

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4982862号  
(P4982862)

(45) 発行日 平成24年7月25日(2012.7.25)

(24) 登録日 平成24年5月11日(2012.5.11)

(51) Int.Cl. F I  
**G06T 19/00 (2011.01)** G O 6 T 17/40 F  
**G06T 15/50 (2011.01)** G O 6 T 15/50

請求項の数 21 (全 36 頁)

(21) 出願番号	特願2007-233315 (P2007-233315)	(73) 特許権者	000134855 株式会社バンダイナムコゲームス 東京都品川区東品川4丁目5番15号
(22) 出願日	平成19年9月7日(2007.9.7)	(73) 特許権者	310021766 株式会社ソニー・コンピュータエンタテインメント 東京都港区港南1丁目7番1号
(65) 公開番号	特開2009-64356 (P2009-64356A)	(74) 代理人	100104710 弁理士 竹腰 昇
(43) 公開日	平成21年3月26日(2009.3.26)	(74) 代理人	100124682 弁理士 黒田 泰
審査請求日	平成22年8月31日(2010.8.31)	(74) 代理人	100090479 弁理士 井上 一

最終頁に続く

(54) 【発明の名称】 プログラム、情報記憶媒体及び画像生成システム

(57) 【特許請求の範囲】

【請求項1】

多眼立体視画像の第1の視点画像を生成するための第1の視点用コマンドが書き込まれる第1の視点用コマンドバッファと、

前記多眼立体視画像の第2の視点画像を生成するための第2の視点用コマンドが書き込まれる第2の視点用コマンドバッファと、

前記多眼立体視画像を生成するための共通コマンドが書き込まれる共通コマンドバッファと、

前記第1の視点用コマンド、前記第2の視点用コマンド、前記共通コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファ、前記共通コマンドバッファに書き込むコマンド生成部と、

前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファ、前記共通コマンドバッファに書き込まれたコマンドを実行するコマンド実行部として、

コンピュータを機能させることを特徴とするプログラム。

【請求項2】

請求項1において、

前記コマンド生成部は、

前記コマンド実行部により実行される各処理のうち、前記第1の視点用コマンド、前記第2の視点用コマンドが必要な処理については、前記第1の視点用コマンド、前記第2の視点用コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマ

ンドバッファに書き込み、前記第1の視点用コマンド、前記第2の視点用コマンドが必要ではない処理については、前記共通コマンドを生成して、前記共通コマンドバッファに書き込むことを特徴とするプログラム。

【請求項3】

請求項1又は2において、

前記コマンド生成部は、

描画処理を行うハードウェアの制御コマンド及び影の生成処理用のコマンドの少なくとも1つを、前記共通コマンドとして、前記共通コマンドバッファに書き込むことを特徴とするプログラム。

【請求項4】

請求項1乃至3のいずれかにおいて、

前記コマンド生成部は、

前記コマンド実行部により実行される各処理のうち、多眼立体視の第1、第2の視点で内容が異なる処理については、異なる内容の前記第1の視点用コマンド、前記第2の視点用コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込み、前記第1、第2の視点で内容が同じになる処理については、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込むことを特徴とするプログラム。

【請求項5】

請求項4において、

前記コマンド生成部は、

カメラ依存パラメータの設定コマンドについては、異なる内容の前記第1の視点用コマンド、前記第2の視点用コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込み、カメラ非依存パラメータの設定コマンドについては、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込むことを特徴とするプログラム。

【請求項6】

請求項5において、

前記コマンド生成部は、

前記カメラ依存パラメータ又は前記カメラ非依存パラメータが設定されるシェーダの設定コマンドを、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込むことを特徴とするプログラム。

【請求項7】

請求項5又は6において、

前記コマンド生成部は、

ワールド座標系からカメラ座標系への変換行列、ワールド座標系から射影座標系への変換行列、及びカメラの位置ベクトルの少なくとも1つの設定コマンドを、前記カメラ依存パラメータの設定コマンドとして、異なる内容の前記第1の視点用コマンド、前記第2の視点用コマンドを前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込むことを特徴とするプログラム。

【請求項8】

請求項5乃至7のいずれかにおいて、

前記コマンド生成部は、

ローカル座標系からワールド座標系への変換行列、モーション用行列、ライトの位置ベクトル、環境色、スペキュラ色、フォグ色、頂点データ、及びテクスチャデータの少なくとも1つの設定コマンドを、前記カメラ非依存パラメータの設定コマンドとして、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドを前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込むことを特徴とするプログラム。

10

20

30

40

50

**【請求項 9】**

請求項 1 乃至 8 のいずれかにおいて、  
前記コマンド実行部は、

前記第 1 の視点用コマンドバッファに書き込まれた前記第 1 の視点用コマンドを実行した後に、前記第 2 の視点用コマンドバッファに書き込まれた前記第 2 の視点用コマンドを実行し、その後に、前記共通コマンドバッファに書き込まれた前記共通コマンドを実行することを特徴とするプログラム。

**【請求項 10】**

請求項 9 において、  
前記コマンド実行部は、

前記第 1 の視点用コマンド、前記第 2 の視点用コマンドの実行後に、多眼立体視画像の第 1 の視点画像と第 2 の視点画像の合成コマンド及びバックバッファとフロントバッファの切り替えコマンドの少なくとも 1 つを、前記共通コマンドとして実行することを特徴とするプログラム。

10

**【請求項 11】**

請求項 9 又は 10 において、  
前記コマンド実行部は、

前記第 1 の視点用コマンド、前記第 2 の視点用コマンドの実行前に、前記共通コマンドバッファに書き込まれた前記共通コマンドを実行することを特徴とするプログラム。

**【請求項 12】**

請求項 11 において、  
前記コマンド実行部は、

前記第 1 の視点用コマンド、前記第 2 の視点用コマンドの実行前に、ハードウェア制御コマンド及び影生成処理用コマンドの少なくとも 1 つを、前記共通コマンドとして実行することを特徴とするプログラム。

20

**【請求項 13】**

請求項 1 乃至 12 のいずれかにおいて、  
前記コマンド生成部は、

前記第 1 の視点用コマンドバッファから前記第 2 の視点用コマンドバッファに対して、前記コマンド実行部の処理をジャンプさせるためのコマンドを、前記第 1 の視点用コマンドバッファに書き込み、前記第 2 の視点用コマンドバッファから前記共通コマンドバッファに対して、前記コマンド実行部の処理をジャンプさせるためのコマンドを、前記第 2 の視点用コマンドバッファに書き込むことを特徴とするプログラム。

30

**【請求項 14】**

請求項 13 において、  
前記コマンド生成部は、

前記共通コマンドバッファから前記第 1 の視点用コマンドバッファに対して、前記コマンド実行部の処理をジャンプさせるためのコマンドを、前記共通コマンドバッファに書き込むことを特徴とするプログラム。

**【請求項 15】**

請求項 1 乃至 14 のいずれかにおいて、

多眼立体視の第 1 の視点と第 2 の視点とで共通のカリング処理を行うカリング処理部として、

コンピュータを機能させ、  
前記コマンド生成部は、

前記共通のカリング処理により得られる頂点データの設定コマンドを、同じ内容の前記第 1 の視点用コマンド、前記第 2 の視点用コマンドとして生成して、前記第 1 の視点用コマンドバッファ、前記第 2 の視点用コマンドバッファに書き込むことを特徴とするプログラム。

40

**【請求項 16】**

50

多眼立体視画像の第1の視点画像を生成するための第1の視点用コマンドと前記多眼立体視画像の第2の視点画像を生成するための第2の視点用コマンドを生成するコマンド生成部と、

前記コマンド生成部により生成されたコマンドを実行するコマンド実行部と、

多眼立体視の第1視点用、第2視点用に共通のカリング処理用ボリュームを用いて、多眼立体視の第1の視点と第2の視点とで共通のカリング処理を行うカリング処理部として

コンピュータを機能させ、

前記コマンド生成部は、

前記共通のカリング処理により得られる頂点データの設定コマンドを、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドとして生成し、

前記コマンド実行部は、

生成された前記第1の視点用コマンド、前記第2の視点用コマンドを実行することを特徴とするプログラム。

【請求項17】

請求項16において、

多眼立体視の第1視点用、第2視点用に共通の前記カリング処理用ボリュームは、第1視点用ビューボリュームと第2視点用ビューボリュームを内包するボリュームであり、

前記コマンド生成部は、

前記カリング処理用ボリュームに存在しないオブジェクトを削除したオブジェクトリストに基づいて頂点データを生成し、生成された頂点データを設定するためのコマンドであるカメラ非依存パラメータ設定コマンドを、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドとして生成することを特徴とするプログラム。

【請求項18】

コンピュータ読み取り可能な情報記憶媒体であって、請求項1乃至17のいずれかに記載のプログラムを記憶したことを特徴とする情報記憶媒体。

【請求項19】

多眼立体視のための画像生成システムであって、

多眼立体視画像の第1の視点画像を生成するための第1の視点用コマンドが書き込まれる第1の視点用コマンドバッファと、

前記多眼立体視画像の第2の視点画像を生成するための第2の視点用コマンドが書き込まれる第2の視点用コマンドバッファと、

前記多眼立体視画像を生成するための共通コマンドが書き込まれる共通コマンドバッファと、

前記第1の視点用コマンド、前記第2の視点用コマンド、前記共通コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファ、前記共通コマンドバッファに書き込むコマンド生成部と、

前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファ、前記共通コマンドバッファに書き込まれたコマンドを実行するコマンド実行部と、

を含むことを特徴とする画像生成システム。

【請求項20】

多眼立体視のための画像生成システムであって、

多眼立体視画像の第1の視点画像を生成するための第1の視点用コマンドと前記多眼立体視画像の第2の視点画像を生成するための第2の視点用コマンドを生成するコマンド生成部と、

前記コマンド生成部により生成されたコマンドを実行するコマンド実行部と、

多眼立体視の第1視点用、第2視点用に共通のカリング処理用ボリュームを用いて、多眼立体視の第1の視点と第2の視点とで共通のカリング処理を行うカリング処理部とを含み、

前記コマンド生成部は、

10

20

30

40

50

前記共通のカリング処理により得られる頂点データの設定コマンドを、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドとして生成し、

前記コマンド実行部は、

生成された前記第1の視点用コマンド、前記第2の視点用コマンドを実行することを特徴とする画像生成システム。

【請求項21】

請求項20において、

第1視点用、第2視点用に共通の前記カリング処理用ボリュームは、第1視点用ビューボリュームと第2視点用ビューボリュームを内包するボリュームであり、

前記コマンド生成部は、

前記カリング処理用ボリュームに存在しないオブジェクトを削除したオブジェクトリストに基づいて頂点データを生成し、生成された頂点データを設定するためのコマンドであるカメラ非依存パラメータ設定コマンドを、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドとして生成することを特徴とする画像生成システム。

10

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、プログラム、情報記憶媒体及び画像生成システムに関する。

【背景技術】

【0002】

従来より、ゲームや映画などの分野において、臨場感溢れる画像を生成するシステムとして、立体視画像の生成が可能な画像生成システムが注目を浴びている。このような画像生成システムとしては例えば特許文献1に開示される従来技術がある。この画像生成システムでは、右目用画像と左目用画像を用意し、これらの画像を表示画面上にフレーム毎（例えば1/60秒毎）に交互に表示する。そして観者は専用の眼鏡を装着し、右目は右目用画像のみを見て、左目は左目用画像のみを見ることで、立体視を実現する。

20

【0003】

しかしながら、従来の立体視の画像生成システムでは、右目用画像の生成に必要な処理と左目用画像の生成に必要な処理を別個独立に行っていた。従って、本来は共通化できるはずの処理が重複して行われてしまう事態が生じ、処理の効率化が不十分であるという課題があった。

30

【特許文献1】特開2004-178581号公報

【発明の開示】

【発明が解決しようとする課題】

【0004】

本発明の幾つかの態様によれば、処理の無駄を防止して多眼立体視画像の効率的な生成を可能にするプログラム、情報記憶媒体及び画像生成システムを提供できる。

【課題を解決するための手段】

【0005】

本発明は、多眼立体視の第1の視点用コマンドが書き込まれる第1の視点用コマンドバッファと、多眼立体視の第2の視点用コマンドが書き込まれる第2の視点用コマンドバッファと、共通コマンドが書き込まれる共通コマンドバッファと、前記第1の視点用コマンド、前記第2の視点用コマンド、前記共通コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファ、前記共通コマンドバッファに書き込むコマンド生成部と、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファ、前記共通コマンドバッファに書き込まれたコマンドを実行するコマンド実行部とを含む画像生成システムに関する。また本発明は、上記各部としてコンピュータを機能させるプログラム、又は該プログラムを記憶したコンピュータ読み取り可能な情報記憶媒体に関する。

40

【0006】

50

本発明によれば、第1の視点用コマンドバッファ、第2の視点用コマンドバッファ、共通コマンドバッファが、別個に用意される。そしてコマンド生成部は、第1の視点用コマンド、第2の視点用コマンド、共通コマンドを生成して、第1の視点用コマンドバッファ、第2の視点用コマンドバッファ、共通コマンドバッファに書き込む。一方、コマンド実行部は、第1の視点用コマンドバッファ、第2の視点用コマンドバッファ、共通コマンドバッファから、第1の視点用コマンド、第2の視点用コマンド、共通コマンドを読み出して、実行する。このようにすれば、例えば第1の視点と第2の視点とで共通で行うことができる処理を、重複して行わなくても済むようになるため、処理の無駄が防止され、多眼立体視画像の効率的な生成が可能になる。

【0007】

10

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド生成部は、前記コマンド実行部により実行される各処理のうち、前記第1の視点用コマンド、前記第2の視点用コマンドが必要な処理については、前記第1の視点用コマンド、前記第2の視点用コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込み、前記第1の視点用コマンド、前記第2の視点用コマンドが必要ではない処理については、前記共通コマンドを生成して、前記共通コマンドバッファに書き込んでよい。

【0008】

このようにすれば、第1、第2の視点用コマンドが必要な処理についてだけ、第1、第2の視点用コマンドが生成されて第1、第2の視点用コマンドバッファに書き込まれ、その他の共通コマンドについては共通コマンドバッファに書き込まれるため、コマンドの生成、書き込み処理を簡素化できる。

20

【0009】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド生成部は、描画処理を行うハードウェアの制御コマンド及び影の生成処理用のコマンドの少なくとも1つを、前記共通コマンドとして、前記共通コマンドバッファに書き込んでよい。

【0010】

このようにすれば、ハードウェア制御コマンドや影生成処理用コマンドについては、第1、第2の視点用コマンドを生成せずに、共通コマンドとして共通コマンドバッファに書き込んで、処理を実行できるようになる。

30

【0011】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド生成部は、前記コマンド実行部により実行される各処理のうち、多眼立体視の第1、第2の視点で内容が異なる処理については、異なる内容の前記第1の視点用コマンド、前記第2の視点用コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込み、前記第1、第2の視点で内容が同じになる処理については、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込んでよい。

【0012】

40

このようにすれば、第1、第2の視点で内容が同じになる処理については、同じ内容の第1、第2の視点用コマンドを生成すれば済むため、処理を効率化できる。

【0013】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド生成部は、カメラ依存パラメータの設定コマンドについては、異なる内容の前記第1の視点用コマンド、前記第2の視点用コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込み、カメラ非依存パラメータの設定コマンドについては、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドを生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込んでよい。

50

## 【 0 0 1 4 】

このようにすれば、カメラ非依存パラメータの設定コマンドについては、同じ内容の第1、第2の視点用コマンドを生成すれば済むため、処理を効率化できる。

## 【 0 0 1 5 】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド生成部は、前記カメラ依存パラメータ又は前記カメラ非依存パラメータが設定されるシェーダの設定コマンドを、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込んでよい。

## 【 0 0 1 6 】

このようにすれば、シェーダにカメラ依存パラメータやカメラ非依存パラメータを設定して、多眼立体視の第1、第2の視点画像をシェーダにより生成できるようになる。

10

## 【 0 0 1 7 】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド生成部は、ワールド座標系からカメラ座標系への変換行列、ワールド座標系から射影座標系への変換行列、及びカメラの位置ベクトルの少なくとも1つの設定コマンドを、前記カメラ依存パラメータの設定コマンドとして、異なる内容の前記第1の視点用コマンド、前記第2の視点用コマンドを前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込んでよい。

## 【 0 0 1 8 】

このようにすれば、これらの変換行列や位置ベクトルなどのカメラ依存パラメータを、第1、第2の視点に応じたパラメータ値に設定にして、第1、第2の視点画像を生成できる。

20

## 【 0 0 1 9 】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド生成部は、ローカル座標系からワールド座標系への変換行列、モーション用行列、ライトの位置ベクトル、環境色、スペキュラ色、フォグ色、頂点データ、及びテクスチャデータの少なくとも1つの設定コマンドを、前記カメラ非依存パラメータの設定コマンドとして、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドを前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込んでよい。

## 【 0 0 2 0 】

このようにすれば、これらのカメラ非依存パラメータを求める処理が、第1、第2の視点で重複して行われてしまう事態を防止できる。

30

## 【 0 0 2 1 】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド実行部は、前記第1の視点用コマンドバッファに書き込まれた前記第1の視点用コマンドを実行した後に、前記第2の視点用コマンドバッファに書き込まれた前記第2の視点用コマンドを実行し、その後に、前記共通コマンドバッファに書き込まれた前記共通コマンドを実行してもよい。

## 【 0 0 2 2 】

このようにすれば、第1の視点用コマンドバッファに書き込まれた第1の視点用コマンドを実行して、第1の視点画像を生成し、第2の視点用コマンドバッファに書き込まれた第2の視点用コマンドを実行して、第2の視点画像を生成し、その後に、共通コマンドを実行できるようになる。

40

## 【 0 0 2 3 】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド実行部は、前記第1の視点用コマンド、前記第2の視点用コマンドの実行後に、多眼立体視画像の第1の視点画像と第2の視点画像の合成コマンド及びバックバッファとフロントバッファの切り替えコマンドの少なくとも1つを、前記共通コマンドとして実行してもよい。

## 【 0 0 2 4 】

50

このようにすれば、第1、第2の視点用コマンドを実行して第1、第2の視点画像を生成した後に、生成された第1、第2の視点画像を合成して多眼立体視画像を生成したり、複数バッファ方式におけるバックバッファとフロントバッファの切り替えを実行できるようになる。

【0025】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド実行部は、前記第1の視点用コマンド、前記第2の視点用コマンドの実行前に、前記共通コマンドバッファに書き込まれた前記共通コマンドを実行してもよい。

【0026】

このようにすれば、第1、第2の視点用コマンドを実行して第1、第2の視点画像を生成する前に、共通コマンドを実行できるようになる。

10

【0027】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド実行部は、前記第1の視点用コマンド、前記第2の視点用コマンドの実行前に、ハードウェア制御コマンド及び影生成処理用コマンドの少なくとも1つを、前記共通コマンドとして実行してもよい。

【0028】

このようにすれば、第1、第2の視点用コマンドを実行して第1、第2の視点画像を生成する前に、共通のハードウェア制御コマンドを実行したり、カメラに依存しない影生成処理を実行することが可能になる。

20

【0029】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド生成部は、前記第1の視点用コマンドバッファから前記第2の視点用コマンドバッファに対して、前記コマンド実行部の処理をジャンプさせるためのコマンドを、前記第1の視点用コマンドバッファに書き込み、前記第2の視点用コマンドバッファから前記共通コマンドバッファに対して、前記コマンド実行部の処理をジャンプさせるためのコマンドを、前記第2の視点用コマンドバッファに書き込んでもよい。

【0030】

このようにすれば、コマンド実行部は、第1の視点用コマンドを第1の視点用コマンドバッファから読み出して実行した後に、ジャンプコマンドを実行することで、第2の視点用コマンドバッファの第2の視点用コマンドを実行できるようになる。

30

【0031】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、前記コマンド生成部は、前記共通コマンドバッファから前記第1の視点用コマンドバッファに対して、前記コマンド実行部の処理をジャンプさせるためのコマンドを、前記共通コマンドバッファに書き込んでもよい。

【0032】

このようにすれば、コマンド実行部は、第2の視点用コマンドを第2の視点用コマンドバッファから読み出して実行した後に、ジャンプコマンドを実行することで、共通コマンドバッファの共通コマンドを実行できるようになる。

40

【0033】

また本発明に係る画像生成システム、プログラム及び情報記憶媒体では、多眼立体視の第1の視点と第2の視点とで共通のカリング処理を行うカリング処理部を含み（カリング処理部としてコンピュータを機能させ）、前記コマンド生成部は、前記共通のカリング処理により得られる頂点データの設定コマンドを、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドとして生成して、前記第1の視点用コマンドバッファ、前記第2の視点用コマンドバッファに書き込んでもよい。

【0034】

このようにすれば、カリング処理を、第1の視点用、第2の視点用に重複して行う必要がなくなるため、処理負荷を軽減できる。

50



## 【 0 0 3 5 】

また本発明は、多眼立体視の第1の視点用コマンドと第2の視点用コマンドを生成するコマンド生成部と、前記コマンド生成部により生成されたコマンドを実行するコマンド実行部と、多眼立体視の第1の視点と第2の視点とで共通のカリング処理を行うカリング処理部とを含み、前記コマンド生成部は、前記共通のカリング処理により得られる頂点データの設定コマンドを、同じ内容の前記第1の視点用コマンド、前記第2の視点用コマンドとして生成し、前記コマンド実行部は、生成された前記第1の視点用コマンド、前記第2の視点用コマンドを実行する画像生成システムに関係する。また本発明は、上記各部としてコンピュータを機能させるプログラム、又は該プログラムを記憶したコンピュータ読み取り可能な情報記憶媒体に関係する。

10

## 【 0 0 3 6 】

本発明によれば、多眼立体視の第1の視点と第2の視点とで共通のカリング処理が実行される。そして、この共通のカリング処理により得られる頂点データの設定コマンドが、同じ内容の第1、第2の視点用コマンドとして生成されて、実行される。このようにすれば、カリング処理を、第1の視点用、第2の視点用に重複して行う必要がなくなるため、処理負荷を軽減できる。

## 【 発明を実施するための最良の形態 】

## 【 0 0 3 7 】

以下、本実施形態について説明する。なお、以下に説明する本実施形態は、特許請求の範囲に記載された本発明の内容を不当に限定するものではない。また本実施形態で説明される構成の全てが、本発明の必須構成要件であるとは限らない。

20

## 【 0 0 3 8 】

## 1. 構成

図1に本実施形態の画像生成システム(ゲームシステム)のブロック図の例を示す。なお本実施形態の画像生成システムは図1の構成要素(各部)の一部を省略した構成としてもよい。

## 【 0 0 3 9 】

操作部160は、プレーヤが操作データを入力するためのものであり、その機能は、方向キー、操作ボタン、アナログスティック、レバー、ステアリング、アクセル、ブレーキ、マイク、或いはタッチパネル型ディスプレイなどにより実現できる。

30

## 【 0 0 4 0 】

記憶部170は、処理部100や通信部196などのワーク領域となるもので、その機能はRAM(DRAM、VRAM)などにより実現できる。この記憶部170は、電源を切るとデータが消えてしまう揮発性のメモリにより構成できるが、補助記憶装置194よりも高速な記憶装置になっている。そしてゲームプログラムや、ゲームプログラムの実行に必要なゲームデータは、この記憶部170に保持される。

## 【 0 0 4 1 】

情報記憶媒体180(コンピュータにより読み取り可能な媒体)は、プログラムやデータなどを格納するものであり、その機能は、光ディスク(CD、DVD)、HDD(ハードディスクドライブ)、或いはメモリ(ROM等)などにより実現できる。処理部100は、情報記憶媒体180に格納されるプログラム(データ)に基づいて本実施形態の種々の処理を行う。即ち情報記憶媒体180には、本実施形態の各部としてコンピュータ(操作部、処理部、記憶部、出力部を備える装置)を機能させるためのプログラム(各部の処理をコンピュータに実行させるためのプログラム)が記憶される。

40

## 【 0 0 4 2 】

表示部190は、本実施形態により生成された画像を出力するものであり、その機能は、CRT、LCD、タッチパネル型ディスプレイ、或いはHMD(ヘッドマウントディスプレイ)などにより実現できる。音出力部192は、本実施形態により生成された音を出力するものであり、その機能は、スピーカ、或いはヘッドフォンなどにより実現できる。

## 【 0 0 4 3 】

50

補助記憶装置 194 (補助メモリ、2次メモリ)は、記憶部 170 の容量を補うために使用される大容量の記憶装置であり、SDメモリーカード、マルチメディアカードなどのメモリーカードや、HDDなどにより実現できる。この補助記憶装置 194 は脱着自在になっているが、内蔵されるものであってもよい。この補助記憶装置 194 は、ゲームの途中結果などのセーブデータや、プレーヤ(ユーザ)の個人的な画像データや音楽データなどを保存するために使用される。

【0044】

通信部 196 は、有線や無線のネットワークを介して外部(例えば他の画像生成システム、サーバ、ホスト装置)との間で通信を行うものであり、その機能は、通信用ASIC又は通信用プロセッサなどのハードウェアや、通信用ファームウェアにより実現できる。

10

【0045】

なお本実施形態の各部としてコンピュータを機能させるためのプログラム(データ)は、サーバ(ホスト装置)が有する情報記憶媒体からネットワーク及び通信部 196 を介して情報記憶媒体 180 (あるいは記憶部 170、補助記憶装置 194)に配信してもよい。このようなサーバ(ホスト装置)による情報記憶媒体の使用も本発明の範囲内に含めることができる。

【0046】

処理部 100 (プロセッサ)は、操作部 160 からの操作データやプログラムなどに基づいて、ゲーム処理、画像生成処理、或いは音生成処理などを行う。処理部 100 は記憶部 170 (主記憶部 172)をワーク領域として各種処理を行う。この処理部 100 の機能は、各種プロセッサ(CPU、GPU等)、ASIC(ゲートアレイ等)などのハードウェアや、プログラムにより実現できる。

20

【0047】

処理部 100 は、ゲーム演算部 102、オブジェクト空間設定部 104、移動体演算部 106、仮想カメラ制御部 108、コマンド生成部 110、コマンド実行部 118を含む。コマンド実行部 118 は描画部 120、音生成部 130を含む。なおこれらの一部を省略する構成としてもよい。

【0048】

ゲーム演算部 102 はゲーム演算処理を行う。ここでゲーム演算としては、ゲーム開始条件が満たされた場合にゲームを開始する処理、ゲームを進行させる処理、ゲーム結果を演算する処理、或いはゲーム終了条件が満たされた場合にゲームを終了する処理などがある。

30

【0049】

オブジェクト空間設定部 104 は、モデルオブジェクト(車、戦闘機、人、ロボット、ミサイル、弾等の移動体)、マップ(地形)、建物、コース(道路)、樹木、壁などの表示物を表す各種オブジェクト(ポリゴン、自由曲面又はサブディビジョンサーフェイスなどのプリミティブ面で構成されるオブジェクト)をオブジェクト空間に配置設定する処理を行う。即ちワールド座標系でのオブジェクトの位置や回転角度(向き、方向と同義)を決定し、その位置(X、Y、Z)にその回転角度(X、Y、Z軸回りの回転角度)でオブジェクトを配置する。具体的には、記憶部 170 のモデルデータ記憶部MSには、オブジェクトの位置、回転角度、移動速度、移動方向等のデータであるオブジェクトデータがオブジェクト番号に対応づけて記憶される。そしてこのオブジェクトデータは、移動体演算部 106 の移動体演算処理等により順次更新される。

40

【0050】

オブジェクト空間設定部 104 はカリング処理部 105 を含む。このカリング処理部 105 は例えばソフトウェアによるカリング処理を行う。具体的には、オブジェクトデータのオブジェクトリストの中から、当該フレームにおいて表示されることがないオブジェクトを削除する処理を行う。例えば当該フレームにおいて仮想カメラの視野範囲内に存在せず、表示処理を行う必要がないオブジェクトを予め削除する。このようにすれば、削除されたオブジェクトを構成するポリゴンについては頂点データ等を作成しなくても済むため

50

、処理負荷を大幅に軽減できる。

【 0 0 5 1 】

移動体演算部（移動体制御部）106は、移動体（移動体オブジェクト）を移動させるための演算を行う。また移動体を動作させるための演算も行う。即ち操作部160によりプレーヤが入力した操作データや、プログラム（移動・動作アルゴリズム）や、各種データ（モーションデータ）などに基づいて、移動体（オブジェクト、モデルオブジェクト）をオブジェクト空間内で移動させたり、移動体を動作（モーション、アニメーション）させる処理を行う。具体的には、移動体の移動情報（位置、回転角度、速度、或いは加速度）や動作情報（パーツオブジェクトの位置、或いは回転角度）を、1フレーム（1/60秒）毎に順次求めるシミュレーション処理を行う。なおフレームは、移動体の移動・動作処理（シミュレーション処理）や画像生成処理を行う時間の単位である。

10

【 0 0 5 2 】

より具体的には、移動体演算部106は、モーションデータ記憶部MTSに記憶されるモーションデータに基づいて、オブジェクトのモーションを再生する処理を行う。即ち、オブジェクト（スケルトン）を構成する各パーツオブジェクト（スケルトンを構成する骨）の位置又は回転角度（方向）等を含むモーションデータを、モーションデータ記憶部MTSから読み出す。そして、オブジェクトの各パーツオブジェクト（骨）を動かすことで（スケルトン形状を変形させることで）、オブジェクトのモーションを再生する。

【 0 0 5 3 】

仮想カメラ制御部108は、オブジェクト空間内の所与（任意）の視点から見える画像を生成するための仮想カメラ（視点）の制御処理を行う。具体的には、仮想カメラの位置（X、Y、Z）又は回転角度（X、Y、Z軸回りの回転角度）を制御する処理（視点位置、視線方向あるいは画角を制御する処理）を行う。

20

【 0 0 5 4 】

例えば仮想カメラにより車、キャラクタ、戦闘機などの移動体を後方から撮影する場合には、移動体の位置又は回転の変化に仮想カメラが追従するように、仮想カメラの位置又は回転角度（仮想カメラの向き）を制御する。この場合には、移動体演算部106で得られた移動体の位置、回転角度又は速度などの情報に基づいて、仮想カメラを制御できる。或いは、仮想カメラを、予め決められた回転角度で回転させたり、予め決められた移動経路で移動させる制御を行ってもよい。この場合には、仮想カメラの位置（移動経路）又は回転角度を特定するための仮想カメラデータに基づいて仮想カメラを制御する。

30

【 0 0 5 5 】

なお本実施形態では仮想カメラ制御部108は、多眼立体視画像を生成するために、複数の視点の仮想カメラの制御を行う。具体的には第1の視点用（例えば右目用）の仮想カメラと第2の視点用（例えば左目用）の仮想カメラを制御して、プレーヤに立体感を感じさせるための両眼視差等を実現する。

【 0 0 5 6 】

コマンド生成部110は、コマンド実行部118（描画プロセッサ、GPU）により実行される各種のコマンドを生成する。

【 0 0 5 7 】

具体的には、記憶部170には、多眼立体視の第1の視点用（例えば右目用）のコマンドが書き込まれる第1の視点用のコマンドバッファCBRや、多眼立体視の第2の視点用（例えば左目用）のコマンドが書き込まれる第2の視点用のコマンドバッファCBLや、共通コマンド（メインコマンド、第1、第2の視点に非依存のコマンド）が書き込まれる共通コマンドバッファCBMが確保（用意）される。例えばn眼立体視（n=2）の場合には、第1～第nの視点用のコマンドバッファが確保される。

40

【 0 0 5 8 】

そしてコマンド生成部110は、第1の視点用コマンド、第2の視点用コマンド、或いは共通コマンドを生成する。そして生成した各々のコマンドを、第1の視点用のコマンドバッファCBR、第2の視点用のコマンドバッファCBL、共通コマンドバッファCBM

50

に書き込む。

【0059】

例えばコマンド生成部110は、コマンド実行部118により実行される各処理のうち、第1、第2の視点用コマンドが必要な処理（例えば第1、第2の視点画像の生成に使用されるコマンド）については、第1、第2の視点用コマンドを生成して、第1、第2の視点用のコマンドバッファCBR、CBLに書き込む。具体的には、コマンド実行部118により実行される各処理のうち、多眼立体視の第1、第2の視点で内容が異なる処理については、異なる内容の第1、第2の視点用コマンドを生成して、コマンドバッファCBR、CBLに書き込み、第1、第2の視点で内容が同じになる処理については、同じ内容の第1、第2の視点用コマンドを生成して、コマンドバッファCBR、CBLに書き込む。

10

【0060】

一方、コマンド生成部110は、第1、第2の視点用コマンドが必要ではない処理（例えば第1、第2の視点画像の生成に使用されないコマンド）については、共通コマンドを生成して、共通コマンドバッファCBMに書き込む。

【0061】

ここで、共通コマンドとしては、例えば描画処理を行うハードウェアの制御コマンドや、ハードウェアでサポートされる影の生成処理用のコマンドなどを想定できる。

【0062】

また、異なる内容の第1、第2の視点用コマンドとしては、カメラ依存パラメータ（カメラの視点、視線方向等に依存するパラメータ）の設定コマンドを想定でき、同じ内容の第1、第2の視点用コマンドとしては、カメラ非依存パラメータ（カメラの視点、視線方向等に依存しないパラメータ）の設定コマンドを想定できる。またこれらのカメラ依存パラメータやカメラ非依存パラメータは、例えばシェーダ（ピクセルシェーダや頂点シェーダのプログラム）に対して設定され、シェーダはこれらのパラメータに基づいてピクセルシェーダや頂点シェーダの処理を行う。

20

【0063】

コマンド実行部118（処理部100のハードウェア部分）は、コマンド生成部110により生成されたコマンドを実行する。具体的には、コマンド生成部110により生成されてコマンドバッファCBR、CBL、CBMに書き込まれたコマンドを実行する。

【0064】

この場合にコマンド実行部118は、コマンドバッファCBRに書き込まれた第1の視点用コマンドを実行した後に、コマンドバッファCBLに書き込まれた第2の視点用コマンドを実行する。そしてその後に、共通コマンドバッファに書き込まれた共通コマンドを実行する。

30

【0065】

第1、第2の視点用コマンドの実行後に実行される共通コマンドとしては、例えば、多眼立体視画像の第1の視点画像と第2の視点画像の合成コマンド（描画バッファへの書き戻しコマンド）や、バックバッファとフロントバッファの切り替えコマンドなどを想定できる。

【0066】

またコマンド実行部118は、第1、第2の視点用コマンドの実行前に、コマンドバッファCBMに書き込まれた共通コマンドを実行するようにしてもよい。この場合に、第1、第2の視点用コマンドの実行前に実行されるコマンドとしては、例えば、ハードウェア制御コマンドや、影生成処理用コマンドなどを想定できる。

40

【0067】

なおカリング処理部105が、多眼立体視の第1の視点と第2の視点とで共通のカリング処理を行う場合に、コマンド生成部110は、共通のカリング処理により得られる頂点データ（ポリゴンデータ）の設定コマンドを、同じ内容の第1、第2の視点用コマンドとして生成する。具体的には、共通のカリング処理により得られる頂点データの設定コマンドを、同じ内容の第1、第2の視点用コマンドとして生成して、コマンドバッファCBR

50

、C B Lに書き込む。そしてコマンド実行部 1 1 8 は、生成された第 1、第 2 の視点用コマンドを実行する。

【 0 0 6 8 】

コマンド実行部 1 1 8 は、描画部 1 2 0 (描画プロセッサ)や音生成部 1 3 0 (サウンドプロセッサ)を含むことができる。例えばコマンド実行部 1 1 8 は、ハードウェアによる専用の処理を行うプロセッサ(メインの処理を行うCPU以外のプロセッサ)により実現できる。

【 0 0 6 9 】

コマンド実行部 1 1 8 が含む描画部 1 2 0 (画像生成部)は、処理部 1 0 0 で行われる種々の処理(ゲーム処理、シミュレーション処理)の結果に基づいて描画処理を行い、これにより画像を生成し、表示部 1 9 0 に出力する。いわゆる 3 次元ゲーム画像を生成する場合には、まずモデル(オブジェクト)の各頂点の頂点データ(頂点の位置座標、テクスチャ座標、色データ、法線ベクトル或いは値等)が生成され、これらのデータに基づいて、頂点処理(頂点シェーダによるシェーディング)が行われる。なお頂点処理を行うに際して、必要に応じてポリゴンを再分割するための頂点生成処理(テッセレーション、曲面分割、ポリゴン分割)を行うようにしてもよい。

10

【 0 0 7 0 】

頂点処理(頂点シェーダ処理)では、頂点処理プログラム(頂点シェーダプログラム、第 1 のシェーダプログラム)に従って、頂点の移動処理や、座標変換(ワールド座標変換、カメラ座標変換)、クリッピング処理、あるいは透視変換等のジオメトリ処理が行われ、その処理結果に基づいて、オブジェクトを構成する頂点群について与えられた頂点データを変更(更新、調整)する。そして、頂点処理後の頂点データに基づいてラスタライズ(走査変換)が行われ、ポリゴン(プリミティブ)の面とピクセルとが対応づけられる。そしてラスタライズに続いて、画像を構成するピクセル(表示画面を構成するフラグメント)を描画するピクセル処理(ピクセルシェーダによるシェーディング、フラグメント処理)が行われる。

20

【 0 0 7 1 】

ピクセル処理(ピクセルシェーダ処理)では、ピクセル処理プログラム(ピクセルシェーダプログラム、第 2 のシェーダプログラム)に従って、テクスチャの読み出し(テクスチャマッピング)、色データの設定/変更、半透明合成、アンチエイリアス等の各種処理を行って、画像を構成するピクセルの最終的な描画色を決定し、透視変換されたモデルの描画色を描画バッファDB(ピクセル単位で画像情報を記憶できるバッファ。VRAM、レンダリングターゲット、フレームバッファ)に出力(描画)する。即ち、ピクセル処理では、画像情報(色、法線、輝度、値等)をピクセル単位で設定あるいは変更するパーピクセル処理を行う。これにより、オブジェクト空間内において仮想カメラ(所与の視点)から見える画像が生成される。

30

【 0 0 7 2 】

なお頂点処理やピクセル処理は、シェーディング言語によって記述されたシェーダプログラムによって、ポリゴン(プリミティブ)の描画処理をプログラム可能にするハードウェア、いわゆるプログラマブルシェーダ(頂点シェーダやピクセルシェーダ)により実現される。プログラマブルシェーダでは、頂点単位の処理やピクセル単位の処理がプログラム可能になることで描画処理内容の自由度が高く、従来のハードウェアによる固定的な描画処理に比べて表現力を大幅に向上させることができる。

40

【 0 0 7 3 】

描画部 1 2 0 は、ライティング処理部 1 2 2、テクスチャマッピング部 1 2 4 を含む。なおこれらの一部を省略する構成としてもよい。

【 0 0 7 4 】

ライティング処理部 1 2 2 は、照明モデル等に基づくライティング処理(シェーディング処理)を行う。具体的にはこのライティング処理は、光源情報(光源ベクトル、光源色、明るさ、光源タイプ等)、仮想カメラ(視点)の視線ベクトル、オブジェクト(半透明

50

オブジェクト)の法線ベクトル、オブジェクトのマテリアル(色、材質)などを用いて行われる。なお照明モデルとしては、アンビエント光とディフューズ光だけを考慮したランバードの拡散照明モデルや、アンビエント光、ディフューズ光に加えてスペキュラ光も考慮するフォンの照明モデルやプリン・フォンの照明モデルなどがある。

【0075】

テクスチャマッピング部124は、オブジェクト(ポリゴン)にテクスチャをマッピングする処理を行う。ここでテクスチャマッピング処理は、テクスチャ記憶部TSに記憶されるテクスチャ(テクセル値)をオブジェクトにマッピングする処理である。具体的には、オブジェクト(プリミティブ面)の頂点やピクセルに設定(付与)されるテクスチャ座標等を用いてテクスチャ記憶部TSからテクスチャ(色、値などの表面プロパティ)を読み出す。そして2次元の画像又はパターンであるテクスチャをオブジェクトにマッピングする。この場合に、ピクセルとテクセルとを対応づける処理やバイリニア補間(広義にはテクセル補間)などを行う。

10

【0076】

そして本実施形態では描画部120は、多眼立体視画像(n眼立体視画像。n=2)を生成するための描画処理を行う。具体的には描画部120は、多眼立体視(2眼立体視)の第1の視点画像(例えば右目用画像)を高さ方向(縦方向)において例えば1/2に圧縮(縮小)して、バッファTBの第1の領域AR1(例えば左半分側の領域)に書き込む。また多眼立体視の第2の視点画像(例えば左目用画像)を高さ方向(縦方向)において例えば1/2に圧縮(縮小)して、バッファTBの第2の領域AR2(例えば右半分側の領域)に書き込む。そしてバッファTBから画像(AR1、AR2の画像)を読み出すことで、第1の領域AR1に書き込まれた第1の視点画像のピクセル画像(R、G、B)と第2の領域AR2に書き込まれた第2の視点画像のピクセル画像とが、ライン毎(例えば走査ライン毎、ビットマップライン毎)に交互に表示される多眼立体視画像を生成する。

20

【0077】

更に具体的には、第1、第2の領域AR1、AR2は、幅がWピクセル(例えば1920ピクセル)であり高さがH/2ピクセル(例えば1080/2=540ピクセル)の領域になっている。

【0078】

そして画像の書き込み時には、バッファTBのメモリピッチを2Nバイト(例えば2×4バイト×2048ピクセル=16384バイト)に設定する。ここでメモリピッチ(ピッチサイズ)は、ビットマップラインの先頭を表すアドレスと、次のラインの先頭を表すアドレスとのバイト単位の距離である。描画部120は、このように設定されたバッファTBにおいて、幅がWピクセルであり高さがHピクセルである第1の視点画像を、第1の領域AR1に書き込む。また幅がWピクセルであり高さがHピクセルである第2の視点画像を、第2の領域AR2に書き込む。例えば第1の視点画像を描画バッファDB(DB1又はDB2)に描画し、描画された第1の視点画像を高さ方向において圧縮してバッファTBの第1の領域AR1にコピーする。また第2の視点画像を描画バッファDB(DB1又はDB2)に描画し、描画された第2の視点画像を高さ方向において圧縮してバッファTBの第2の領域AR2にコピーする。この場合のコピー処理は、第1、第2の領域AR1、AR2のサイズのポリゴン(画面に表示されない仮想的なポリゴン)に対して、第1又は第2の視点画像をテクスチャとしてマッピングし、当該ポリゴンを第1又は第2の領域AR1、AR2に描画することで実現できる。

30

40

【0079】

一方、画像の読み出し時には、バッファTBのメモリピッチをNバイトに設定する。描画部120は、このように設定されたバッファTBから画像を読み出すことで、多眼立体視画像を生成する。例えば第1、第2の視点画像が第1、第2の領域AR1、AR2にコピーされた後に、バッファTBの画像を描画バッファDB(DB1又はDB2)に書き戻す(ライトバックする)。この場合の書き戻し処理は、バッファTBの画像を、描画バッ

50

ファDB (DB1又はDB2)のサイズのポリゴンにテクスチャマッピングし、当該ポリゴンを描画バッファDBに描画することなどで実現できる。

【0080】

コマンド実行部118が含む音生成部130は、処理部100で行われる種々の処理の結果に基づいて音処理を行い、BGM、効果音、又は音声などのゲーム音を生成し、音出力部192に出力する。

【0081】

なお、本実施形態の画像生成システムは、1人のプレーヤのみがプレイできるシングルプレーヤモード専用のシステムにしてもよいし、複数のプレーヤがプレイできるマルチプレーヤモードも備えるシステムにしてもよい。また複数のプレーヤがプレイする場合に、これらの複数のプレーヤに提供するゲーム画像やゲーム音を、1つの端末を用いて生成してもよいし、ネットワーク(伝送ライン、通信回線)などで接続された複数の端末(ゲーム機、携帯電話)を用いて分散処理により生成してもよい。

【0082】

2. 本実施形態の手法

2.1 立体視方式

まず、図2、図3を用いて、本実施形態において採用される立体視方式の一例について説明する。

【0083】

図2の画像生成システム300(例えばゲーム装置)により生成された多眼立体視画像は、表示装置310(例えばハイビジョン対応テレビ)の表示画面に表示される。表示画面には、右目用偏光フィルタと左目用偏光フィルタが走査ライン毎に交互に配列されたフィルム312が貼られている。プレーヤは、これらの偏光フィルタに対応する偏光眼鏡320を掛けて表示画面を見ることで、立体視映像を鑑賞する。

【0084】

具体的には表示装置310の表示画面には、走査ライン毎に右目用画像と左目用画像とが交互に表示される。プレーヤは、偏光眼鏡320の右目部分322に設けられた偏光フィルタを介して、フィルム312の右目用偏光フィルタにより偏光された各走査ラインの右目用画像を見る。また偏光眼鏡320の左目部分324を介して、フィルム312の左目用偏光フィルタにより偏光された各走査ラインの左目用画像を見る。

【0085】

図2の立体視方式によれば、プレーヤは、右目用画像及び左目用画像を偏光眼鏡320の右目部分322及び左目部分324を介して同時に見ることができる。従って、右目用画像と左目用画像がフレーム毎に交互に表示される従来の立体視方式に比べて、チラツキが少なく疲れにくい理想的な立体視表示を実現できる。特に3次元ゲーム映像のように、動きの速い映像では、従来の立体視方式では立体感が失われたり、眼の疲労が大きくなるなどの欠点があるが、図2の立体視方式によれば、このような欠点を解消できる。

【0086】

図3に右目用画像IR、左目用画像IL、多眼立体視画像ISを概念的に示す。図3の多眼立体視画像ISでは、奇数番目の走査ラインには右目用画像IRのピクセル画像が表示され、偶数番目の走査ラインには左目用画像ILのピクセル画像が表示される。プレーヤは、奇数番目の走査ラインのピクセル画像を偏光眼鏡320の右目部分322を介して見ると共に、偶数番目の走査ラインのピクセル画像を左目部分324を介して見ることによって、立体感のある映像を楽しむことができる。

【0087】

図4(A)、図4(B)、図5に、本実施形態の画像生成システムにより生成される右目用画像IR、左目用画像IL、多眼立体視画像ISの一例を示す。これらは3次元レースゲームに本実施形態の手法を適用した場合の画像例である。

【0088】

図4(A)、図4(B)では、全ての走査ラインについて右目用画像IR、左目用画像

10

20

30

40

50

I Lのピクセル画像が生成されている。例えばフルハイビジョン対応の場合には、1080本の全ての走査ラインについてピクセル画像が生成される。そして図5の多眼立体視画像I Sは、これらのフルハイビジョン対応の右目用画像I R、左目用画像I Lを合成することで生成される。従って、フルハイビジョンの解像度で立体視ゲーム映像を表示できるため、プレーヤの仮想現実感を飛躍的に向上できる。即ち従来の立体視方式では、立体視画像にすることで、解像度が落ちたり、フレームレートが低下するため、画像品質が低下してしまうという問題があったが、図4(A)~図5の手法によればこのような問題を解消できる。

【0089】

## 2.2 コマンド生成・実行

次に本実施形態のコマンド生成・実行手法について説明する。なお本実施形態のコマンド生成・実行手法を適用できる立体視方式は図2、図3の方式に限定されるものではなく、例えば右目用画像と左目用画像をフレーム毎に交互に表示する立体視方式であっても、本実施形態の手法は適用できる。また以下では、第1の視点画像が右目用画像であり、第2の視点画像が左目用画像である場合を例にとり説明するが、本実施形態はこれに限定されない。例えば第1の視点画像が左目用画像等であり、第2の視点画像が右目用画像等であってもよい。また、以下では、2眼立体視を例にとり説明するが、本実施形態はこれに限定されない。例えばn=3となるn眼立体視にも本実施形態の手法は適用できる。

【0090】

図6に示すように本実施形態では、右目用のコマンドバッファC B R、左目用のコマンドバッファC B L、共通のコマンドバッファC B Mが別個独立に用意されている。即ち、これまでの画像生成システムに用意されていたメインのコマンドバッファC B Mに加えて、コマンドバッファC B R、C B Lが追加されている。

【0091】

コマンド生成部110は、コマンド実行部118に実行させる各処理のうち、右目用、左目用の処理として別個に必要な処理については、右目用のコマンドC R、左目用のコマンドC Lを生成(作成)して、コマンドバッファC B R、C B Lに書き込む。この場合、コマンド生成にあたって、右目用、左目用に共通に必要な処理(例えばモデル単位でのカリング処理、モーション用行列の計算)については、右目用、左目用の共通な処理として1回だけ行うようにする。これにより、これらの処理を重複して行わなくても済むようになるため、処理負荷を大幅に軽減できる。また右目と左目で内容が同じ処理については、同じコマンドを生成して、コマンドC R、C LとしてコマンドバッファC B R、C B Lに書き込む。これにより、コマンドの生成処理や書き込み処理を重複して行わなくても済むようになるため、処理負荷を軽減できる。

【0092】

またコマンド生成部110は、コマンド実行部118に実行させる各処理のうち、右目用、左目用として別個な処理が必要ではない処理については、共通コマンドC Mを生成して、共通のコマンドバッファC B Mに書き込む。例えばハードウェア制御コマンドや影生成処理用コマンドなど、右目や左目の視点情報が無くても実行可能なコマンドについては、共通のコマンドバッファC B Mに書き込む。これにより、重複したコマンドの作成が不要になるため、処理負荷を軽減できる。

【0093】

次に図7を用いて、右目用コマンドC R、左目用コマンドC L、共通コマンドC Mの具体的な生成、書き込み処理について説明する。

【0094】

図7では、まずコマンド生成部110は、シェーダ設定コマンドC S S Tを生成して、コマンドバッファC B R、C B Lの両方に書き込む。なお図7等において、コマンドバッファC B R、C B L、C B MはF I F O構成のバッファになっている。

【0095】

ここでシェーダ設定コマンドC S S Tは、例えばピクセルシェーダや頂点シェーダのプ

10

20

30

40

50



ログラムを特定するためのID情報（アドレス情報）を含む。即ち、ハードウェアであるコマンド実行部118（描画プロセッサ）に対して、右目用画像、左目用画像の生成に使用するシェーダ（シェーダプログラム）を指定するための情報を含む。

**【0096】**

この場合にシェーダの設定処理は、右目と左目（広義には第1、第2の視点）で同じシェーダを使用するため、右目と左目で内容が同じ処理になる。従ってコマンド生成部110は、同じ内容のシェーダ設定コマンドC S S Tを生成して、右目用、左目用のコマンドC R、C LとしてコマンドバッファC B R、C B Lに対して例えば同じタイミングで書き込む。

**【0097】**

次にコマンド生成部110は、カメラ依存パラメータ（右目用パラメータ）の設定コマンドC C D Rを生成して、右目用のコマンドバッファC B Rに書き込む。またカメラ依存パラメータ（左目用パラメータ）の設定コマンドC C D Lを生成して、左目用のコマンドバッファC B Lに書き込む。

**【0098】**

ここでカメラ依存パラメータ設定コマンドC C D R、C C D Lは、右目用、左目用の仮想カメラの視点位置、視線方向等に依存するパラメータを設定するためのコマンドである。具体的には、例えば前述のシェーダ設定コマンドC S S Tで指定されるシェーダに対して、その処理に必要なカメラ依存パラメータを設定するためのコマンドである。

**【0099】**

この場合にカメラ依存パラメータ設定処理は、右目と左目（第1、第2の視点）でパラメータの値が異なるため、右目と左目で内容が異なる処理になる。従ってコマンド生成部110は、異なる内容のカメラ依存パラメータの設定コマンドC C D R、C C D Lを生成して、右目用、左目用のコマンドC R、C LとしてコマンドバッファC B R、C B Lに書き込む。

**【0100】**

次にコマンド生成部110は、カメラ非依存パラメータの設定コマンドC C N Dを生成して、右目用、左目用のコマンドバッファC B R、C B Lの両方に書き込む。

**【0101】**

ここでカメラ非依存パラメータ設定コマンドC C N Dは、右目用、左目用の仮想カメラの視点位置、視線方向等に依存しないパラメータを、シェーダ設定コマンドC S S Tで指定されるシェーダに対して設定するコマンドである。

**【0102】**

この場合にカメラ非依存パラメータ設定処理は、右目と左目でパラメータの値が同じになるため、右目と左目で内容が同じ処理になる。従ってコマンド生成部110は、同じ内容のカメラ依存パラメータ設定コマンドC C N Dを生成して、右目用、左目用のコマンドC R、C LとしてコマンドバッファC B R、C B Lに例えば同じタイミングで書き込む。

**【0103】**

次にコマンド生成部110は、シェーダ実行コマンドC S E Xを生成して、右目用、左目用のコマンドバッファC B R、C B Lの両方に書き込む。

**【0104】**

ここでシェーダ実行コマンドC S E Xは、シェーダ設定コマンドC S S Tで指定されるシェーダに対して、シェーダ処理の実行を指示するコマンドである。

**【0105】**

この場合にシェーダの実行を指示する処理は、右目と左目で内容が同じ処理になる。従ってコマンド生成部110は、同じ内容のシェーダ実行コマンドC S E Xを生成して、右目用、左目用のコマンドC R、C LとしてコマンドバッファC B R、C B Lに例えば同じタイミングで書き込む。

**【0106】**

次にコマンド生成部110は、右目用画像と左目用画像の合成コマンドC S Y Nを生成

10

20

30

40

50

して、共通のコマンドバッファ C B M に書き込む。

【 0 1 0 7 】

ここで合成コマンド C S Y N は、右目用画像と左目用画像を合成するためのコマンドであり、例えば後述するように、バッファ T B から描画バッファ D B ( D B 1 又は D B 2 ) への画像の書き戻しコマンドなどである。

【 0 1 0 8 】

次にコマンド生成部 1 1 0 は、バックバッファとフロントバッファの切り替えコマンド C S W を生成して、共通のコマンドバッファ C B M に書き込む。

【 0 1 0 9 】

ここで切り替えコマンド C S W は、例えば第 1、第 2 の描画バッファ D B 1、D B 2 の一方のバッファをバックバッファからフロントバッファに切り替え、他方のバッファをフロントバッファからバックバッファに切り替えるためのコマンドである。この切り替え処理は、右目用画像、左目用画像の生成には無関係であり、右目用、左目用コマンドは不要な処理であるため、切り替えコマンド C S W は、共通のコマンドバッファ C B M に書き込まれる。

【 0 1 1 0 】

例えば本実施形態の比較例の手法として、右目用、左目用のコマンドバッファ C B R、C B L を用意せずに、共通(メイン)のコマンドバッファ C B M のみを用いる手法が考えられる。この手法では、まず右目用画像を生成するためのコマンドを生成して、コマンドバッファ C B M に書き込む。次に、左目用画像を生成するためのコマンドを生成して、コマンドバッファ C B M に書き込む。その後、合成コマンドや切り替えコマンドを生成して、コマンドバッファ C B M に書き込む。

【 0 1 1 1 】

しかしながら、この比較例の手法では、例えばカメラ非依存パラメータ設定コマンド等の生成や書き込みが、異なるタイミングで行われる。従って、パラメータの生成処理(例えばモーション用行列のパラメータの生成処理)が、本来は 1 回だけの処理で済むのに、重複して行われてしまい、処理に無駄が生じる。このため、例えば 1 フレーム期間内に行う必要がある他の処理を圧迫してしまい、処理が間に合わなくなってしまうという問題が生じる。特に、立体視画像の生成では、右目用画像の描画処理と左目用画像の描画処理を 1 フレーム期間内に行う必要があり、処理時間に余裕がないため、この問題は深刻となる。

【 0 1 1 2 】

この点、図 6、図 7 の本実施形態の手法によれば、右目用、左目用のコマンド C R、C L の生成処理や書き込み処理が、同じタイミングで行われる。従って、パラメータ(例えばモーション用行列)の生成処理等が重複して行われてしまう事態を防止でき、処理を大幅に効率化できる。従って、1 フレーム期間内に行う必要がある処理が間に合わなくなってしまうという問題を解消でき、立体視画像の生成に最適なコマンド生成・書き込み手法を提供できる。

【 0 1 1 3 】

2.3 共通コマンド、カメラ依存パラメータ、カメラ非依存パラメータ

次に図 7 の共通コマンド、カメラ依存パラメータ、カメラ非依存パラメータの具体例について図 8 ( A ) ~ 図 8 ( C ) を用いて説明する。

【 0 1 1 4 】

図 8 ( A ) は共通コマンドの例である。影生成処理用コマンドは、例えば影生成処理がハードウェアによりサポートされている場合に、影生成処理用のシェーダやパラメータを設定するためのコマンドである。

【 0 1 1 5 】

このような影生成処理としては、例えばシャドウマップによる影生成処理がある。例えばシャドウマップによる影生成処理では、影生成のシェーダ(シェーダプログラム)の設定コマンドや、影生成のための光源位置や影の色などの影生成処理用のパラメータの設定

10

20

30

40

50

コマンドが、影生成処理用コマンドになる。そして、このような影生成処理用コマンドは、右目、左目の視点位置に無関係であるため、共通コマンドCMとして共通コマンドバッファCBMに書き込まれる。

【0116】

図8(A)のハードウェア制御コマンドは、ハードウェアの制御のために共通に使用されるコマンドである。このようなハードウェア制御コマンドとしては、例えばシェーディング方法、カリング(ハードウェア・カリング)方法、ブレンディング、デプステスト、ステンシルテスト等の設定コマンドや、バックバッファとフロントバッファの切り替えコマンドや、描画バッファのクリアコマンドなどがある。

【0117】

例えばシェーディング方法の設定コマンドでは、グローシェーディングやフラットシェーディングの指定を行う。カリング方法の設定コマンドでは、ハードウェア(描画プロセッサ)によるカリング(クリッピング)のオン・オフの指定や裏面・表面のカリングの指定などを行う。ブレンディングの設定コマンドでは、ブレンディングのオン・オフの指定やブレンディングの方法(式)の指定などを行う。デプステストやステンシルテストの設定コマンドでは、デプステスト、ステンシルテストのオン・オフの指定やその設定を行う。バックバッファとフロントバッファの切り替えコマンドは、図7で説明した切り替えコマンドCSWである。描画バッファDBのクリアコマンドは、描画バッファDBに画像を描画する前などに、描画バッファDBの画像をクリアするためのコマンドである。

【0118】

これらのハードウェア制御コマンドは、右目、左目の視点位置に無関係に、共通の制御コマンドとして使用されるものであるため、共通コマンドCMとして共通コマンドバッファCBMに書き込まれる。

【0119】

図8(B)のカメラ依存パラメータとしては、ワールドビュー行列のパラメータ、ワールドビュープロジェクション行列のパラメータ、或いはカメラの位置ベクトルなどを想定できる。

【0120】

ここでワールドビュー行列は、ワールド座標系からカメラ座標系(視点座標系)への変換行列であり、ワールドビュープロジェクション行列はワールド座標系から射影座標系への変換行列である。右目用の仮想カメラと左目用の仮想カメラとでは、その視点位置や視線方向が異なるため、ワールドビュー行列やワールドビュープロジェクション行列のパラメータ(行列要素)も異なる。従って、これらの行列のパラメータを設定するコマンドは、カメラ依存パラメータ設定コマンドとして、右目用と左目用で異なる内容のコマンドが生成されてコマンドバッファCBR、CBLに書き込まれる。

【0121】

カメラの位置ベクトルは、例えばオブジェクトのサーフェスの鏡面反射処理などを行う場合に使用される。即ちフォンの照明モデル等では、反射ベクトルは、オブジェクトの法線ベクトルとカメラの位置ベクトルにより決定され、この反射ベクトルと光源ベクトルと鏡面反射指数(ハイライト係数)により、スペキュラ光の輝度が求められる。従って、このような処理を行うためにはカメラの位置ベクトルが必要であり、この位置ベクトルは右目用の仮想カメラと左目用の仮想カメラで異なる。従って、このカメラの位置ベクトルの設定コマンドは、カメラ依存パラメータ設定コマンドとして、右目用と左目用で異なる内容のコマンドが生成されてコマンドバッファCBR、CBLに書き込まれる。

【0122】

図8(C)のカメラ非依存パラメータとしては、ワールド行列のパラメータ、モーション用行列のパラメータ、ライトの位置ベクトル、環境色、スペキュラ色、フォグ色、頂点データ、或いはテクスチャデータなどを想定できる。

【0123】

10

20

30

40

50

ここでワールド行列はローカル座標系（モデル座標系）からワールド座標系（絶対座標系）への変換行列である。このワールド行列は、右目用、左目用の仮想カメラの視点位置や視線方向に無関係である。従って、ワールド行列のパラメータの設定コマンドは、カメラ非依存パラメータ設定コマンドとして、右目用と左目用で同じ内容のコマンドが生成されてコマンドバッファC B R、C B Lに書き込まれる。

#### 【 0 1 2 4 】

モーション用行列のパラメータは、例えば、モデルオブジェクトのスケルトンを構成する骨の位置又は回転角度（親の骨に対する子の骨の相対的な方向）等を含むモーションデータに基づいて各フレーム毎に計算される。このモーション用行列のパラメータは、右目、左目の視点に依存しないパラメータであり、このパラメータの生成処理が重複して行われると、処理が無駄になる。この点、本実施形態の手法によれば、モーション用行列のパラメータの計算は、各フレーム毎に1回だけ実行され、それにより得られたパラメータの設定コマンドが、カメラ非依存パラメータ設定コマンドとして、右目用と左目用で同じ内容のコマンドが生成されてコマンドバッファC B R、C B Lに書き込まれる。従って、処理負荷を大幅に軽減できる。

10

#### 【 0 1 2 5 】

ライトの位置ベクトル、環境色、スペキュラ色は、照明モデルを用いたライティング処理に用いられるパラメータである。またフォグ色はフォグの生成処理に用いられるパラメータである。これらのパラメータは、右目用、左目用の仮想カメラの視点位置や視線方向に無関係であるため、カメラ非依存パラメータ設定コマンドとして、右目用と左目用で同じ内容のコマンドが生成されてコマンドバッファC B R、C B Lに書き込まれる。

20

#### 【 0 1 2 6 】

頂点データは、例えば法線、接線、頂点色、テクスチャ座標、頂点座標などである。テクスチャデータは、デカルテクスチャ、法線マップなどである。この頂点データ、テクスチャデータは、右目、左目の視点位置等に無関係であるため、カメラ非依存パラメータ設定コマンドとして、右目用と左目用で同じ内容のコマンドが生成されてコマンドバッファC B R、C B Lに書き込まれる。

#### 【 0 1 2 7 】

### 2 . 4 コマンドの実行

次に図9（A）、図9（B）、図10を用いて、コマンド実行部118の動作について説明する。

30

#### 【 0 1 2 8 】

図9（A）に示すようにコマンド実行部118は、右目用のコマンドバッファC B Rに書き込まれた右目用のコマンドC S S T、C C D R、C C N D、C S E X等を、順次読み出して実行する。即ち図7等の手法でコマンド生成部110が生成してコマンドバッファC B Rに書き込んだコマンドを実行する。具体的には、コマンドC S S Tにより、使用するシェーダを設定し、コマンドC C D Rにより右目用のカメラ依存パラメータをシェーダに設定し、コマンドC C N Dによりカメラ非依存パラメータをシェーダに設定する。そしてコマンドC S E Xによりシェーダの実行を開始する。このようなコマンドC S S T、C C D R、C C N D、C S E Xの実行により、図4（A）に例示されるような右目用画像I Rが生成される。

40

#### 【 0 1 2 9 】

次に図9（B）に示すようにコマンド実行部118は、左目用のコマンドバッファC B Lに書き込まれた左目用のコマンドC S S T、C C D L、C C N D、C S E X等を、順次読み出して実行する。即ち図7等の手法でコマンド生成部110が生成してコマンドバッファC B Lに書き込んだコマンドを実行する。具体的には、コマンドC S S Tにより、使用するシェーダを設定し、コマンドC C D Lにより左目用のカメラ依存パラメータをシェーダに設定し、コマンドC C N Dによりカメラ非依存パラメータをシェーダに設定する。そしてコマンドC S E Xによりシェーダの実行を開始する。このようなコマンドC S S T、C C D L、C C N D、C S E Xの実行により、図4（B）に例示されるような左目用画

50

像 I L が生成される。

【 0 1 3 0 】

このような右目用コマンド、左目用コマンドの実行後に、図 1 0 に示すようにコマンド実行部 1 1 8 は、共通のコマンドバッファ C B M に書き込まれた共通コマンド C S Y N、C S W 等を実行する。即ち図 7 等の手法でコマンド生成部 1 1 0 が生成してコマンドバッファ C B M に書き込んだコマンドを実行する。具体的には、図 9 ( A )、図 9 ( B ) で生成された右目用画像 I R、左目用画像 I L を、コマンド C S Y N により合成して、多眼立体視画像 I S を生成して、例えば第 1 の描画バッファ D B 1 ( 第 1、第 2 の描画バッファの一方のバッファ ) に書き込む。そしてコマンド C S W により、第 1 の描画バッファ D B 1 をバックバッファからフロントバッファに切り替える。また第 2 の描画バッファ D B 2 ( 第 1、第 2 の描画バッファの他方のバッファ ) をフロントバッファからバックバッファに切り替える。これにより、表示装置の画面上に、図 5 に例示されるような多眼立体視画像 I S が表示されるようになる。

10

【 0 1 3 1 】

なおコマンド実行部 1 1 8 は、右目用、左目用コマンドの実行前に、共通のコマンドバッファ C B M に書き込まれた共通コマンドを実行するようにしてもよい。この場合に実行される共通コマンドとしては、図 8 ( A ) に例示したようなハードウェア制御コマンドや影生成処理用コマンドなどを想定できる。

【 0 1 3 2 】

図 9 ( A ) ~ 図 1 0 に示すようなシーケンスでコマンドを実行すれば、コマンド生成部 1 1 0 により同じタイミング ( ほぼ同じタイミング ) で書き込まれた右目用、左目用のコマンドを、別のタイミングで実行して、右目用画像 I R、左目用画像 I L を生成することが可能になる。また右目用、左目用コマンドの実行後に、必要なハードウェア制御等を適正に実行できるようになる。

20

【 0 1 3 3 】

2 . 5 ジャンプ処理

本実施形態では図 9 ( A ) ~ 図 1 0 に示すようなシーケンスのコマンド実行を実現するために、ジャンプコマンドを用意して、コマンドバッファ C B R、C B L 等に記載している。

【 0 1 3 4 】

例えば図 1 1 に示すようにコマンド生成部 1 1 0 は、右目用 ( 第 1 の視点用 ) のコマンドバッファ C B R から左目用 ( 第 2 の視点用 ) のコマンドバッファ C B L に対して、コマンド実行部 1 1 8 の処理をジャンプさせるためのコマンド J P R L を、コマンドバッファ C B R に書き込む。具体的には、右目用画像の生成に必要なコマンド C S S T、C C D R、C C N D、C S E X 等を書き込んだ後、最後にジャンプコマンド J P R L をコマンドバッファ C B R に書き込む。

30

【 0 1 3 5 】

またコマンド生成部 1 1 0 は、左目用のコマンドバッファ C B L から共通のコマンドバッファ C B M に対して、コマンド実行部 1 1 8 の処理をジャンプさせるためのコマンド J P L M を、コマンドバッファ C B L に書き込む。具体的には、左目用画像の生成に必要なコマンド C S S T、C C D L、C C N D、C S E X 等を書き込んだ後、最後にジャンプコマンド J P L M をコマンドバッファ C B L に書き込む。

40

【 0 1 3 6 】

図 1 2 のようにすれば、コマンド実行部 1 1 8 は、右目用のコマンド C S S T、C C D R、C C N D、C S E X 等をコマンドバッファ C B R から読み出して実行した後に、ジャンプコマンド J P R L を実行することで、コマンドバッファ C B L の例えば先頭アドレスにジャンプする。そしてコマンドバッファ C B L のコマンドの読み出しを開始する。

【 0 1 3 7 】

次にコマンド実行部 1 1 8 は、左目用のコマンド C S S T、C C D L、C C N D、C S E X 等をコマンドバッファ C B L から読み出して実行した後に、ジャンプコマンド J P L

50

Mを実行することで、コマンドバッファCBMの例えば先頭アドレスにジャンプする。そして、コマンドバッファCBMのコマンドの読み出しを開始する。次に、コマンド実行部118は、コマンドバッファCBMのコマンドCSYN、CSWを実行する。

【0138】

図11の本実施形態のジャンプ手法によれば、コマンド生成部110がジャンプコマンドJPR L、JPLMを生成してコマンドバッファCBR、CBLに書き込むだけで、図9(A)~図10で説明したようなシーケンスのコマンド実行を実現できるため、処理の簡素化、効率化を図れる。

【0139】

なお図12に示すようにコマンド生成部110は、共通のコマンドバッファCBMから右目用のコマンドバッファCBRに対して、コマンド実行部118の処理をジャンプさせるためのコマンドJPMRを、共通コマンドバッファCBMに書き込むようにしてもよい。具体的には、右目用画像、左目画像の生成前に実行する必要がある共通コマンドCHCN、CSHAをコマンドバッファCBMに書き込んだ後に、ジャンプコマンドJPMRをコマンドバッファCBMに書き込む。

【0140】

このようにすれば、コマンド実行部118は、共通コマンドCHCN、CSHAをコマンドバッファCBMから読み出して実行した後に、ジャンプコマンドJPMRを実行することで、コマンドバッファCBRの例えば先頭アドレスにジャンプする。そしてコマンドバッファCBRのコマンドの読み出しを開始して、コマンドCSST、CCDR、CCND、CSEX等を実行する。

【0141】

図12の手法によれば、ジャンプコマンドJPMRを生成してコマンドバッファCBMに書き込むだけで、右目用画像、左目用画像の生成前に、ハードウェア制御コマンドなどのコマンドを実行できるようになり、効率的なコマンド実行のシーケンス制御を簡素な処理で実現できる。

【0142】

## 2.6 カリング処理

本実施形態では、多眼立体視の右目(第1の視点)と左目(第2の視点)とで共通のカリング処理を行うようにしている。このカリング処理は、いわゆるハードウェアによる裏面ポリゴン等のカリング処理ではなく、その前段階に行うソフトウェアによるオブジェクト(モデルオブジェクト)のカリング処理(削除処理、クリッピング処理)である。

【0143】

例えば図13において、右目用仮想カメラVCRのビューボリュームVVR内に存在しないオブジェクトは、右目用画像の生成に不要なオブジェクトである。また左目用仮想カメラVCLのビューボリュームVVL内に存在しないオブジェクトは、左目用画像の生成に不要なオブジェクトである。

【0144】

この場合に比較例の手法として、例えば右目用画像の生成処理の際に、右目用のビューボリュームVVR内に存在し得ないオブジェクトをオブジェクトデータのオブジェクトリストから削除すると共に、左目用画像の生成処理の際に、左目用のビューボリュームVVL内に存在し得ないオブジェクトをオブジェクトリストから削除する手法も考えられる。

【0145】

しかしながら、この比較例の手法では、ソフトウェアによるオブジェクトのカリング処理を、2回行う必要があり、処理に無駄があった。特に、オブジェクト空間内に配置されるオブジェクトの数が多い場合には、この2回のカリング処理の負荷は非常に重い。従って、比較例の手法を採用すると、1フレームに行う必要がある他の処理を圧迫してしまう。

【0146】

この点、図13のカリング手法では、多眼立体視の右目と左目とで共通のカリング処理

を行っている。具体的には、右目用、左目用に共通のカリング処理用ボリューム A R C L (非カリング範囲)を設定する。このカリング処理用ボリューム A R C Lは、右目用のビューボリューム V V R 及び左目用のビューボリューム V V L を内包するものである。そして、共通のカリング処理の際には、このカリング処理用ボリューム A R C L 内に存在し得ないオブジェクトを削除する。

【 0 1 4 7 】

そしてコマンド生成部 1 1 0 は、このような共通のカリング処理を行うことで取得される頂点データの設定コマンドを、同じ内容の右目用、左目用コマンドとして生成して、コマンドバッファ C B R、C B L に書き込む。具体的には、カリング処理用ボリューム A R C L 内に存在し得ないオブジェクトを削除したオブジェクトリストに基づいて、頂点データを生成する。そして、生成された頂点データを設定するためのコマンドであるカメラ非依存パラメータ設定コマンド C C N D を、図 7 に示すようにコマンドバッファ C B R、C B L に書き込む。

10

【 0 1 4 8 】

このようにすれば、図 1 3 のカリング処理を、右目用、左目用に重複して行う必要がなくなり、1回で済むようになるため、処理負荷を軽減できる。特に、オブジェクト空間に設定されるオブジェクトの数が多い場合には、比較例の手法に比べて大幅に処理負荷を軽減できる。

【 0 1 4 9 】

なお図 1 1 のカリング処理用ボリューム A R C L として、例えば右目用、左目用ではない元の仮想カメラのカリング処理用ボリュームを使用してもよい。例えば立体視ではない既存のゲームプログラムが存在する場合には、そのゲームプログラムで使用される元の仮想カメラの視点位置や視線方向等のカメラ情報に基づいて、右目用、左目用の仮想カメラ V C R、V C L のカメラ情報を設定する。このようにすれば、これまでのゲームプログラムの資産を有効活用して、立体視のゲームを開発できるため、開発期間を短縮できる。そして、この場合に元のゲームプログラムで使用されていた仮想カメラのカリング処理用ボリュームが、十分に広い範囲をカバーしており、右目用、左目用のビューボリューム V V R、V V L を内包するものである場合には、その元のカリング処理用ボリュームを、右目用、左目用の共通のカリング処理用ボリューム A R C L として使用すればよい。

20

【 0 1 5 0 】

なお図 1 3 のように共通のカリング処理を行う手法では、図 6 ~ 図 1 2 等で説明したコマンドバッファ C B R、C B L、C B M を使用せずにコマンドの生成、実行処理を行うようにしてもよい。

30

【 0 1 5 1 】

## 2.7 立体視画像の生成手法

次に、本実施形態の立体視画像の生成手法の一例について説明する。但し、本実施形態の立体視画像の生成手法は以下に説明する手法に限定されるものではない。

【 0 1 5 2 】

図 1 4 では、バッファ T B に第 1、第 2 の領域 A R 1、A R 2 を確保している。そして図 4 ( A ) に例示されるような右目用画像 I R (広義には第 1 の視点画像)を生成し、この右目用画像 I R を高さ方向(縦方向)において圧縮(縮小)して、バッファ T B の第 1 の領域 A R 1 に書き込む。また図 4 ( B ) に例示されるような左目用画像 I L (広義には第 2 の視点画像)を生成し、この左目用画像 I L を高さ方向において圧縮して、バッファ T B の第 2 の領域 A R 2 に書き込む。

40

【 0 1 5 3 】

そして、このバッファ T B から画像を読み出すことで、図 5 に例示されるような多眼立体視画像 I S を生成(合成)する。即ち領域 A R 1 の右目用のピクセル画像と領域 A R 2 の左目用のピクセル画像とが、ライン毎(走査ライン毎、ビットマップライン毎)に交互に表示される多眼立体視画像 I S を生成する。

【 0 1 5 4 】

50

具体的には右目用画像 I R、左目用画像 I L は、幅が W ピクセル、高さが H ピクセルの画像になっている。フルハイビジョン対応の場合には例えば W = 1920、H = 1080 になる。一方、領域 A R 1、A R 2 は、幅が W ピクセル、高さが H / 2 ピクセルの領域になっている。

【 0 1 5 5 】

バッファ T B への画像の書き込み時においては、バッファ T B のメモリピッチ（ラインの先頭アドレスから次のラインの先頭アドレスまでのバイト単位の距離）は 2 N バイトに設定される。例えば 1 ピクセル（R、G、B、 ）が 4 バイトである場合には、フルハイビジョン対応では例えば 2 N バイト = 2 × 4 × 2048 バイトになる。そして図 14 では、幅が W ピクセル、高さが H ピクセルである右目用画像 I R が、高さ方向において 1 / 2 に圧縮されて、領域 A R 1 に書き込まれ、幅が W ピクセルであり高さが H ピクセルである左目用画像 I L が、高さ方向において 1 / 2 に圧縮されて、領域 A R 2 に書き込まれる。

10

【 0 1 5 6 】

一方、バッファ T B からの画像の読み出し時においては、バッファ T B のメモリピッチは N バイトに設定される。そしてこのように設定されたバッファ T B から右目用画像 I R、左目用画像 I L を読み出すことで、図 5 に例示されるような多眼立体視画像 I S が合成される。

【 0 1 5 7 】

具体的には、書き込み時に 2 N バイトに設定されたメモリピッチ（ピッチサイズ）を、読み出し時に N バイトに設定することで、領域 A R 1 の第 1 のラインの右目用ピクセル画像が、多眼立体視画像 I S の第 1 のラインに表示され、領域 A R 2 の第 1 のラインの左目用ピクセル画像が、多眼立体視画像 I S の第 2 のラインに表示されるようになる。また領域 A R 1 の第 2 のラインの右目用ピクセル画像が、多眼立体視画像 I S の第 3 のラインに表示され、領域 A R 2 の第 2 のラインの左目用ピクセル画像が、多眼立体視画像 I S の第 4 のラインに表示されるようになる。即ち右目用ピクセル画像と左目用ピクセル画像がライン毎に交互に表示されるようになる。

20

【 0 1 5 8 】

2 . 8 描画バッファの利用

レンダリングにより生成された右目用画像 I R、左目用画像 I L は、直接にバッファ T B の領域 A R 1、A R 2 に描画してもよいが、図 15（A）、図 15（B）に示すように、描画バッファ D B に描画した後に、領域 A R 1、A R 2 にコピーするようにしてもよい。

30

【 0 1 5 9 】

即ち図 15（A）では、右目用画像 I R を描画バッファ D B に描画し、描画された右目用画像 I R を高さ方向において圧縮して、テンポラリバッファ（ワークバッファ）として機能するバッファ T B の領域 A R 1 にコピーする。また図 15（B）では、左目用画像 I L を描画バッファ D B に描画し、描画された左目用画像 I L を高さ方向において圧縮して、バッファ T B の領域 A R 2 にコピーする。

【 0 1 6 0 】

このようにすれば、高さ方向での画像の圧縮（平均化、間引き、縮小）は、描画バッファ D B から領域 A R 1、A R 2 へのコピー処理の際に行えばよく、画像圧縮と右目用・左目用画像の生成を同時に行わなくても済むため、処理の簡素化、効率化を図れる。

40

【 0 1 6 1 】

この場合に図 15（A）、図 15（B）のコピー処理は、例えばテクスチャマッピングを利用して実現できる。

【 0 1 6 2 】

具体的には図 16（A）に示すように、描画バッファ D B に描画された右目用画像 I R を、領域 A R 1 のサイズのポリゴン P L（画面に表示されない仮想的なプリミティブ面）にテクスチャマッピングする。この場合に、テクスチャマッピングは、ポイントサンプリング方式ではなく例えばテクセル補間方式（狭義にはバイリニアフィルタモード）に設定

50



する。そしてこのように右目用画像 I R がテクスチャとしてマッピングされたポリゴン P L を領域 A R 1 に描画することで、図 1 5 ( A ) のコピー処理を実現する。

【 0 1 6 3 】

また図 1 6 ( B ) に示すように、描画バッファ D B に描画された左目用画像 I L を、領域 A R 2 のサイズのポリゴン P L にテクスチャマッピングする。この場合に、テクスチャマッピングはテクセル補間方式に設定する。また高さ方向にテクスチャ座標 ( U V 空間の例えば V 座標 ) を例えば 1 テクセル分だけシフトしてテクスチャマッピングを行う。そしてこのように左目用画像 I L がテクスチャとしてマッピングされたポリゴン P L を領域 A R 2 に描画することで、図 1 5 ( B ) のコピー処理を実現する。

【 0 1 6 4 】

図 1 6 ( A )、図 1 6 ( B ) のようにテクスチャマッピングによりコピー処理を実現すれば、ポリゴン P L のサイズの設定だけで高さ方向の画像圧縮を実現できるため、処理の簡素化、効率化を図れる。また、後述するような隣り合うラインのピクセル画像の平均化処理を、テクスチャマッピングのテクセル補間方式 ( バイリニアフィルタ ) を利用して実現できるため、更に処理を効率化できる。

【 0 1 6 5 】

なお図 1 5 ( A )、図 1 5 ( B ) のように右目用画像 I R、左目用画像 I L が領域 A R 1、A R 2 にコピーされた後に、バッファ T B の画像は描画バッファ D B に書き戻される。具体的にはバッファ T B のメモリピッチを 2 N バイトから N バイトに変更して、バッファ T B の画像を描画バッファ D B に書き戻し、図 5 に例示されるような多眼立体視画像 I S を生成する。

【 0 1 6 6 】

このようにすれば、描画バッファ D B を利用した既存の描画アルゴリズムに対して、テンポラリバッファであるバッファ T B やコピー・書き戻しアルゴリズム等を追加するだけで、右目用画像 I R、左目用画像 I L のテンポラリバッファ T B へのコピー処理を実現して、多眼立体視画像 I S を生成できるようになる。

【 0 1 6 7 】

なお描画バッファ D B への画像の書き戻し ( ライトバック ) は、図 1 6 ( A )、図 1 6 ( B ) と同様にテクスチャマッピングを利用した手法により実現できる。具体的には、バッファ T B のピッチサイズを 2 N バイトから N バイトに変更する。そしてバッファ T B の画像 ( T B の全体の画像 ) を、描画バッファ D B ( D B 1 又は D B 2 ) のサイズのポリゴンにテクスチャマッピングする。そして、バッファ T B の画像がテクスチャマッピングされたポリゴンを描画バッファ D B に描画することで、バッファ T B の画像を描画バッファ D B に書き戻す。

【 0 1 6 8 】

## 2.9 ダブルバッファ方式

描画バッファ D B としてダブルバッファ方式 ( 或いはトリプルバッファ等の複数バッファ方式 ) を採用する場合には、図 1 7 ( A )、図 1 7 ( B ) に示す手法を採用することが望ましい。

【 0 1 6 9 】

例えば図 1 7 ( A ) では、描画バッファ D B として、第 1、第 2 の描画バッファ D B 1、D B 2 が用意 ( 確保 ) されている。また図 1 7 ( A ) の期間では、第 1 の描画バッファ D B 1 がバックバッファ ( 描画用バッファ ) に設定され、第 2 の描画バッファ D B 2 がフロントバッファ ( 表示用バッファ ) に設定されている。なおトリプルバッファ等の複数バッファ方式では、第 1、第 2 の描画バッファ D B 1、D B 2 を含む 3 個以上の描画バッファを用意 ( 確保 ) すればよい。

【 0 1 7 0 】

そして図 1 7 ( A ) では、描画バッファ D B 1 ( 広義には、第 1、第 2 の描画バッファの一方の描画バッファ ) からバッファ T B に対して、図 1 5 ( A )、図 1 5 ( B ) で説明した画像のコピー処理が行われる。またバッファ T B から描画バッファ D B 1 への画像の

10

20

30

40

50

書き戻し処理が行われる。これにより、描画バッファDB1には、次のフレームに表示すべき多眼立体視画像ISが生成される。

【0171】

このように多眼立体視画像ISが描画バッファDB1に生成された後に、図17(B)に示すように、描画バッファDB1をバックバッファからフロントバッファに切り替える(フリップする)。これにより、描画バッファDB1の多眼立体視画像ISが表示装置の画面に表示される。またこの時に、描画バッファDB2(広義には、一方とは異なる他方の描画バッファ)は、フロントバッファからバックバッファに切り替えられる。

【0172】

そして図17(B)では、描画バッファDB2からバッファTBに対して、図15(A)、図15(B)で説明した画像のコピー処理が行われる。またバッファTBから描画バッファDB2への画像の書き戻し処理が行われる。これにより、描画バッファDB2には、次のフレームに表示すべき多眼立体視画像ISが生成される。

【0173】

以上の図17(A)、図17(B)の手法によれば、ダブルバッファ方式の既存の描画アルゴリズムに対して、テンポラリバッファであるバッファTBやコピー・書き戻しアルゴリズム等を追加するだけで、ダブルバッファ方式と本実施形態による多眼立体視画像ISの生成手法を両立できるようになる。

【0174】

なお図14に示すように高さ方向(走査ラインの並ぶ方向)での圧縮処理を行う場合には、隣り合うラインのピクセル画像の平均化処理(補間処理)を行うことが望ましい。

【0175】

具体的には、右目用画像IR(第1の視点画像)を高さ方向において圧縮して領域AR1に書き込む際に、右目用画像の第i(iは自然数)のラインのピクセル画像と次の第i+1のラインのピクセル画像の平均化処理を行う。同様に、左目用画像IL(第2の視点画像)を高さ方向において圧縮して領域AR2に書き込む際に、左目用画像の第iのラインのピクセル画像と次の第i+1のラインのピクセル画像の平均化処理を行う。

【0176】

例えば、右目用画像IRの第1、第2、第3、第4のラインのピクセル画像(ライン中の特定ピクセルのRGB画像)をR1、R2、R3、R4とする。この場合には、領域AR1の第1のラインのピクセル画像は、R1とR2の平均化処理により $(R1 + R2) / 2$ になり、第2のラインのピクセル画像は、R3とR4の平均化処理により $(R3 + R4) / 2$ になる。右目用画像IRの他のラインや、左目用画像ILについても同様である。

【0177】

このような平均化処理を行えば、多眼立体視画像において右目用ピクセル画像と左目用ピクセル画像がライン毎に交互に表示される場合にも、例えばハイビジョン等に対応した高精細な解像度を維持した立体視画像の表示が可能になる。また、ジャギーやエイリアシングの発生を低減でき、高品位な立体視画像を生成できる。

【0178】

なおこのような平均化処理は、図16(A)、図16(B)で説明したように、テクスチャのマッピングモードをテクセル補間方式(バイリニアフィルタモード)に設定して、高さ方向にテクスチャを圧縮したテクスチャマッピングを行うことで、自動的に実行できる。具体的には、右目用画像IR、左目用画像ILを、領域AR1、AR2のサイズのポリゴンにテクセル補間方式でテクスチャマッピングして領域AR1、AR2に描画することで、第iのラインのピクセル画像と第i+1のラインのピクセル画像の平均化処理を実現する。

【0179】

なお平均化処理を行う場合には、多眼立体視画像ISの第m(mは自然数)のラインについては、右目用画像IR(第1の視点画像)の第j(jは自然数)のラインのピクセル画像と次の第j+1のラインのピクセル画像の平均化を行うことが望ましい。また多眼立

10

20

30

40

50

体視画像 I S の第  $m + 1$  のラインについては、左目用画像 I L (第 2 の視点画像) の第  $j + 1$  のラインのピクセル画像と次の第  $j + 2$  のラインのピクセル画像の平均化を行うことが望ましい。これは図 16 (B) で説明したテクスチャ座標の 1 テクセル分のシフトにより実現できる。

【0180】

このような手法を採用すれば、多眼立体視画像 I S において適正な位置に適正な右目用ピクセル画像又は左目用ピクセル画像が表示されるようになるため、ジャギーやアンチエイリアシングを効果的に低減できる。

【0181】

2.10 詳細な処理例

10

次に本実施形態の詳細な処理例を図 18 ~ 図 20 のフローチャートを用いて説明する。

【0182】

まず、右目用 (第 1 の視点用) のコマンドバッファ C B R、左目用 (第 2 の視点用) のコマンドバッファ C B L、共通のコマンドバッファ C B M を記憶部 170 に確保する (ステップ S 21)。

【0183】

次に、各種設定のためのハードウェア制御コマンド (図 8 (A) 参照) を生成して、コマンドバッファ C B M に書き込む (ステップ S 22)。また、必要であれば、影生成処理用コマンドを生成して、コマンドバッファ C B M に書き込む (ステップ S 23)。そして図 12 で説明したように、コマンドバッファ C B R へのジャンプコマンド J P M R を生成して、コマンドバッファ C B M に書き込む (ステップ S 24)。

20

【0184】

次に、背景の描画処理を行う (ステップ S 25)。具体的には、図 7 で説明したように背景を描画するためのシェーダの設定コマンド C S S T を生成して、コマンドバッファ C B R、C B L に書き込む (ステップ S 26)。次に、図 8 (B) に示すようなカメラ依存 (右目用仮想カメラ) の背景用パラメータをシェーダに設定するコマンド C C D R を生成して、コマンドバッファ C B R に書き込む (ステップ S 27)。また、カメラ依存 (左目用仮想カメラ) の背景用パラメータをシェーダに設定するコマンド C C D L を生成して、コマンドバッファ C B L に書き込む (ステップ S 28)。

【0185】

30

次に、図 8 (C) に示すようなカメラ非依存の背景用パラメータをシェーダに設定するコマンド C C N D を生成して、コマンドバッファ C B R、C B L に書き込む (ステップ S 29)。最後に、シェーダを実行するためのコマンド C S E X を生成して、コマンドバッファ C B R、C B L に書き込む (ステップ S 30)。

【0186】

次に、車 (モデルオブジェクト) の描画処理を行う (ステップ S 31)。具体的には、車を描画するためのシェーダの設定コマンド C S S T を生成して、コマンドバッファ C B R、C B L に書き込む (ステップ S 32)。次に、カメラ依存 (右目用仮想カメラ) の車用パラメータをシェーダに設定するコマンド C C D R を生成して、コマンドバッファ C B R に書き込む (ステップ S 33)。またカメラ依存 (左目用仮想カメラ) の車用パラメータをシェーダに設定するコマンド C C D L を生成して、コマンドバッファ C B L に書き込む (ステップ S 34)。

40

【0187】

次に、カメラ非依存の車用パラメータをシェーダに設定するコマンド C C N D を生成して、コマンドバッファ C B R、C B L に書き込む (ステップ S 35)。最後に、シェーダを実行するためのコマンド C S E X を生成して、コマンドバッファ C B R、C B L に書き込む (ステップ S 36)。

【0188】

次に、必要であれば、その他の表示物 (Z D パネル等) の描画処理を行う (ステップ S 37)。

50

## 【0189】

次に、図15(A)で説明したように、描画バッファDB(DB1又はDB2)からバッファTBの第1の領域AR1に右目用画像IRをコピーするコマンドを生成して、コマンドバッファCBRに書き込む(ステップS38)。また図15(B)で説明したように、描画バッファDB(DB1又はDB2)からバッファTBの第2の領域AR2に左目用画像ILをコピーするコマンドを生成して、コマンドバッファCBLに書き込む(ステップS39)。

## 【0190】

次に、図11で説明したように、コマンドバッファCBLへのジャンプコマンドJPLを生成して、コマンドバッファCBRに書き込む(ステップS40)。またコマンドバッファCBMへのジャンプコマンドJPLMを生成して、コマンドバッファCBLに書き込む(ステップS41)。

10

## 【0191】

次に、図17(A)、図17(B)で説明したように、バッファTBから描画バッファDB(DB1又はDB2)に画像を書き戻すコマンドを生成して、コマンドバッファCBMに書き込む(ステップS42)。そして、VSYNC(フレーム更新タイミング)をウエイトするコマンドを生成して、コマンドバッファCBMに書き込む(ステップS43)。更に、バックバッファとフロントバッファの切り替えコマンドを生成して、コマンドバッファCBMに書き込む(ステップS44)。最後に、バッファ切り替えをウエイトするコマンドを生成して、コマンドバッファCBMに書き込む(ステップS45)。

20

## 【0192】

## 3. ハードウェア構成

図21(A)に本実施形態を実現できるハードウェアの構成例を示す。

## 【0193】

CPU900(メインプロセッサ)は、複数のCPUコア1、CPUコア2、CPUコア3を含むマルチコアプロセッサである。またCPU900は図示しないキャッシュメモリを含む。CPUコア1、2、3の各々にはベクタ演算器等が設けられている。そしてCPUコア1、2、3の各々は、例えば2つのH/Wスレッド処理をコンテキストスイッチをすることなしに並列実行でき、マルチスレッド機能がハードウェアでサポートされている。そして3つのCPUコア1、2、3の合計で、6つのH/Wスレッド処理を並列実行できる。

30

## 【0194】

GPU910(描画プロセッサ)は、頂点処理やピクセル処理を行って、描画(レンダリング)処理を実現する。具体的には、シェーダプログラムに従って、頂点データの作成・変更やピクセル(フラグメント)の描画色の決定を行う。1フレーム分の画像がVRAM920(フレームバッファ)に書き込まれると、その画像はビデオ出力を介してTVなどのディスプレイに表示される。なおメインメモリ930はCPU900やGPU910のワークメモリとして機能する。またGPU910では、複数の頂点スレッド、複数のピクセルスレッドが並列実行され、描画処理のマルチスレッド機能がハードウェアでサポートされている。またGPU910にはハードウェアのテッセレータも備えられている。またGPU910は、頂点シェーダとピクセルシェーダとがハードウェア的に区別されていないユニファイド・シェーダ・タイプとなっている。

40

## 【0195】

ブリッジ回路940(サウスブリッジ)は、システム内部の情報流通を制御する回路であり、USBコントローラ(シリアルインターフェース)、ネットワークの通信コントローラ、IDEコントローラ、或いはDMAコントローラなどのコントローラを内蔵する。そしてこのブリッジ回路940により、ゲームコントローラ942、メモリーカード944、HDD946、DVDドライブ948との間のインターフェース機能が実現される。

## 【0196】

なお本実施形態を実現できるハードウェア構成は図21(A)に限定されず、例えば図

50

21(B)のような構成であってもよい。

【0197】

図21(B)ではCPU902が、プロセッサエレメントPPと8つのプロセッサエレメントPE1~PE8を含む。プロセッサエレメントPPは汎用的なプロセッサコアであり、プロセッサエレメントPE1~PE8は比較的シンプルな構成のプロセッサコアである。そしてプロセッサエレメントPPとプロセッサエレメントPE1~PE8のアーキテクチャは異なっており、プロセッサエレメントPE1~PE8は、複数のデータに対して1命令で同じ処理を同時にできるSIMD型のプロセッサコアとなっている。これによりストリーミング処理などのマルチメディア処理を効率的に実行できる。プロセッサエレメントPPは、2つのH/Wスレッド処理を並列実行でき、プロセッサエレメントPE1~PE8の各々は、1つのH/Wスレッド処理を実行できる。従って、CPU902では、合計で10個のH/Wスレッド処理の並列実行が可能になる。

10

【0198】

図21(B)では、GPU912とCPU902の連携が密になっており、GPU912は、CPU902側のメインメモリ930に直接にレンダリング処理を行うことができる。また例えばCPU902がジオメトリ処理を行って、頂点データを転送したり、GPU912からCPU902にデータを戻すことも容易にできる。またCPU902が、レンダリングのプリプロセッシング処理やポストプロセッシング処理を行うことも容易であり、テッセレーション(平面分割)やドットフィルをCPU902が実行できる。例えば抽象度の高い処理はCPU902が行い、抽象度が低い細かな処理はGPU912が行う

20

【0199】

なお本実施形態の各部の処理をハードウェアとプログラムにより実現する場合には、情報記憶媒体には、ハードウェア(コンピュータ)を本実施形態の各部として機能させるためのプログラムが格納される。より具体的には、上記プログラムが、ハードウェアであるプロセッサ(CPU、GPU)に処理を指示すると共に、必要であればデータを渡す。そして、プロセッサは、その指示と渡されたデータとに基づいて本発明の各部の処理を実現する。

【0200】

なお、上記のように本実施形態について詳細に説明したが、本発明の新規事項および効果から実体的に逸脱しない多くの変形が可能であることは当業者には容易に理解できるであろう。従って、このような変形例はすべて本発明の範囲に含まれるものとする。例えば、明細書又は図面において、少なくとも一度、より広義または同義な異なる用語(第1の視点画像、第2の視点画像、第1の視点、第2の視点等)と共に記載された用語(右目用画像、左目用画像、右目、左目等)は、明細書又は図面のいかなる箇所においても、その異なる用語に置き換えることができる。

30

【0201】

また、コマンドの生成・書き込み処理、コマンドの実行処理、第1、第2の視点画像の描画処理等も、本実施形態で説明したものに限定されず、これらと均等な手法も本発明の範囲に含まれる。また本発明は種々のゲームに適用できる。また本発明は、業務用ゲームシステム、家庭用ゲームシステム、多数のプレーヤが参加する大型アトラクションシステム、シミュレータ、マルチメディア端末、ゲーム画像を生成するシステムボード、携帯電話等の種々の画像生成システムに適用できる。

40

【図面の簡単な説明】

【0202】

【図1】本実施形態の画像生成システムのブロック図の例。

【図2】本実施形態が採用する立体視方式の一例を説明する図。

【図3】本実施形態が採用する立体視方式の一例を説明する図。

【図4】図4(A)、図4(B)は本実施形態により生成された右目用画像、左目用画像の例。

50

【図5】本実施形態により生成された多眼立体視画像の例。

【図6】本実施形態のコマンドの生成・実行手法の説明図。

【図7】コマンドの生成、書き込み処理の説明図。

【図8】図8(A)～図8(C)は共通コマンド、カメラ依存パラメータ、カメラ非依存パラメータの具体例。

【図9】図9(A)、図9(B)はコマンドの実行処理の説明図。

【図10】コマンドの実行処理の説明図。

【図11】ジャンプコマンドを用いる手法の説明図。

【図12】ジャンプコマンドを用いる手法の説明図。

【図13】右目用、左目用に共通に行うカリング処理の説明図。

10

【図14】多眼立体視画像生成手法の一例の説明図。

【図15】図15(A)、図15(B)は右目用画像、左目用画像のコピー処理の説明図

。

【図16】図16(A)、図16(B)はテクスチャマッピングを利用したコピー処理の説明図。

【図17】図17(A)、図17(B)はダブルバッファ方式を採用した場合の本実施形態の手法の説明図。

【図18】本実施形態の詳細な処理を説明するフローチャート。

【図19】本実施形態の詳細な処理を説明するフローチャート。

【図20】本実施形態の詳細な処理を説明するフローチャート。

20

【図21】図21(A)、図21(B)はハードウェア構成例。

【符号の説明】

【0203】

C B R 第1の視点用のコマンドバッファ、C B L 第2の視点用のコマンドバッファ、

C B M 共通コマンドバッファ、

D B 描画バッファ、D B 1、D B 2 第1、第2の描画バッファ、

T B バッファ、A R 1、A R 2 第1、第2の領域、M S モデルデータ記憶部、

T S テクスチャ記憶部、M T S モーションデータ記憶部、

1 0 0 処理部、1 0 2 ゲーム演算部、1 0 4 オブジェクト空間設定部、

1 0 5 カリング処理部、1 0 6 移動体演算部、1 0 8 仮想カメラ制御部、

30

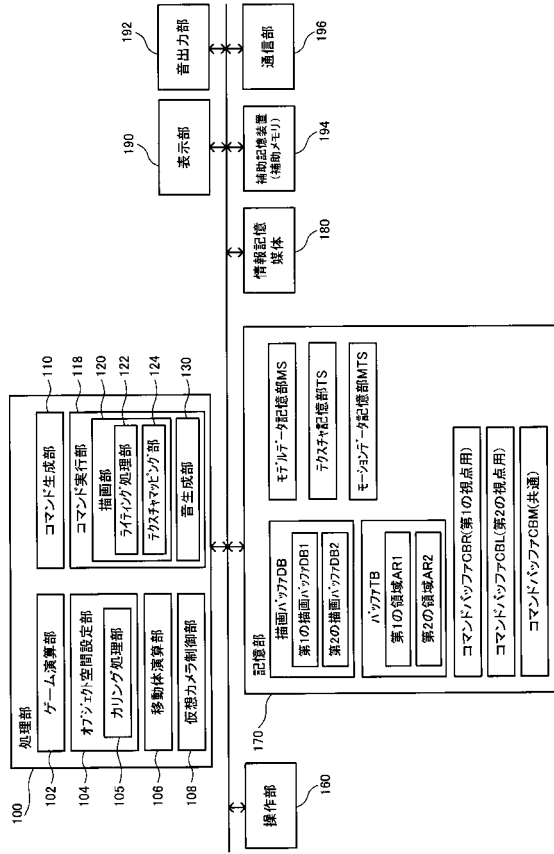
1 1 0 コマンド生成部、1 1 8 コマンド実行部、1 2 0 描画部、

1 2 2 ライティング処理部、1 2 4 テクスチャマッピング部、

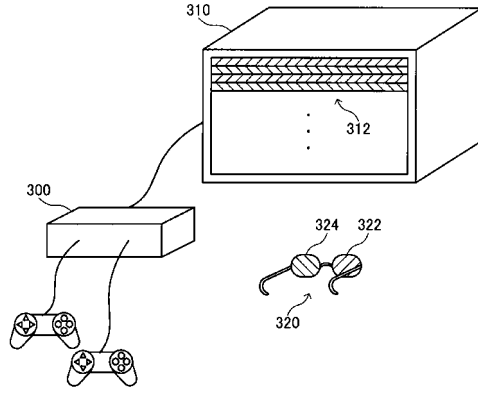
1 3 0 音生成部、1 6 0 操作部、1 7 0 記憶部、1 8 0 情報記憶媒体、

1 9 0 表示部、1 9 2 音出力部、1 9 4 補助記憶装置、1 9 6 通信部

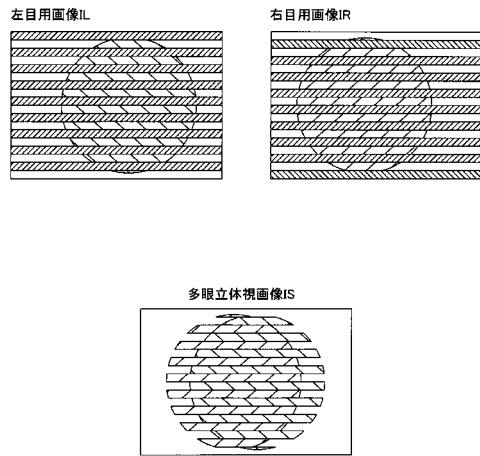
【図1】



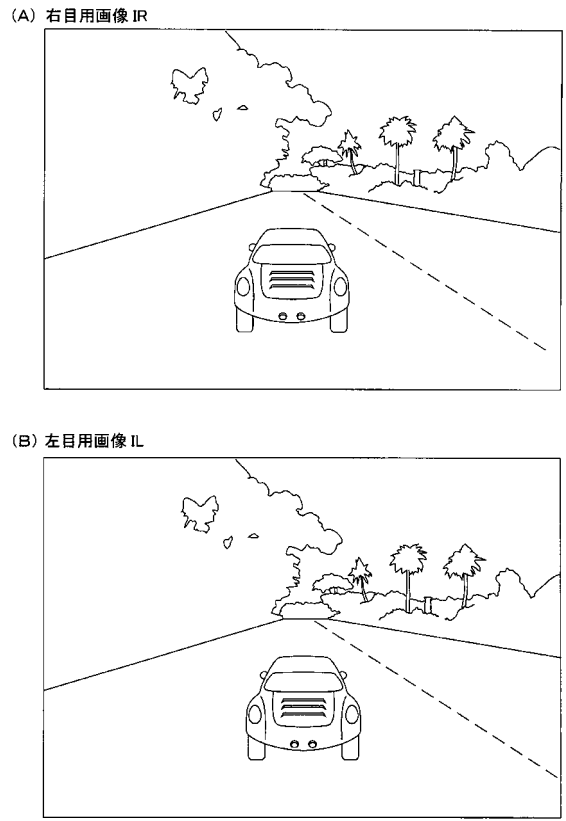
【図2】



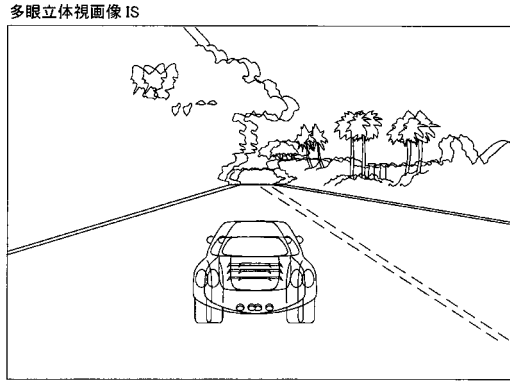
【図3】



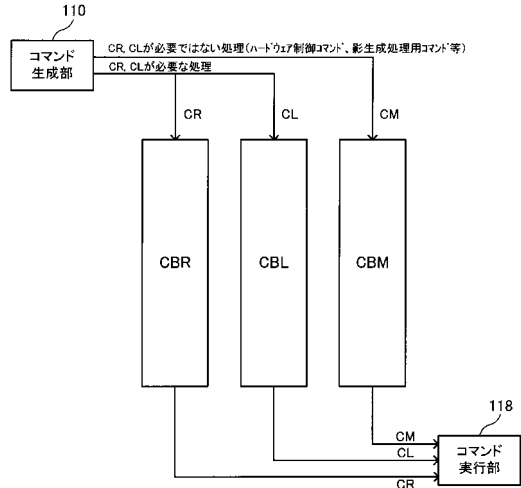
【図4】



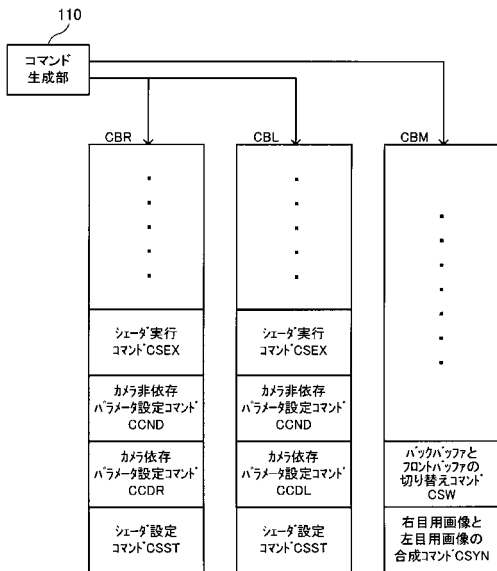
【図5】



【図6】



【図7】

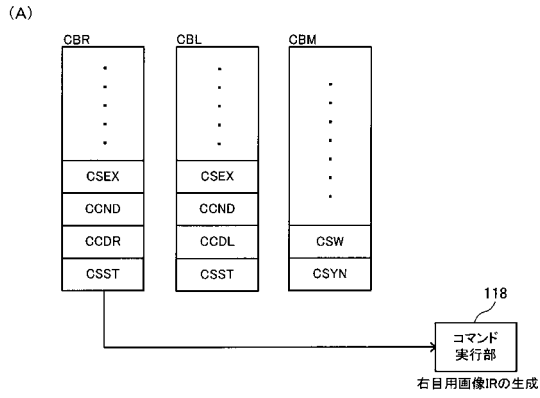


【図8】

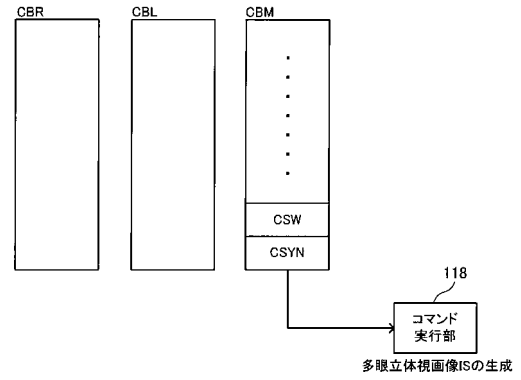
- (A) 共通コマンド
- | 影生成処理用コマンド   |  |
|--|--|
| ハードウェア制御コマンド<br>シェーディング方法、カリング方法、<br>αブレンディング、デプステスト、<br>ステンシルテスト等の設定、<br>バックバッファとフロントバッファの切り替え、<br>描画バッファのクリア |  |
| ⋮  |  |
- (B) カメラ依存パラメータ
- |                                      |
|--------------------------------------|
| ワールドビュー行列<br>(ワールド座標系→カメラ座標系)        |
| ワールドビュープロジェクション行列<br>(ワールド座標系→射影座標系) |
| カメラの位置ベクトル<br>(鏡面反射等に使用)             |
| ⋮                                    |
- (C) カメラ非依存パラメータ
- |                             |
|-----------------------------|
| ワールド行列<br>(ローカル座標系→ワールド座標系) |
| モーション用行列                    |
| ライトの位置ベクトル                  |
| 環境色、スペキュラ色                  |
| フォグ色                        |
| 頂点データ                       |
| テクスチャデータ                    |
| ⋮                           |



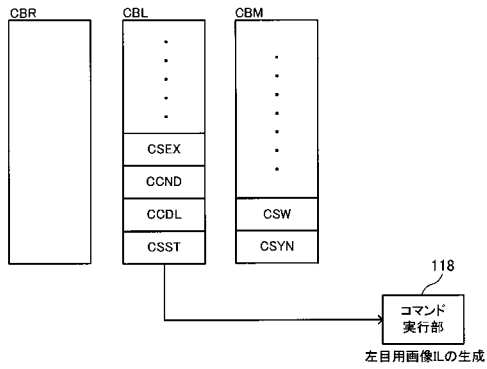
【図9】



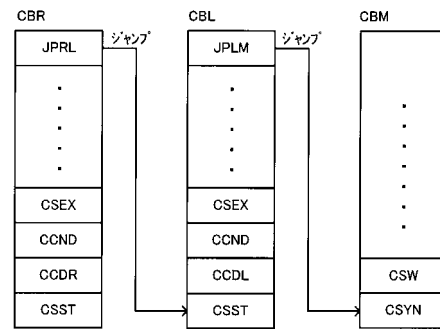
【図10】



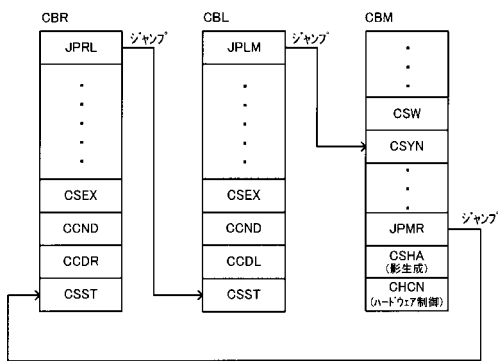
(B)



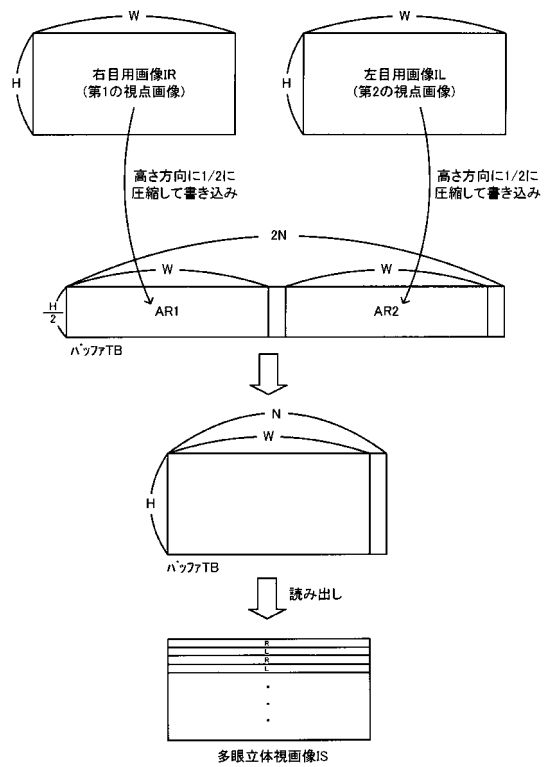
【図11】



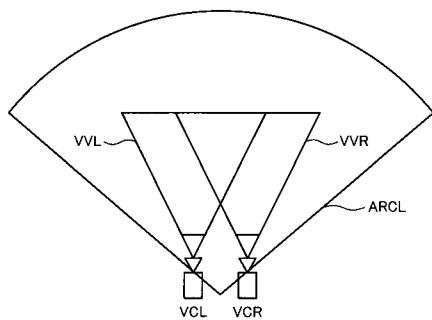
【図12】



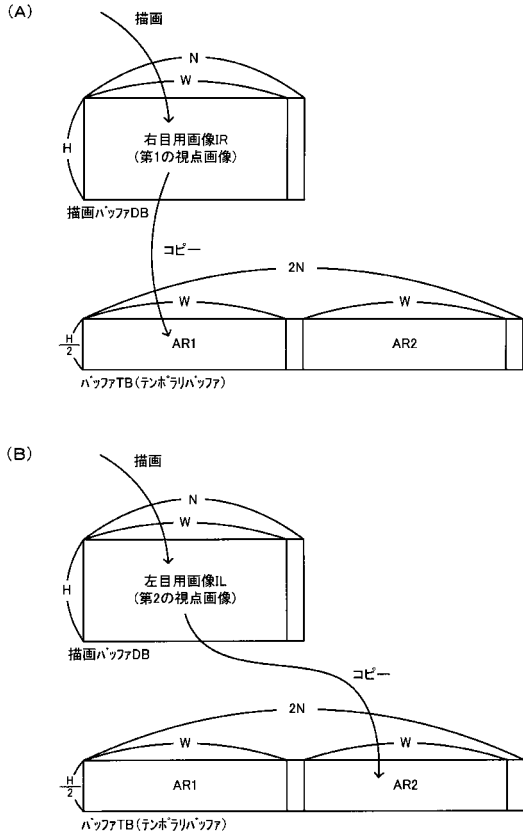
【図14】



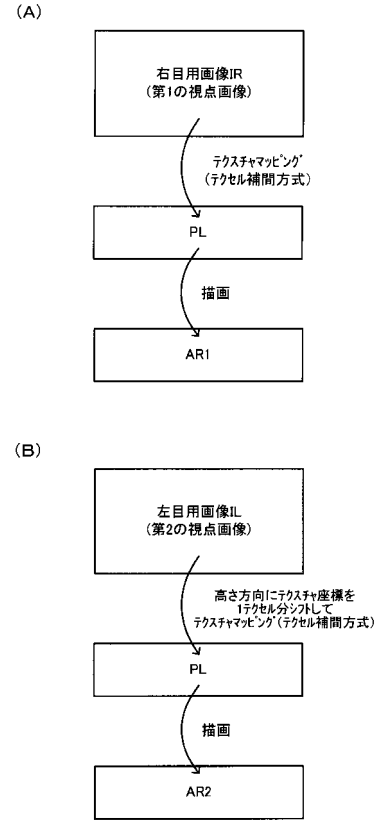
【図13】



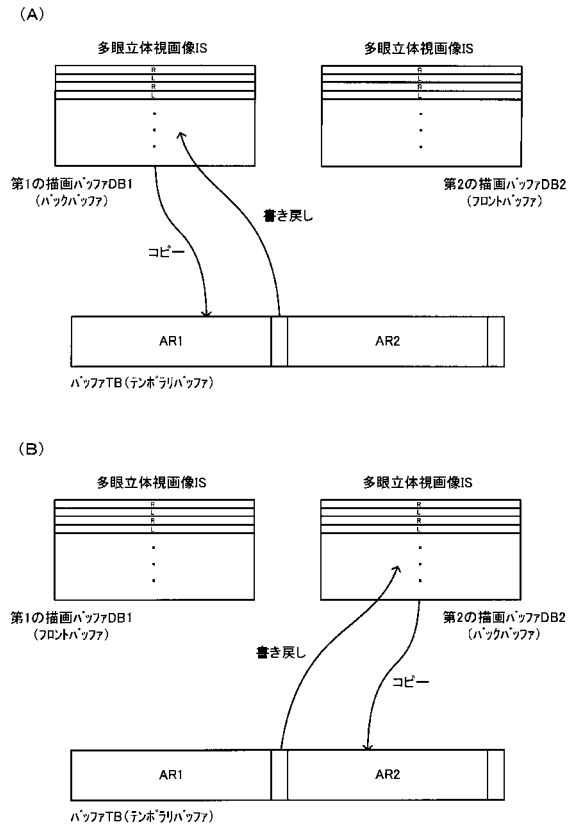
【図15】



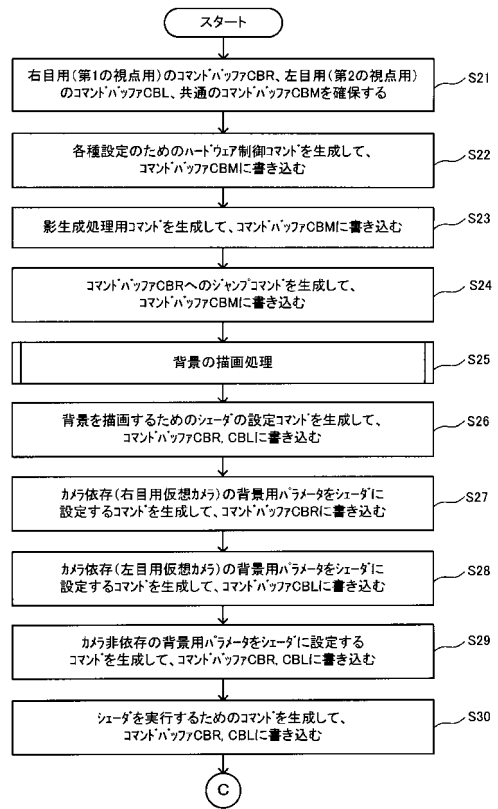
【図16】



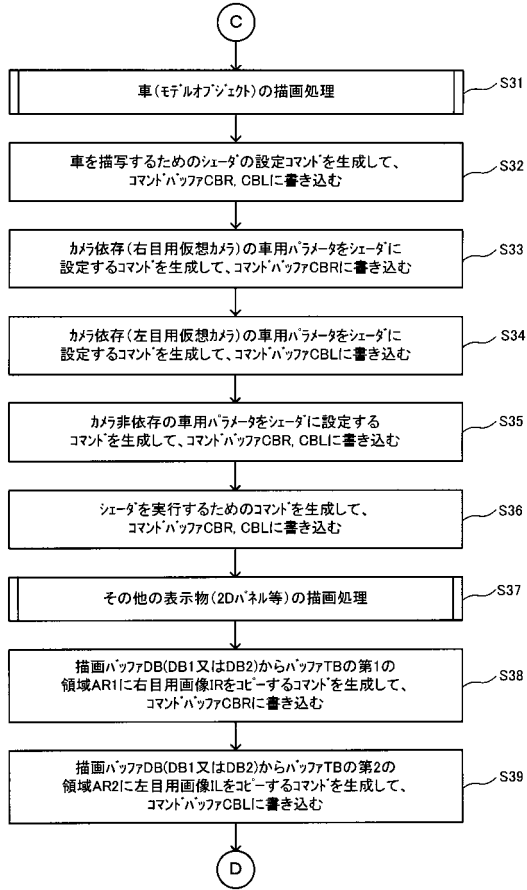
【図17】



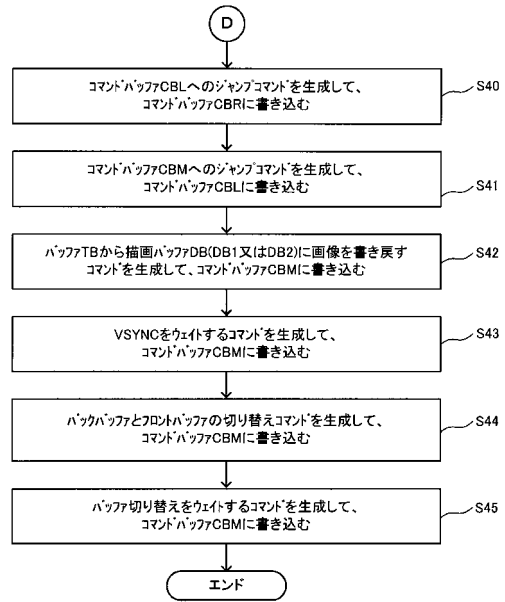
【図18】



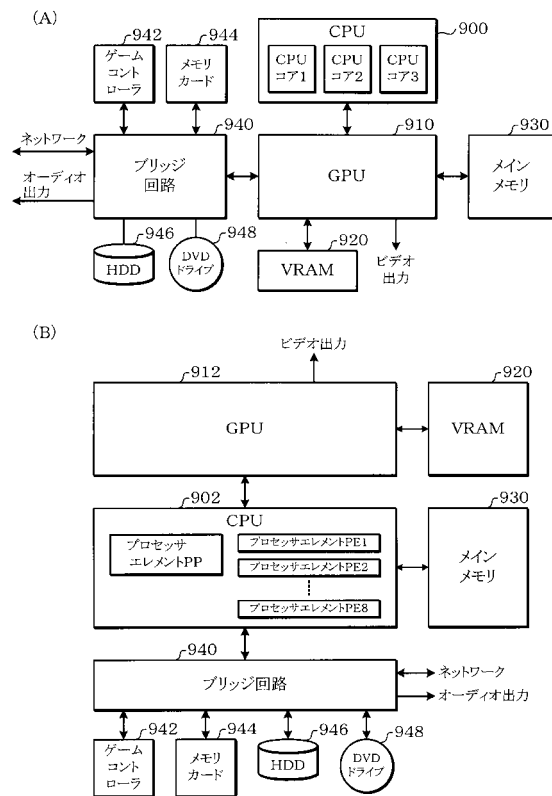
【図19】



【図20】



【図21】



## フロントページの続き

(72)発明者 大戸 友博

東京都渋谷区渋谷三丁目27番11号 祐真ビル新館2階 株式会社セリウス内

(72)発明者 坪崎 瞬

東京都渋谷区渋谷三丁目27番11号 祐真ビル新館2階 株式会社セリウス内

審査官 千葉 久博

(56)参考文献 特開2004-178581(JP,A)

特開2004-126902(JP,A)

特開2003-316345(JP,A)

特開2003-230707(JP,A)

特開2000-293155(JP,A)

特開平07-239952(JP,A)

特開平07-200870(JP,A)

特開平04-160483(JP,A)

川西裕幸, "OpenGL入門-[第1回]", 日経CG, 日本, 日経BP社, 1995年 1月 1日, 第100号, p.203-209

(58)調査した分野(Int.Cl., DB名)

G06T 19/00, 19/20

G06T 15/00-15/87

A63F 9/24, 13/00-13/12

G09G 5/00-5/42