状態（B）は、ユーザ190が頭部（すなわちHMD装置110）を移動させることにより、攻撃オブジェクトAの攻撃を回避した状態を表している。このように、回避動作が成功した場合には、一連の攻撃動作が実行された際に、コリジョンエリアCCとコリジョンエリアCFとの衝突は検知されない。したがって、図19に示される例では、プロセッサ10は、ユーザ190のアクションを積極的な動作であると判定する。

30

#### 【0155】

このように、実際の攻撃動作が実行される前に、攻撃オブジェクトAの仮のモーション（軌道T）を生成し、軌道Tに基づく判定を行うことにより、ユーザ190のアクションが回避動作に該当するか否かを精度良く判定することができる。

40

#### 【0156】

ステップS104において、プロセッサ10は、アクション評価モジュール235として、アクションの判定結果に応じてユーザ190に関連付けられる評価値を変化させる。具体的には、プロセッサ10は、上記例示した第4, 6~8の判定例のように、アクションを積極的な動作であると判定した場合には、上記評価値を増大させる。一方、プロセッサ10は、上記例示した第1~3, 5の判定例のように、アクションを消極的な動作であると判定した場合には、上記評価値を減少させる。これにより、仮想空間2に反映されるユーザ190のアクションが格好の良い動作である場合に評価値を増大させ、ユーザ190のアクションが格好の良くない動作である場合に評価値を減少させることができる。

#### 【0157】

50

ステップS105において、プロセッサ10は、アクション評価モジュール235として、ユーザ190の個々のアクションの判定結果（評価値の変化）に基づく所定のゲーム制御を実行する。例えば、プロセッサ10は、アクションの判定結果に対応するメッセージ（例えば「Cool！」、「Uncool」等の文字）が表示された視界画像をディスプレイ112に出力してもよい。また、プロセッサ10は、アクションの判定結果に対応する音声をスピーカ等に出力してもよい。このようにユーザ190のアクションの判定結果をユーザ190に通知することにより、アクションの判定結果が良かった場合にはユーザ190に快感を与えることができ、アクションの判定結果が悪かった場合にはユーザ190に格好の良い動作を促すことができる。

【0158】

10

ステップS106において、プロセッサ10は、クエストの終了判定を行う。プロセッサ10は、クエストが終了するまで、ステップS102～S105の処理を繰り返し実行する。クエストが終了した場合、ステップS107において、プロセッサ10は、アクション評価モジュール235として、ユーザ190のクエスト内におけるアクションの判定結果（最終的な評価値）に基づく所定のゲーム制御を実行する。例えば、プロセッサ10は、最終的な評価値に応じたランクまたはスコアを決定する。また、プロセッサ10は、このようにして決定されたランク等に基づいて、ユーザ190に与える報酬（例えばゲーム内で利用可能な特定のアイテム等）を決定してもよい。

【0159】

20

以上述べたアクション評価の処理によれば、積極的な動作に該当するアクションを高く評価することで、ユーザ190に積極的な動作を取るように促すことができる。すなわち、ゲーム提供者が意図する格好の良いアクションを取ることをユーザ190に促すことができ、実際にそのようなアクションを取ったユーザ190に対して高い評価値を与えることにより、ユーザ190のゲームに対する高揚感を効果的に高め得る。

【0160】

30

以上、本開示の実施形態について説明したが、本発明の技術的範囲は、本実施形態の説明によって限定的に解釈されるべきではない。本実施形態は一例であって、特許請求の範囲に記載された発明の範囲内において、様々な実施形態の変更が可能であることが当業者によって理解されるところである。本発明の技術的範囲は、特許請求の範囲に記載された発明の範囲およびその均等の範囲に基づいて定められるべきである。

【0161】

例えば、本実施形態では、操作オブジェクト（武器オブジェクトW）と対象オブジェクト（敵オブジェクトEまたは攻撃オブジェクトA）との間のコリジョン制御について説明したが、上述したコリジョン制御の対象となる組み合わせは、上記例に限られない。例えば、素手（手オブジェクト400）で敵オブジェクトEを攻撃したり、攻撃オブジェクトAを叩き落としたりするゲームにおいては、プロセッサ10は、手オブジェクト400と敵オブジェクトEまたは攻撃オブジェクトAとの間で、上述したコリジョン制御を実行してもよい。

【0162】

40

また、本実施形態では、プレイヤキャラクタPCがユーザ190の頭部に相当する部分のみである場合について説明したが、プレイヤキャラクタPCは、例えば胴体部および脚部等の頭部以外の部分を含んでもよい。また、HMDシステム100がユーザ190の頭部および手以外の身体の部分の動きのトラッキングが可能なセンサおよびカメラ等を備える場合には、プロセッサ10は、当該部分の動きに応じて、当該部分に対応するプレイヤキャラクタPCの部分を動かしてもよい。

【0163】

50

また、本実施形態で説明したフローチャートに示される処理手順は一例であり、処理の一部が省略または変更されてもよいし、他の処理が付け加えられてもよいし、処理の順序が変更されてもよい。また、上述したプロセッサ10の処理において、二つの数値の大小関係を比較する際には、「以上」及び「よりも大きい」という二つの基準のどちらを用い

てもよく、「以下」及び「未満」の二つの基準のうちのどちらを用いてもよい。このような基準の選択は、二つの数値の大小関係を比較する処理についての技術的意義を変更するものではない。

#### 【0164】

また、本実施形態では、ユーザ190の手の動きを示すコントローラ160の動きに応じて、手オブジェクトの移動が制御されているが、ユーザ190の手自身の移動量に応じて、仮想空間内における手オブジェクトの移動が制御されてもよい。例えば、コントローラ160を用いる代わりに、ユーザの手指に装着されるグローブ型デバイスおよび指輪型デバイス等が用いられてもよい。この場合、HMDセンサ120により、ユーザ190の手の位置および移動量等を検出することができるとともに、ユーザ190の手指の動きおよび状態等を検出することができる。また、HMDセンサ120の代わりに、ユーザ190の手（手指を含む）を撮像するように構成されたカメラによって、ユーザ190の手指の動きおよび状態等が検出されてもよい。カメラを用いてユーザ190の手を撮像することにより、ユーザ190の手指に直接何らのデバイスを装着させる必要がなくなる。この場合、ユーザ190の手が表示された画像データに基づいて、ユーザ190の手の位置および移動量等を検出することができるとともに、ユーザ190の手指の動きおよび状態等を検出することができる。

10

#### 【0165】

また、本実施形態では、ユーザ190の手の動きに連動する手オブジェクトが操作オブジェクトとして用いられたが、本実施形態はこれに限定されない。例えば、ユーザ190の足の動きに連動する足オブジェクトが、手オブジェクトの代わりに、あるいは手オブジェクトとともに、操作オブジェクトとして用いられてもよい。

20

#### 【0166】

また、本実施形態では、仮想カメラ1によって定義されるユーザの視野を仮想空間2におけるプレイヤキャラクタPCの視野と一致させることで、1人称視点における仮想体験がユーザ190に提供されたが、本実施形態はこれに限定されない。例えば、仮想カメラ1がプレイヤキャラクタPCの後方に配置されることで、プレイヤキャラクタPCが視界画像に含まれる3人称視点における仮想体験が、ユーザ190に提供されてもよい。

#### 【0167】

また、仮想空間2において提供されるゲームは、上記実施形態で説明したようなバトルゲームに限らず、種々のジャンルのゲームを含み得る。また、上記ゲームは、ゲームコンテンツ（ゲームソフト）として提供されるゲームに限られない。例えば、上記ゲームは、仮想空間2における複数ユーザ間のチャット（VRチャット）等の通常時にはゲーム要素のないコンテンツにおいて提供されるミニゲーム等であってもよい。

30

#### 【0168】

また、本実施形態においては、HMD装置110によってユーザ190が没入する仮想空間（VR空間）を例示して説明したが、HMD装置110として、透過型のHMD装置を採用してもよい。この場合、透過型のHMD装置を介してユーザ190が視認する現実空間に仮想空間を構成する画像の一部を合成した視界画像を出力することにより、拡張現実（AR：Augmented Reality）空間または複合現実（MR：Mixed Reality）空間における仮想体験をユーザ190に提供してもよい。この場合、手オブジェクト400に代えて、ユーザ190の手の動きに基づいて、仮想空間2内における対象オブジェクトへの作用を生じさせてもよい。具体的には、プロセッサ10は、現実空間におけるユーザ190の手の位置の座標情報を特定するとともに、仮想空間2内における対象オブジェクトの位置を現実空間における座標情報との関係で定義してもよい。これにより、プロセッサ10は、現実空間におけるユーザ190の手と仮想空間2における対象オブジェクトとの位置関係を把握し、ユーザ190の手と対象オブジェクトとの間で上述したコリジョン制御等に対応する処理を実行可能となる。その結果、ユーザ190の手の動きに基づいて対象オブジェクトに作用を与えることが可能となる。

40

#### 【0169】

50

操作オブジェクトの移動速度に基づいてコリジョン効果を決定する手法は限定されない。例えば、操作オブジェクトの移動速度が大きい程、プロセッサ10はコリジョンエリアのサイズを大きくしてもよいし小さくしてもよい。あるいは、操作オブジェクトの移動速度が大きい程、プロセッサ10は所定のパラメータ（例えば、対象オブジェクトに関連付けられたパラメータ、またはユーザに関連付けられたパラメータ）を大きくしてもよいし小さくしてもよい。

#### 【0170】

本明細書に開示された主題は、例えば、以下のような項目として示される。

##### （項目1）

表示部（ディスプレイ112）を備えるヘッドマウントデバイス（HMD装置110）を介してユーザ190に仮想体験を提供するためにコンピュータ200によって実行される情報処理方法であって、10

前記ユーザ190の手に関連付けられた手オブジェクト400および前記手オブジェクト400により操作されるオブジェクト（本実施形態では、武器オブジェクトW）の少なくとも一方を含む操作オブジェクトと、前記操作オブジェクトと相互に作用する対象オブジェクト（本実施形態では、敵オブジェクトEまたは攻撃オブジェクトA）と、を含む仮想空間2を規定する仮想空間データを生成するステップ（図10のS1）と、

前記ヘッドマウントデバイスの動きと前記ユーザ190の手の動きとを検出するステップ（図10のS5, S7）と、

前記ユーザ190の手の動きに応じて、前記操作オブジェクトを動かすステップ（図10のS8）と、20

前記操作オブジェクトの移動速度に基づいて、前記操作オブジェクトに関連付けられるコリジョン効果を決定するステップ（図12のS901およびS902、または、図14のS912およびS913）と、

前記操作オブジェクトと前記対象オブジェクトとが衝突した場合に、前記操作オブジェクトと前記対象オブジェクトとの関係に応じた処理を実行するステップ（図12のS906または図14のS917）と、

前記ヘッドマウントデバイスの動きと前記仮想空間データとに基づいて視界画像を生成し、前記表示部に前記視界画像を表示させるステップ（図10のS10）と、30  
を含む、情報処理方法。

この情報処理方法によれば、操作オブジェクトの移動速度に応じて、コリジョン効果を決定することにより、ユーザに対してより直感的な仮想体験を提供し得る。その結果、ユーザの仮想体験のエンタテイメント性を向上させ得る。

##### （項目2）

前記コリジョン効果は、前記操作オブジェクトに関連付けられるコリジョンエリアのサイズ、または、前記コリジョンエリアに基づいて前記操作オブジェクトと前記対象オブジェクトとが衝突したと判定された場合に生じさせる所定のパラメータの変動量である、項目1に記載の情報処理方法。

この情報処理方法によれば、操作オブジェクトの移動速度に応じて、コリジョンエリアのサイズまたは効果（パラメータの変動量）を決定することにより、ユーザに対してより直感的な仮想体験を提供し得る。その結果、ユーザの仮想体験のエンタテイメント性を向上させ得る。40

##### （項目3）

前記操作オブジェクトの移動速度が大きい程、前記コリジョンエリアのサイズを大きくまたは小さくする、

項目2の情報処理方法。

この情報処理方法によれば、操作オブジェクトの移動速度に応じて、当該操作オブジェクトと他のオブジェクトとの衝突のし易さを調整することで、ユーザに対してより直感的な仮想体験を提供し得る。

##### （項目4）

50

20

30

40

50

前記所定のパラメータは、前記対象オブジェクトに関連付けられたパラメータであり、前記操作オブジェクトの移動速度が大きい程、前記所定のパラメータの変動量を大きくまたは小さくする、

項目 2 の情報処理方法。

この情報処理方法によれば、例えば操作オブジェクトである武器オブジェクトの移動速度が大きい程、対象オブジェクトである敵オブジェクトに与えるダメージ量を大きくまたは小さくするといったことが可能となる。これにより、ダイナミックに身体を動かしてゲームを楽しむことをユーザに促すことができる。

(項目 5)

前記所定のパラメータは、前記ユーザに関連付けられたパラメータであり、

前記操作オブジェクトの移動速度が大きい程、前記所定のパラメータの変動量を小さくまたは大きくする、

項目 2 の情報処理方法。

この情報処理方法によれば、例えば操作オブジェクトである武器オブジェクトの移動速度が大きい程、対象オブジェクトである攻撃オブジェクトから受けるプレイヤキャラクタの被ダメージ量を小さくまたは大きくするといったことが可能となる。これにより、ダイナミックに身体を動かしてゲームを楽しむことをユーザに促すことができる。

(項目 6)

前記操作オブジェクトの移動速度が予め定められた閾値以下の場合には、前記コリジョンエリアを無効化する、

項目 1 ~ 5 のいずれかの情報処理方法。

この情報処理方法によれば、ユーザが操作オブジェクトを実質的に移動させない消極的なプレイ(例えば、武器オブジェクトを構えてひたすら防御するといったプレイ)をすることを抑制することができる。

(項目 7)

前記対象オブジェクトに対する前記操作オブジェクトの移動方向に基づいて、前記コリジョン効果を決定するステップをさらに含む、

項目 1 ~ 5 のいずれかの情報処理方法。

この情報処理方法によれば、対象オブジェクトに対する操作オブジェクトの移動方向に基づいて、コリジョン効果を適切に決定することができる。これにより、ゲーム性をより一層向上させ得る。

(項目 8)

項目 1 ~ 7 のいずれかの情報処理方法をコンピュータに実行させる、プログラム。

(項目 9)

少なくともメモリと、前記メモリに結合されたプロセッサとを備え、前記プロセッサの制御により項目 1 ~ 7 のいずれかの情報処理方法を実行する、装置。

【符号の説明】

【0 1 7 1】

1 ... 仮想カメラ、2 ... 仮想空間、5 ... 基準視線、10 ... プロセッサ、11 ... メモリ、12 ... ストレージ、13 ... 入出力インターフェース、14 ... 通信インターフェース、15 ... バス、19 ... ネットワーク、21 ... 中心、22 ... 仮想空間画像、23 ... 視界領域、24 ... 領域、31 ... フレーム、32 ... 天面、33, 34, 36, 37 ... ボタン、35 ... 赤外線 LED、38 ... アナログスティック、100 ... HMD システム、110 ... HMD 装置、112 ... ディスプレイ、114 ... センサ、120 ... HMD センサ、130 ... モーションセンサ、140 ... 注視センサ、150 ... サーバ、160 ... コントローラ、160L ... 左コントローラ、160R ... 右コントローラ、190 ... ユーザ、200 ... コンピュータ、220 ... 表示制御モジュール、221 ... 仮想カメラ制御モジュール、222 ... 視界領域決定モジュール、223 ... 視界画像生成モジュール、224 ... 基準視線特定モジュール、230 ... 仮想空間制御モジュール、231 ... 仮想空間定義モジュール、232 ... 仮想オブジェクト制御モジュール、233 ... 操作オブジェクト制御モジュール、234 ... コリジョン制御

10

20

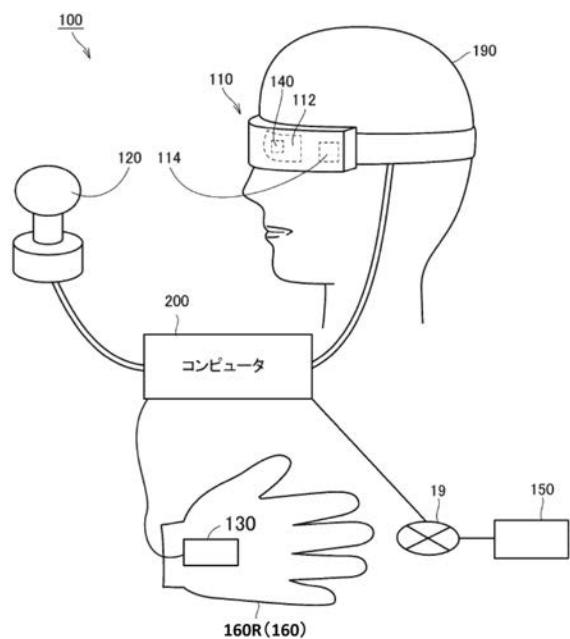
30

40

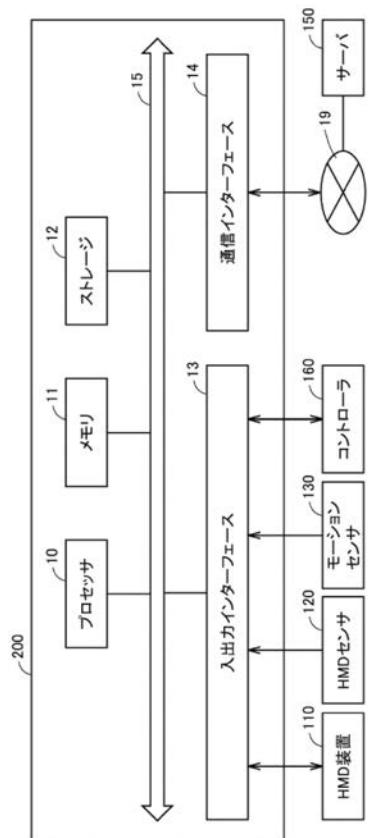
50

モジュール、235...アクション評価モジュール、240...メモリモジュール、241...コンテンツ情報、242...オブジェクト情報、243...ユーザ情報、250...通信制御モジュール、400...手オブジェクト、400L...左手オブジェクト、400R...右手オブジェクト、500...対象オブジェクト、810...右手、A...攻撃オブジェクト、C A, C B, C C, C D, C E, C F, C F 1, C F 2...コリジョンエリア、D...防具オブジェクト、E...敵オブジェクト、W...武器オブジェクト、P C...プレイヤキャラクタ。

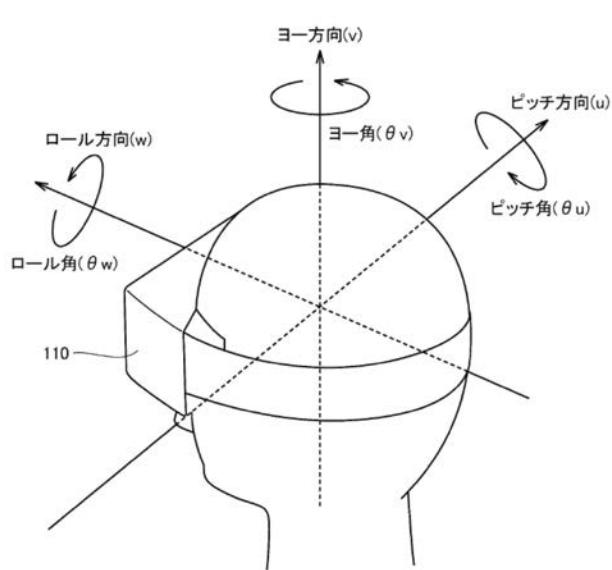
【図1】



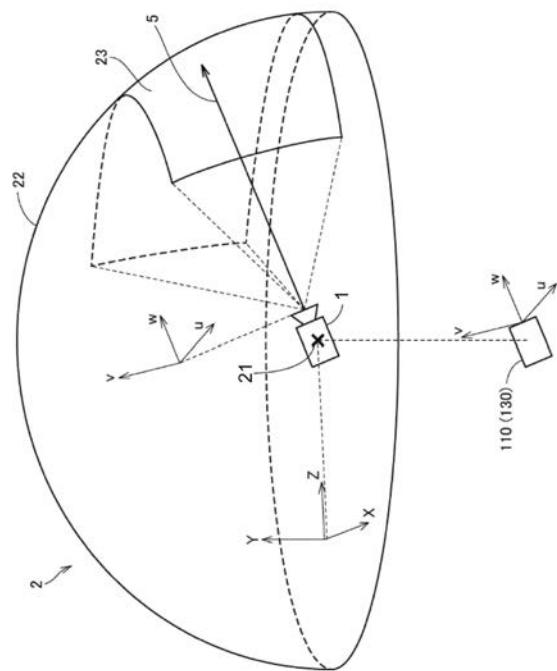
【図2】



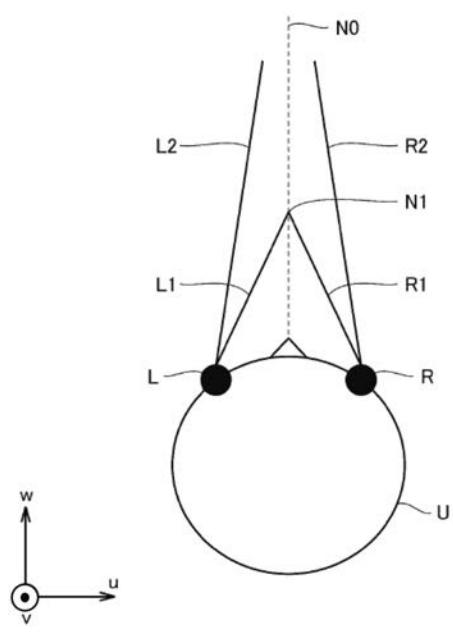
【図3】



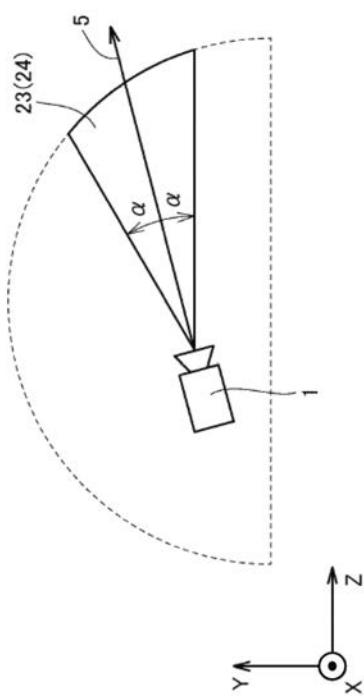
【図4】



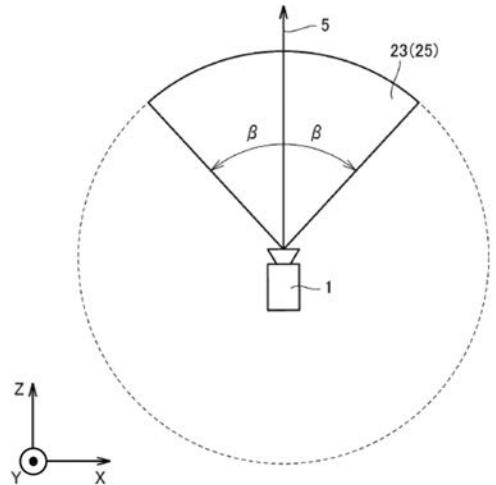
【図5】



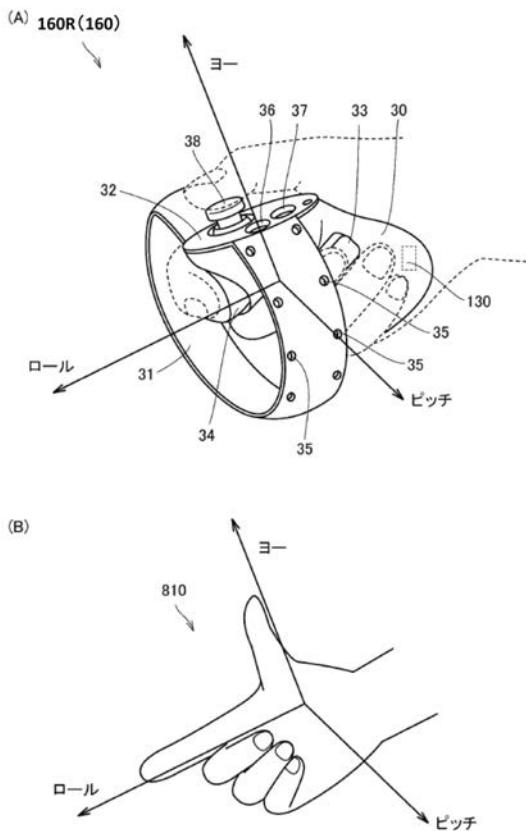
【図6】



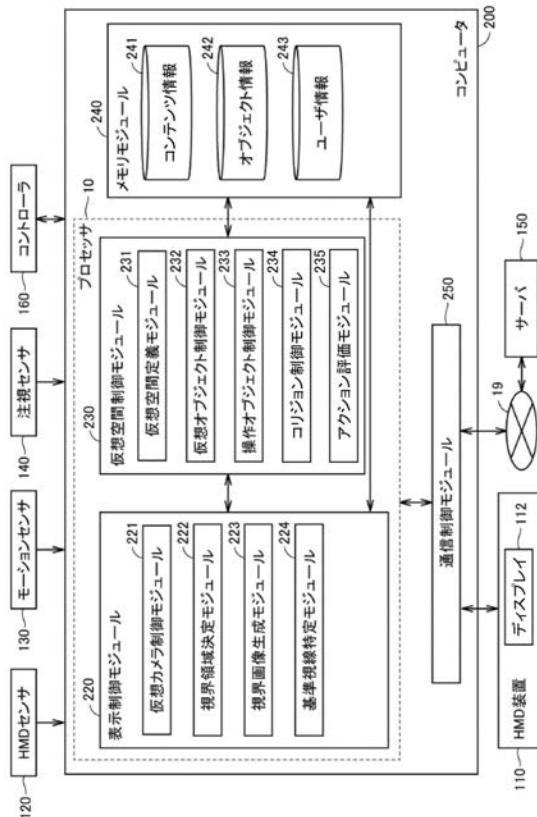
【図 7】



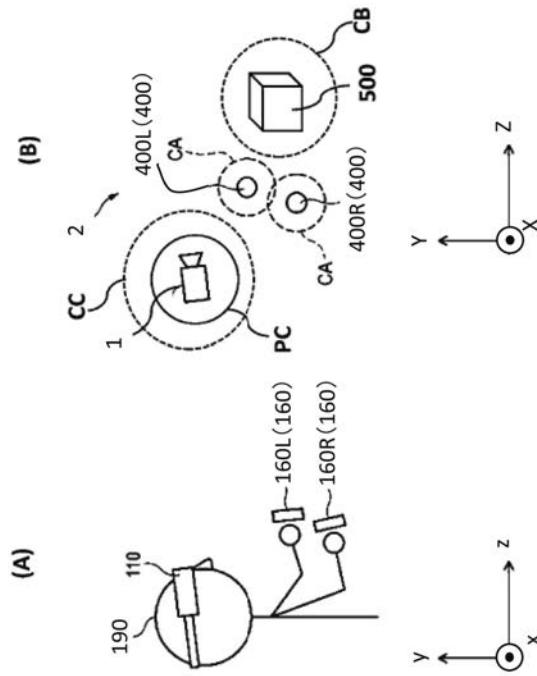
【図 8】



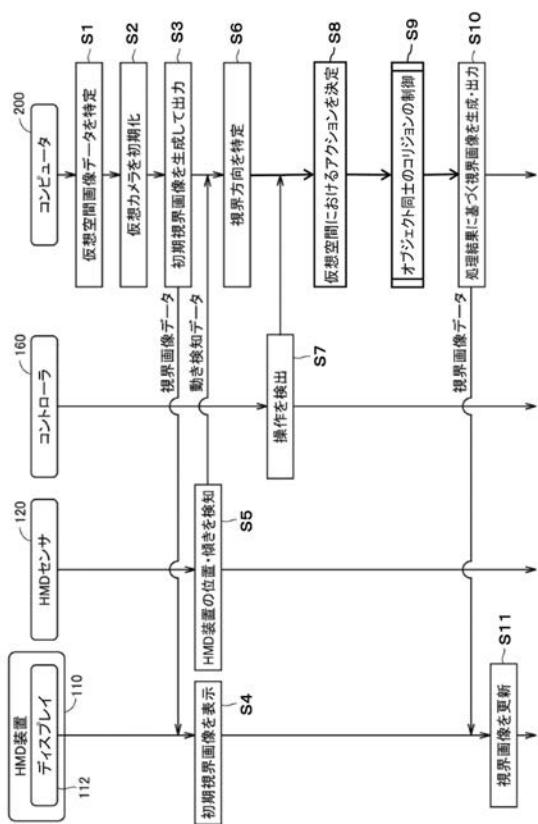
【図 9】



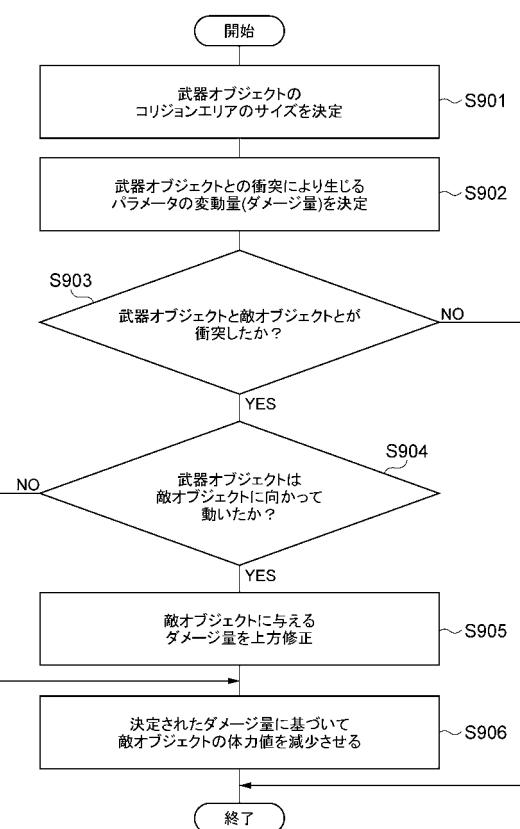
【図 10】



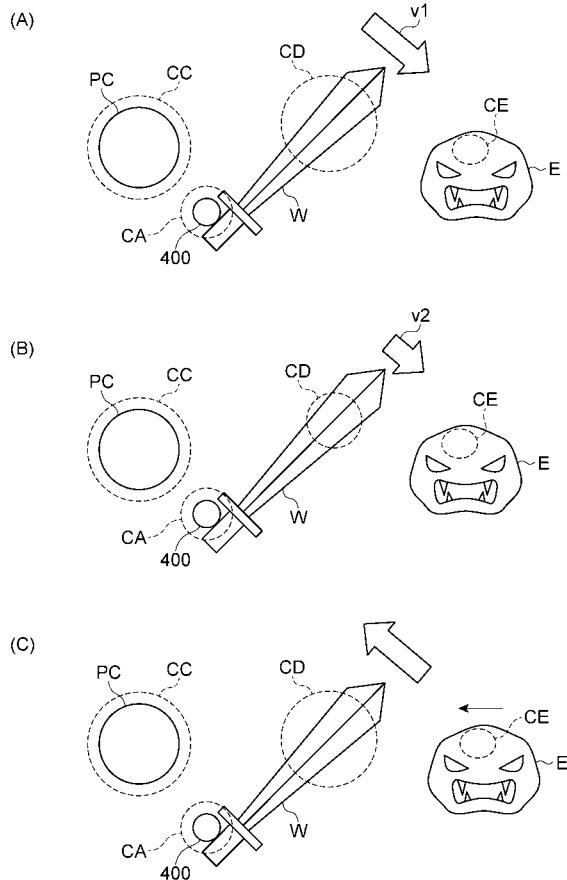
【図 1 1】



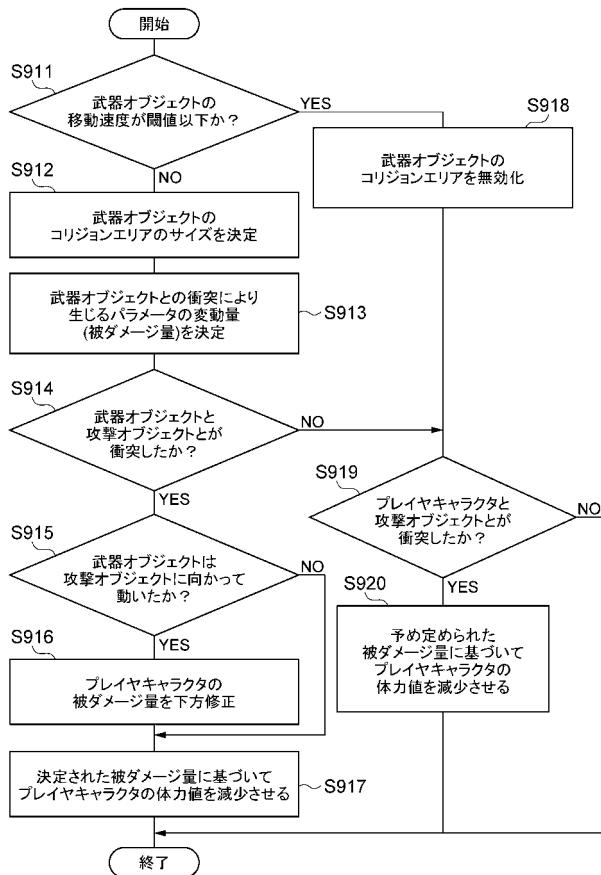
【図 1 2】



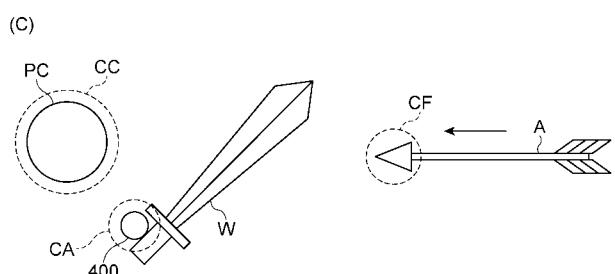
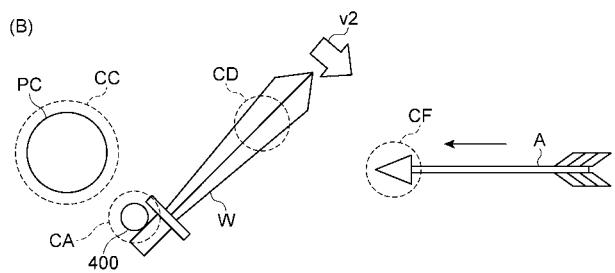
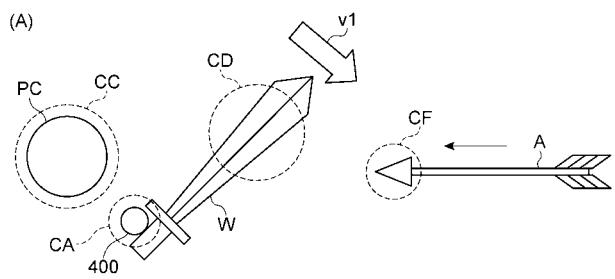
【図 1 3】



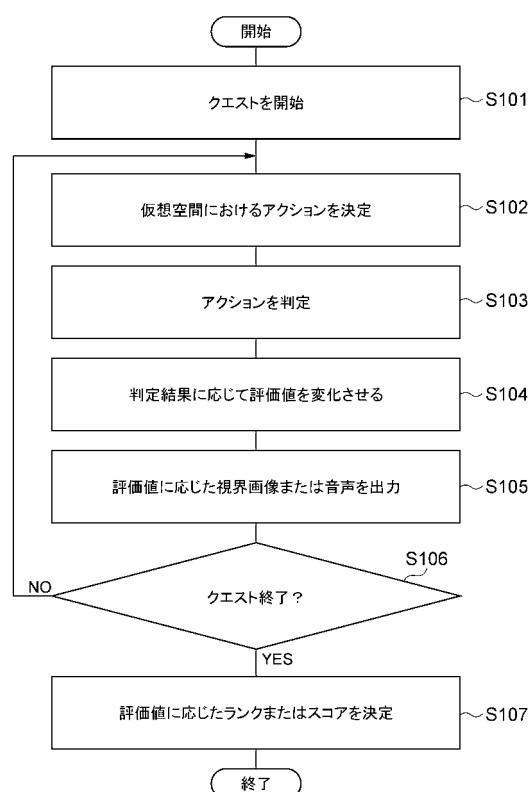
【図 1 4】



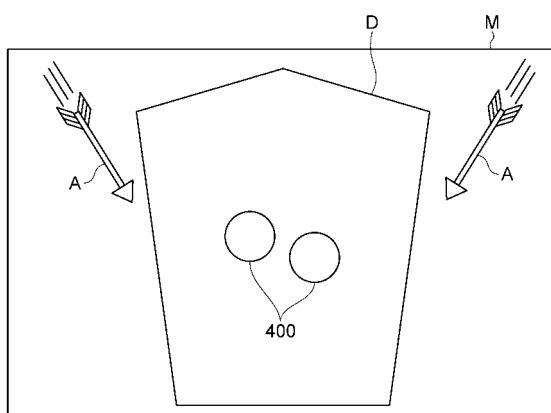
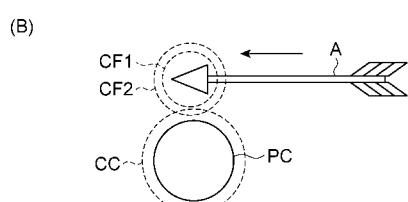
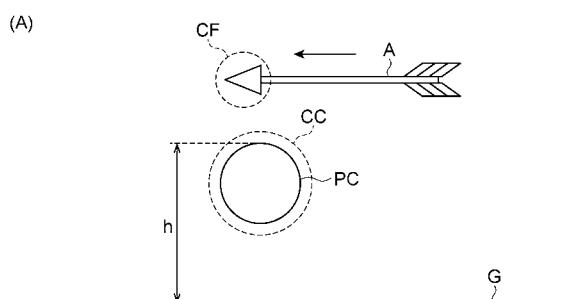
【図 1 5】



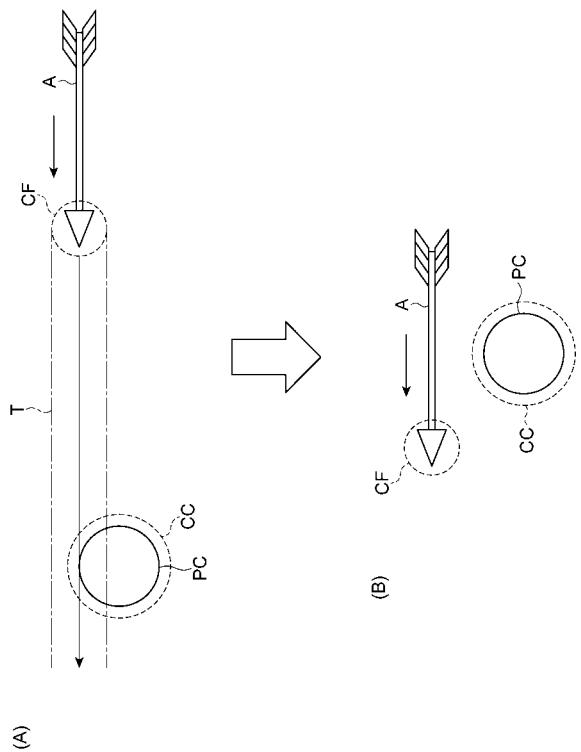
【図 1 6】



【図 1 8】



【図 19】



## フロントページの続き

---

(51) Int.Cl.	F I	テーマコード(参考)
<b>A 6 3 F 13/577 (2014.01)</b>	A 6 3 F 13/428	
<b>A 6 3 F 13/52 (2014.01)</b>	A 6 3 F 13/577	
<b>A 6 3 F 13/211 (2014.01)</b>	A 6 3 F 13/52	
	A 6 3 F 13/211	