

(19) 日本国特許庁(JP)

## (12) 特 許 公 報(B2)

(11) 特許番号

特許第6273380号  
(P6273380)

(45) 発行日 平成30年1月31日(2018.1.31)

(24) 登録日 平成30年1月12日(2018.1.12)

(51) Int.Cl.

G 0 6 T 15/06 (2011.01)

F 1

G O 6 T 15/06

請求項の数 21 (全 20 頁)

(21) 出願番号 特願2016-563169 (P2016-563169)  
 (86) (22) 出願日 平成27年4月17日 (2015.4.17)  
 (65) 公表番号 特表2017-516209 (P2017-516209A)  
 (43) 公表日 平成29年6月15日 (2017.6.15)  
 (86) 國際出願番号 PCT/US2015/026406  
 (87) 國際公開番号 WO2015/164204  
 (87) 國際公開日 平成27年10月29日 (2015.10.29)  
 審査請求日 平成29年8月28日 (2017.8.28)  
 (31) 優先権主張番号 14/257,385  
 (32) 優先日 平成26年4月21日 (2014.4.21)  
 (33) 優先権主張国 米国(US)

早期審査対象出願

(73) 特許権者 507364838  
 クアルコム、インコーポレイテッド  
 アメリカ合衆国 カリフォルニア 921  
 21 サンディエゴ モアハウス ドラ  
 イブ 5775  
 (74) 代理人 100108453  
 弁理士 村山 靖彦  
 (74) 代理人 100163522  
 弁理士 黒田 晋平  
 (72) 発明者 ユライ・オペール  
 アメリカ合衆国・カリフォルニア・921  
 21-1714・サン・ディエゴ・モアハ  
 ウス・ドライブ・5775

最終頁に続く

(54) 【発明の名称】レイトレーシングアプリケーションにおけるツリー横断のための開始ノード決定

## (57) 【特許請求の範囲】

## 【請求項1】

少なくとも1つのプロセッサによって、階層データ構造体内に複数の**バウンディングボリューム**を編成するステップと、

前記少なくとも1つのプロセッサによって、前記階層データ構造体の非ルートノードに関連する**前記複数のバウンディングボリュームの表示**をオフスクリーンレンダリングターゲットに**ピクセルとしてラスタライズ**するステップと、

前記少なくとも1つのプロセッサによって、かつ、**レイにマッピングされる前記オフスクリーンレンダリングターゲットに、ラスタライズされた前記ピクセルのうちの1つの色値をサンプリング**することに少なくとも部分的に基づいて、前記階層データ構造体の横断を開始するための前記階層データ構造体内の開始ノードとして前記階層データ構造体の非ルートノードを決定するステップと、

前記少なくとも1つのプロセッサによって、**前記レイが前記複数のバウンディングボリュームのうちの1つによってバウンディングされるプリミティブを交差するか否かを決定**するために前記開始ノードから開始して前記階層データ構造体を横断するステップとを含む、方法。

## 【請求項2】

前記少なくとも1つのプロセッサによって、前記開始ノードとして**前記ピクセルのうちの前記1つに**関連する前記階層データ構造体のノードを決定するステップをさらに含む、請求項1に記載の方法。

**【請求項 3】**

前記少なくとも1つのプロセッサによって、前記複数のバウンディングボリュームの前記表示のうちの2つ以上が重複すると判断するステップと、

前記少なくとも1つのプロセッサによって、最も低い共通の先祖の表示の色値を有する前記複数のバウンディングボリュームの前記表示のうちの前記2つ以上の重複の領域を充填するステップと、

前記少なくとも1つのプロセッサによって、前記開始ノードとして前記最も低い共通の先祖の表示の前記色値に関連付けられた前記階層データ構造体のノードを決定するステップと

をさらに含む、請求項1に記載の方法。

10

**【請求項 4】**

前記少なくとも1つのプロセッサによって、前記複数のバウンディングボリュームの前記表示を、前記複数のバウンディングボリュームよりも比較的低い解像度で前記オフスクリーンレンダリングターゲットにラスタライズするステップをさらに含む、請求項1に記載の方法。

**【請求項 5】**

前記複数のバウンディングボリュームが、前記階層データ構造体の前記非ルートノードのすべてよりも少なく関連する、請求項1に記載の方法。

**【請求項 6】**

前記階層データ構造体の前記非ルートノードに関連する前記複数のバウンディングボリュームの前記表示が、前記複数のバウンディングボリュームによってバウンディングされるプリミティブの表示を含まない、請求項1に記載の方法。

20

**【請求項 7】**

グラフィックスデータを処理するように構成された装置であって、階層データ構造体内に編成された複数のバウンディングボリュームを記憶するように構成されたメモリと、

前記階層データ構造体の非ルートノードに関連する前記複数のバウンディングボリュームの表示をオフスクリーンレンダリングターゲットにピクセルとしてラスタライズすることと、

レイにマッピングされる前記オフスクリーンレンダリングターゲットに、ラスタライズされた前記ピクセルのうちの1つの色値をサンプリングすることに少なくとも部分的に基づいて、前記階層データ構造体の横断を開始するための前記階層データ構造体の開始ノードとして前記階層データ構造体の非ルートノードを決定することと、

前記レイが前記複数のバウンディングボリュームのうちの1つによってバウンディングされるプリミティブを交差するか否かを決定するために前記開始ノードから開始して前記階層データ構造体を横断することと

を行うように構成された少なくとも1つのプロセッサとを含む、装置。

**【請求項 8】**

前記少なくとも1つのプロセッサが、前記開始ノードとして前記ピクセルのうちの前記1つに関連する前記階層データ構造体のノードを決定することを行なうようにさらに構成される、請求項7に記載の装置。

40

**【請求項 9】**

前記少なくとも1つのプロセッサが、前記複数のバウンディングボリュームの前記表示のうちの2つ以上が重複すると判断することと、

最も低い共通の先祖の表示の色値を有する前記複数のバウンディングボリュームの前記表示のうちの前記2つ以上の重複の領域を充填することと、

前記開始ノードとして前記最も低い共通の先祖の表示の前記色値に関連付けられた前記階層データ構造体のノードを決定することと

50

を行うようにさらに構成される、請求項7に記載の装置。

**【請求項 1 0】**

前記少なくとも1つのプロセッサが、

前記複数のバウンディングボリュームの前記表示を、前記複数のバウンディングボリュームよりも比較的低い解像度で前記オフスクリーンレンダリングターゲットにラスタライズする

ようにさらに構成される、請求項7に記載の装置。

**【請求項 1 1】**

前記複数のバウンディングボリュームが、前記階層データ構造体の前記非ルートノードのすべてよりも少なく関連する、請求項7に記載の装置。 10

**【請求項 1 2】**

階層データ構造体内に複数のバウンディングボリュームを編成するための手段と、

前記階層データ構造体の非ルートノードに関連する前記複数のバウンディングボリュームの表示をオフスクリーンレンダリングターゲットにピクセルとしてラスタライズするための手段と、

レイにマッピングされる前記オフスクリーンレンダリングターゲットに、ラスタライズされた前記ピクセルのうちの1つの色値をサンプリングすることに少なくとも部分的に基づいて、前記階層データ構造体の横断を開始するための前記階層データ構造体の開始ノードとして前記階層データ構造体の非ルートノードを決定するための手段と、

前記レイが前記複数のバウンディングボリュームのうちの1つによってバウンディングされるプリミティブを交差するか否かを決定するために前記開始ノードから開始して前記階層データ構造体を横断するための手段とを含む、装置。 20

**【請求項 1 3】**

前記開始ノードとして前記ピクセルのうちの前記1つに関連する前記階層データ構造体のノードを決定するための手段をさらに含む、請求項12に記載の装置。

**【請求項 1 4】**

前記複数のバウンディングボリュームの前記表示のうちの2つ以上が重複すると判断するための手段と、

最も低い共通の先祖の表示の色値を有する前記複数のバウンディングボリュームの前記表示のうちの前記2つ以上の重複の領域を充填するための手段と、 30

前記開始ノードとして前記最も低い共通の先祖の表示の前記色値に関連付けられた前記階層データ構造体のノードを決定するための手段とをさらに含む、請求項12に記載の装置。

**【請求項 1 5】**

前記複数のバウンディングボリュームの前記表示を、前記複数のバウンディングボリュームよりも比較的低い解像度で前記オフスクリーンレンダリングターゲットにラスタライズするための手段

をさらに含む、請求項12に記載の装置。

**【請求項 1 6】**

前記複数のバウンディングボリュームが、前記階層データ構造体の前記非ルートノードのすべてよりも少なく関連する、請求項12に記載の装置。 40

**【請求項 1 7】**

実行されたとき、1つまたは複数のプログラマブルプロセッサに、

少なくとも1つのプロセッサによって、階層データ構造体内に複数のバウンディングボリュームを編成することと、

前記少なくとも1つのプロセッサによって、前記階層データ構造体の非ルートノードに関連する前記複数のバウンディングボリュームの表示をオフスクリーンレンダリングターゲットにピクセルとしてラスタライズすることと、

前記少なくとも1つのプロセッサによって、かつ、レイにマッピングされる前記オフス 50

クリーンレンダリングターゲットに、ラスタライズされた前記ピクセルのうちの1つの色値をサンプリングすることに少なくとも部分的に基づいて、前記階層データ構造体の横断を開始するための前記階層データ構造体内の開始ノードとして前記階層データ構造体の非ルートノードを決定することと、

前記少なくとも1つのプロセッサによって、前記レイが前記複数のバウンディングボリュームのうちの1つによってバウンディングされるプリミティブを交差するか否かを決定するために前記開始ノードから開始して前記階層データ構造体を横断することとを行わせる命令を記憶する、非一時的なコンピュータ可読記憶媒体。

#### 【請求項18】

実行されたとき、1つまたは複数のプログラマブルプロセッサに、  
前記開始ノードとして前記ピクセルのうちの前記1つに関連する前記階層データ構造体のノードを決定すること  
を行わせる命令をさらに含む、請求項17に記載のコンピュータ可読記憶媒体。

#### 【請求項19】

実行されたとき、1つまたは複数のプログラマブルプロセッサに、  
前記複数のバウンディングボリュームの前記表示のうちの2つ以上が重複すると判断することと、  
最も低い共通の先祖の表示の色値を有する前記複数のバウンディングボリュームの前記表示のうちの前記2つ以上の重複の領域を充填することと、  
前記開始ノードとして前記最も低い共通の先祖の表示の前記色値に関連付けられた前記階層データ構造体のノードを決定することと  
を行わせる命令をさらに含む、請求項17に記載のコンピュータ可読記憶媒体。

#### 【請求項20】

実行されたとき、1つまたは複数のプログラマブルプロセッサに、  
前記複数のバウンディングボリュームの前記表示を、前記複数のバウンディングボリュームよりも比較的低い解像度で前記オフスクリーンレンダリングターゲットにラスタライズさせる  
命令をさらに含む、請求項17に記載のコンピュータ可読記憶媒体。

#### 【請求項21】

前記複数のバウンディングボリュームが、前記階層データ構造体の前記非ルートノードのすべてよりも少なく関連する、請求項17に記載のコンピュータ可読記憶媒体。

#### 【発明の詳細な説明】

##### 【技術分野】

##### 【0001】

本開示は、レイトレーシングのレイプリミティブ交点を決定するために階層データ構造体(hierarchical data structure)を横断する(traverse)ことを含む、グラフィックス処理に関する。

##### 【背景技術】

##### 【0002】

コンピュータグラフィックスでは、レイトレーシングは、3Dシーンを介して3次元(3D)レイをトレースし、レイプリミティブ交点を見つける処理である。レイプリミティブ交点を見つける処理を加速させるために、シーンプリミティブを階層的にグループ化する加速データ構造体(ADS: acceleration data structure)内にシーンジオメトリが配置される。ADSは、レイが交差する、シーンのプリミティブを決定するためにシーンプリミティブの階層上でレイ交差テストを実行することによって再帰的に横断され得る。

#### 【発明の概要】

#### 【課題を解決するための手段】

##### 【0003】

グラフィックス処理ユニット(GPU)は、ルートノードからADSを横断するのではなく、代わりに、内部ノードからADSの横断を開始し、それによって、GPUが階層上で実行し得るレ

10

20

30

40

50

イ交差テストの数を低減し得る。GPUは、階層データ構造体の内部ノードに関連するバウンディングボリュームのサブセットをオフスクリーンレンダリングターゲットにラスタライズすることによって横断を開始する、ADSの内部ノードを決定し得る。GPUは、オフスクリーンレンダリングターゲットからサンプリングされた色値から、レイが交差する内部ノードのノードインデックスを決定する場合があり、そのノードからADSの横断を開始し、それによって、ADSを横断するためにGPUが実行し得るレイ交差テストの数を低減し得る。

#### 【0004】

本開示の一例では、一方法は、少なくとも1つのプロセッサによって、階層データ構造体内に複数のプリミティブを編成するステップを含み得る。本方法は、少なくとも1つのプロセッサによって、階層データ構造体の非ルートノードに関連する複数のバウンディングボリュームをオフスクリーンレンダリングターゲットにラスタライズするステップをさらに含み得る。本方法は、少なくとも1つのプロセッサによって、また複数のバウンディングボリュームをオフスクリーンレンダリングターゲットにラスタライズするステップに基づいて、階層データ構造体の横断を開始するための階層データ構造体内の開始ノードとして階層データ構造体の非ルートノードを決定するステップをさらに含み得る。本方法は、少なくとも1つのプロセッサによって、レイが交差するプリミティブを決定するために開始ノードから開始して階層データ構造体を横断するステップをさらに含み得る。

10

#### 【0005】

本開示の別の例では、グラフィックス処理のための装置は、メモリと、階層データ構造体内に複数のプリミティブを編成することと、階層データ構造体の非ルートノードに関連する複数のバウンディングボリュームをオフスクリーンレンダリングターゲットにラスタライズすることと、複数のバウンディングボリュームをオフスクリーンレンダリングターゲットにラスタライズすることに基づいて、階層データ構造体の横断を開始するための階層データ構造体内の開始ノードとして階層データ構造体の非ルートノードを決定することと、レイが交差するプリミティブを決定するために開始ノードから開始して階層データ構造体を横断することとを行うように構成された少なくとも1つのプロセッサとを含み得る。

20

#### 【0006】

本開示の別の例では、グラフィックス処理のための装置は、階層データ構造体内に複数のプリミティブを編成するための手段を含み得る。本装置は、階層データ構造体のノードに関連する複数のバウンディングボリュームをオフスクリーンレンダリングターゲットにラスタライズするための手段をさらに含み得る。本装置は、複数のバウンディングボリュームをオフスクリーンレンダリングターゲットにラスタライズすることに基づいて、階層データ構造体の横断を開始するための階層データ構造体内の開始ノードとして階層データ構造体の非ルートノードを決定するための手段をさらに含み得る。本装置は、レイが交差するプリミティブを決定するために開始ノードから開始して階層データ構造体を横断するための手段をさらに含み得る。

30

#### 【0007】

本開示の別の例では、コンピュータ可読記憶媒体は、実行されたとき、1つまたは複数のプログラマブルプロセッサに、少なくとも1つのプロセッサによって階層データ構造体内に複数のプリミティブを編成することと、少なくとも1つのプロセッサによって階層データ構造体の非ルートノードに関連する複数のバウンディングボリュームをオフスクリーンレンダリングターゲットにラスタライズすることと、少なくとも1つのプロセッサによって、また複数のバウンディングボリュームをオフスクリーンレンダリングターゲットにラスタライズすることに基づいて、階層データ構造体の横断を開始するための階層データ構造体内の開始ノードとして階層データ構造体の非ルートノードを決定することと、少なくとも1つのプロセッサによってレイが交差するプリミティブを決定するために開始ノードから開始して階層データ構造体を横断することを行わせる命令を記憶し得る。

40

#### 【0008】

1つまたは複数の例の詳細が、以下の添付の図面および説明において述べられる。他の

50

特徴、目的、および利点は、説明、図面、および特許請求の範囲から明らかになるであろう。

**【図面の簡単な説明】**

**【0009】**

【図1】本開示の1つまたは複数の態様を実装するように構成され得る例示的なコンピューティングデバイスを示すブロック図である。

【図2】図1のCPU、GPU、およびシステムメモリの例示的な実装形態をさらに詳細に示すブロック図である。

【図3】GPUがレイトレーシングを実行し得る例示的なシーンを示すブロック図である。

【図4】例示的なシーンをバウンディングボリュームに区分することを示すブロック図である。10

【図5】図4に示すバウンディングボリュームおよびプリミティブに関連付けられたノードを有する例示的な階層データ構造体を示す図である。

【図6】例示的なオフスクリーンレンダリングターゲットにラスタライズされたバウンディングボリュームの例示的な表示を示すブロック図である。

【図7】レイプリミティブ交点を見つけるために例示的な階層ツリー構造体を横断するための開始ノードを決定するための例示的な処理を示すフローチャートである。

**【発明を実施するための形態】**

**【0010】**

一般に、本開示は、階層データ構造体の内部ノードからシーンプリミティブの階層データ構造体の横断を開始することによってレイトレーシングの目的でレイが交差するプリミティブを決定するための技法について説明する。グラフィカル処理ユニット(GPU)上で加速データ構造体(ADS)などの階層データ構造体を横断するとき、ルートノードから階層データ構造体の横断を開始することは、準最適な性能をもたらし得るが、その理由は、そのような横断が、階層データ構造体内のシーンプリミティブにおいて無関係のレイ交差テストを実行することをGPUに求める場合があるからである。20

**【0011】**

GPUは、ルートノードから階層データ構造体を横断するのではなく、代わりに、ルートノード以外のノードから階層データ構造体の横断を開始し、それによって、GPUが実行し得るレイ交差テストの数を低減し得る。GPUは、階層データ構造体の内部ノードおよびリーフノードに関連するバウンディングボリューム(bounding volume)のサブセットをオフスクリーンレンダリングターゲットにラスタライズするために、そのグラフィックス処理パイプラインからシェーダ(shader)ユニットを利用することによって横断を開始する、階層データ構造体の非ルートノードを決定し得る。GPUは、オフスクリーンレンダリングターゲットから、階層データ構造体の横断を開始する内部非ルートノードを決定し、それによって、階層データ構造体を横断するためにGPUが実行し得るレイ交差テストの数を低減し得る。オフスクリーンレンダリングターゲットにラスタライズすることが、レイ交差テストを実行することよりも比較的計算コストが低いので、GPUは、ルートノードから階層データ構造体を横断することと比較してレイトレーシング性能の大幅な増大を実現し得る。30

。

**【0012】**

本開示の態様によれば、GPUは、階層データ構造体内に複数のプリミティブを編成するように構成され得る。GPUは、階層データ構造体の非ルートノードに関連する複数のバウンディングボリュームをオフスクリーンレンダリングターゲットにラスタライズするようさら構成され得る。GPUは、複数のバウンディングボリュームからレイが交差するバウンディングボリュームを決定するようにさらに構成され得る。GPUは、階層データ構造体の横断を開始するための階層データ構造体内の開始ノードとして、バウンディングボリュームに関連付けられた、階層データ構造体の非ルートノードを決定するようにさらに構成され得る。GPUは、レイが交差するプリミティブを決定するために開始ノードから開始して階層データ構造体を横断するようにさらに構成され得る。40

50

## 【0013】

図1は、本開示の1つまたは複数の態様を実装するように構成され得る例示的なコンピューティングデバイスを示すブロック図である。図1に示すように、デバイス2は、限定はしないが、ビデオデバイス、メディアプレーヤ、セットトップボックス、携帯電話およびいわゆるスマートフォンなどのワイヤレスハンドセット、携帯情報端末(PDA)、デスクトップコンピュータ、ラップトップコンピュータ、ゲームコンソール、ビデオ会議ユニット、タブレットコンピューティングデバイスなどを含むコンピューティングデバイスであり得る。図1の例では、デバイス2は、中央処理ユニット(CPU)6、システムメモリ10、およびGPU12を含み得る。デバイス2は、ディスプレイプロセッサ14、トランシーバモジュール3、ユーザインターフェース4、およびディスプレイ8も含み得る。トランシーバモジュール3およびディスプレイプロセッサ14は、どちらもCPU6および/もしくはGPU12と同じ集積回路(IC)の一部である場合があるか、どちらもCPU6および/もしくはGPU12を含む1つもしくは複数のICの外部にある場合があるか、またはCPU6および/もしくはGPU12を含むICの外部にあるIC内に形成される場合がある。10

## 【0014】

デバイス2は、明快のために、図1には示さない追加のモジュールまたはユニットを含み得る。たとえば、デバイス2は、デバイス2がモバイルワイヤレス電話である例では電話通信をもたらし、デバイス2がメディアプレーヤである場合にはスピーカーをもたらすために、図1には示していない、スピーカーおよびマイクロфонを含み得る。デバイス2は、ビデオカメラも含み得る。さらに、デバイス2に示す様々なモジュールおよびユニットは、デバイス2のすべての例には必要でない場合がある。たとえば、ユーザインターフェース4およびディスプレイ8は、デバイス2が、デスクトップコンピュータ、または外部のユーザインターフェースもしくはディスプレイとインターフェースするために装備された他のデバイスである例ではデバイス2の外部にある場合がある。20

## 【0015】

ユーザインターフェース4の例としては、限定はしないが、トラックボール、マウス、キーボード、および他のタイプの入力デバイスがある。ユーザインターフェース4は、タッチスクリーンである場合もあり、ディスプレイ8の一部として組み込まれる場合がある。トランシーバモジュール3は、デバイス2と別のデバイスまたはネットワークとの間のワイヤレス通信または有線通信を可能にするための回路を含み得る。トランシーバモジュール3は、変調器、復調器、増幅器、および有線通信またはワイヤレス通信用の他のそのような回路を含み得る。30

## 【0016】

CPU6は、実行用のコンピュータプログラムの命令を処理するように構成された中央処理ユニット(CPU)などのマイクロプロセッサである場合がある。CPU6は、デバイス2の演算を制御する汎用プロセッサまたは専用プロセッサを含み得る。ユーザは、CPU6に1つまたは複数のソフトウェアアプリケーションを実行させるためにデバイス2に入力を提供し得る。CPU6上で実行するソフトウェアアプリケーションは、たとえば、オペレーティングシステム、ワードプロセッサアプリケーション、電子メールアプリケーション、スプレッドシートアプリケーション、メディアプレーヤアプリケーション、ビデオゲームアプリケーション、グラフィカルユーザインターフェースアプリケーション、または別のプログラムを含み得る。加えて、CPU6は、GPU12の演算を制御するためのGPUドライバ22を実行し得る。ユーザは、キーボード、マウス、マイクロфон、タッチパッド、またはユーザインターフェース4を介してデバイス2に結合される別の入力デバイスなどの1つまたは複数の入力デバイス(図示せず)を介してデバイス2に入力を提供し得る。40

## 【0017】

CPU6上で実行するソフトウェアアプリケーションは、グラフィックスデータのディスプレイ8へのレンダリングをもたらすようにCPU6に命令する1つまたは複数のグラフィックスレンダリング命令を含み得る。いくつかの例では、ソフトウェア命令は、たとえば、Open Graphics Library(OpenGL(登録商標)) API、Open Graphics Library Embedded Systems(50

OpenGL ES) API、Direct3D API、X3D API、RenderMan API、 WebGL API、または任意の他の公的規格もしくはプロプライエタリ規格のグラフィックスAPIなどのグラフィックスアプリケーションプログラミングインターフェース(API)に準拠し得る。グラフィックスレンダリング命令を処理するために、CPU6は、GPU12にグラフィックスデータのレンダリングのいくつかまたはすべてを実行させるように(たとえば、GPUドライバ22を介して)1つまたは複数のグラフィックスレンダリングコマンドをGPU12に発出し得る。いくつかの例では、レンダリングされるグラフィックスデータは、グラフィックスプリミティブ、たとえば、点、線、三角形、四辺形、三角形ストリップなどのリストを含み得る。

#### 【0018】

GPU12は、1つまたは複数のグラフィックスプリミティブをディスプレイ8にレンダリングするためにグラフィックス演算を実行するように構成され得る。したがって、CPU6上で実行するソフトウェアアプリケーションのうちの1つがグラフィックス処理を要求するとき、CPU6は、ディスプレイ8にレンダリングするためにグラフィックスコマンドおよびグラフィックスデータをGPU12に提供し得る。グラフィックスデータは、たとえば、描画コマンド、状態情報、プリミティブ情報、テクスチャ情報などを含み得る。GPU12は、いくつかの事例では、CPU6よりも効率的な複雑なグラフィック関連の演算の処理を提供する高度並列構造で構築され得る。たとえば、GPU12は、複数の頂点またはピクセル上で並列に動作するように構成されたシェーダユニットなどの複数の処理要素を含み得る。GPU12の高度並列性質は、いくつかの事例では、GPU12が、CPU6を使用してディスプレイ8にシーンを直接描画するよりも迅速にディスプレイ8上にグラフィックス画像(たとえば、GUIならびに2次元(2D)グラフィックスシーンおよび/または3次元(3D)グラフィックスシーン)を描画することを可能にし得る。10

#### 【0019】

GPU12は、いくつかの事例では、デバイス2のマザーボードに統合され得る。他の事例では、GPU12は、デバイス2のマザーボード内のポート内にインストールされたグラフィックスカード上に存在する場合があるか、または場合によっては、デバイス2と相互動作するように構成された周辺デバイス内に組み込まれる場合がある。GPU12は、1つもしくは複数のマイクロプロセッサ、特定用途向け集積回路(ASIC)、フィールドプログラマブルゲートアレイ(FPGA)、デジタル信号プロセッサ(DSP)、または他の均等な集積論理回路もしくは個別論理回路などの1つまたは複数のプロセッサを含み得る。GPU12は、1つまたは複数のプロセッサコアを含む場合もあり、その結果、GPU12は、マルチコアプロセッサと呼ばれる場合がある。20

#### 【0020】

GPU12は、グラフィックスメモリ40に直接結合され得る。したがって、GPU12は、バスを使用することなくグラフィックスメモリ40からデータを読み取り、グラフィックスメモリ40にデータを書き込む場合がある。言い換えれば、GPU12は、オフチップメモリの代わりに、ローカルストレージを使用してローカルにデータを処理し得る。そのようなグラフィックスメモリ40は、オンチップメモリと呼ばれる場合がある。このグラフィックスメモリ40は、GPU12が、重いバストラフィックを経る場合があるバスを介してGPU12がデータを読み書きする必要性を除去することによって、より効率的に動作することを可能にする。しかしながら、いくつかの事例では、GPU12は、別個のメモリを含まないが、代わりに、バスを介してシステムメモリ10を利用する場合がある。グラフィックスメモリ40は、たとえば、ランダムアクセスメモリ(RAM)、スタティックRAM(SRAM)、ダイナミックRAM(DRAM)、消去可能プログラマブルROM(EPROM)、電気消去可能プログラマブルROM(EEPROM)、フラッシュメモリ、磁気データ媒体、または光記憶媒体などの、1つまたは複数の揮発性または不揮発性のメモリもしくは記憶デバイスを含み得る。30

#### 【0021】

いくつかの例では、GPU12は、システムメモリ10内に完全に形成された画像を記憶し得る。ディスプレイプロセッサ14は、画像を表示するためにディスプレイ8のピクセルを照明させる、システムメモリ10からの画像および出力値を取り出す場合がある。ディスプレ40

イ8は、GPU12によって生成された画像コンテンツを表示するデバイス2のディスプレイである場合がある。ディスプレイ8は、液晶ディスプレイ(LCD)、有機発光ダイオードディスプレイ(OLED)、陰極線管(CRT)ディスプレイ、プラズマディスプレイ、または別のタイプのディスプレイデバイスである場合がある。

#### 【 0 0 2 2 】

本開示の態様によれば、GPU12は、階層データ構造体内に複数のプリミティブを編成(organize)し得る。GPU12は、階層データ構造体のノードに関連する1つまたは複数のバウンディングボリューム(bounding volume)をオフスクリーンレンダリングターゲットにラスター化し得る。GPU12は、階層データ構造体の横断を開始するための階層データ構造体内的開始ノードとして階層データ構造体の内部非ルートノード(interior non-root node)を決定し得る。GPU12は、レイが交差するプリミティブを決定するために開始ノードから開始して階層データ構造体を横断し得る。10

#### 【 0 0 2 3 】

図2は、図1のCPU6、GPU12、およびシステムメモリ10の例示的な実装形態をさらに詳細に示すブロック図である。図2に示すように、CPU6は、各々がCPU6上で実行する1つまたは複数のソフトウェアアプリケーションまたはサービスである場合がある、少なくとも1つのソフトウェアアプリケーション18、グラフィックスAPI20、およびGPUドライバ22を含み得る。

#### 【 0 0 2 4 】

CPU6およびGPU12が利用可能なメモリは、システムメモリ10およびフレームバッファ16を含み得る。フレームバッファ16は、システムメモリ10の一部である場合があるか、またはシステムメモリ10とは別である場合がある。フレームバッファ16は、レンダリングされた画像データを記憶し得る。20

#### 【 0 0 2 5 】

ソフトウェアアプリケーション18は、GPU12の機能を利用する任意のアプリケーションである場合がある。たとえば、ソフトウェアアプリケーション18は、GUIアプリケーション、オペレーティングシステム、ポータブルマッピングアプリケーション、工学用アプリケーションもしくは芸術用アプリケーションのためのコンピュータ支援設計プログラム、ビデオゲームアプリケーション、または2Dグラフィックスもしくは3Dグラフィックスを使用する別のタイプのソフトウェアアプリケーションである場合がある。30

#### 【 0 0 2 6 】

ソフトウェアアプリケーション18は、グラフィカルユーザインターフェース(GUI)および/またはグラフィックスシーンをGPU12にレンダリングするように命令する1つまたは複数の描画命令を含み得る。たとえば、描画命令は、GPU12によってレンダリングされる1つまたは複数のグラフィックスプリミティブのセットを定義する命令を含み得る。いくつかの例では、描画命令は、GUIに使用される複数のウィンドウ面のすべてまたは一部をまとめて定義する場合がある。追加の例では、描画命令は、アプリケーションによって定義されたモデル空間またはワールド空間内に1つまたは複数のグラフィックスオブジェクトを含むグラフィックスシーンのすべてまたは一部をまとめて定義する場合がある。

#### 【 0 0 2 7 】

ソフトウェアアプリケーション18は、1つまたは複数のグラフィックスプリミティブを表示可能グラフィックス画像にレンダリングするために、グラフィックスAPI20を介して1つまたは複数のコマンドをGPU12に発出するようにGPUドライバ22を起動し得る。たとえば、ソフトウェアアプリケーション18は、グラフィックスAPI20を介して、プリミティブ定義をGPU12に提供するようにGPUドライバ22を起動し得る。いくつかの事例では、プリミティブ定義は、描画プリミティブ、たとえば、三角形、矩形、三角形ファン、三角形ストリップなどのリストの形態でGPU12に提供され得る。プリミティブ定義は、レンダリングされるプリミティブに関連する1つまたは複数の頂点を指定する頂点仕様を含み得る。頂点仕様は、各頂点の位置座標を含み、いくつかの事例では、たとえば、色座標、法線ベクトル、およびテクスチャ座標などの、頂点に関連する他の属性を含み得る。プリミティブ定4050

義は、プリミティブタイプ情報(たとえば、三角形、矩形、三角形ファン、三角形ストリップなど)、スケーリング情報、回転情報などを含む場合もある。ソフトウェアアプリケーション18によってGPUドライバ22に発出された命令に基づいて、GPUドライバ22は、プリミティブをレンダリングするためにGPU12が実行する1つまたは複数の演算を指定する1つまたは複数のコマンドを策定し得る。GPU12がCPU6からコマンドを受け取ると、グラフィックス処理パイプライン24は、コマンドを復号し、コマンドに指定された演算を実行するようにグラフィックス処理パイプライン24を構成する。たとえば、グラフィックス処理パイプライン24内の入力アセンブラー26は、プリミティブデータを読み取り、データをアセンブルし、グラフィックス処理パイプライン24の他のグラフィックスパイプラインステージが使用するためのプリミティブにする場合がある。グラフィックス処理パイプライン24は、指定された演算を実行した後、レンダリングされたデータをディスプレイデバイスに関連するフレームバッファ16に出力する。

10

#### 【0028】

フレームバッファ16は、GPU12用の宛先ピクセルを記憶する。各宛先ピクセルは、一意のスクリーンピクセルロケーションに関連付けられる場合がある。いくつかの例では、フレームバッファ16は、各宛先ピクセルに関する色成分および宛先アルファ値を記憶し得る。たとえば、フレームバッファ16は、各ピクセルの赤、緑、青、アルファ(RGBA)成分を記憶し得るが、ここにおいて、「RGB」成分は色値に相当し、「A」成分は宛先アルファ値に相当する。フレームバッファ16およびシステムメモリ10は別のメモリユニットであるものとして示されているが、他の例では、フレームバッファ16は、システムメモリ10の一部である場合がある。

20

#### 【0029】

いくつかの例では、グラフィックス処理パイプライン24は、頂点シェーダステージ、ハルシェーダステージ、ドメインシェーダステージ、ジオメトリシェーダステージ、およびピクセルシェーダステージのうちの1つまたは複数を含み得る。グラフィックス処理パイプライン24のこれらのステージは、シェーダステージと見なされ得る。これらのシェーダステージは、GPU12内のシェーダユニット46上で実行する1つまたは複数のシェーダプログラムとして実装され得る。シェーダユニット46は、処理構成要素のプログラマブルパイプラインとして構成され得る。いくつかの例では、シェーダユニット46は、「シェーダプロセッサ」または「統合シェーダ」と呼ばれる場合があり、ジオメトリシェーディング演算、頂点シェーディング演算、ピクセルシェーディング演算、またはグラフィックスをレンダリングするための他のシェーディング演算を実行する場合がある。シェーダユニット46は、プロセッサコア48を含む場合があり、プロセッサコア48の各々は、演算をフェッチおよび復号するための1つまたは複数の構成要素、算術計算を実行するための1つまたは複数の算術論理ユニット、1つまたは複数のメモリ、キャッシュ、およびレジスタを含む場合がある。

30

#### 【0030】

GPU12は、グラフィックス処理パイプライン24内の頂点シェーダステージ、ハルシェーダステージ、ドメインシェーダステージ、ジオメトリシェーダステージ、およびピクセルシェーダステージのうちの1つまたは複数を実行するコマンドをシェーダユニット46に送ることによって、頂点シェーディング、ハルシェーディング、ドメインシェーディング、ジオメトリシェーディング、ピクセルシェーディングなどの様々なシェーディング演算を実行するようにシェーダユニット46を指定し得る。いくつかの例では、GPUドライバ22は、1つまたは複数のシェーダプログラムをコンパイルし、コンパイルされたシェーダプログラムをGPU12内に含まれる1つまたは複数のプログラマブルシェーダユニット上にダウンロードするように構成され得る。シェーダプログラムは、たとえば、OpenGL Shading Language(GLSL)、High Level Shading Language(HLSL)、C for Graphics(Cg) shading languageなどの高レベルシェーディング言語で書かれる場合がある。コンパイルされたシェーダプログラムは、GPU12内のシェーダユニット46の演算を制御する1つまたは複数の命令を含み得る。たとえば、シェーダプログラムは、頂点シェーダステージの関数を実行するた

40

50

めにシェーダユニット46によって実行され得る頂点シェーダプログラム、ハルシェーダステージの関数を実行するためにシェーダユニット46によって実行され得るハルシェーダプログラム、ドメインシェーダステージの関数を実行するためにシェーダユニット46によって実行され得るドメインシェーダプログラム、ジオメトリシェーダステージの関数を実行するためにシェーダユニット46によって実行され得るジオメトリシェーダプログラム、および/またはピクセルシェーダの関数を実行するためにシェーダユニット46によって実行され得るピクセルシェーダプログラムを含み得る。頂点シェーダプログラムは、プログラムブル頂点シェーダユニットまたは統合シェーダユニットの実行を制御し、1つまたは複数の頂点単位演算を指定する命令を含む場合がある。

## 【0031】

10

グラフィックスメモリ40は、GPU12の集積回路に物理的に統合されたオンチップストレージまたはメモリである。グラフィックスメモリ40がオンチップであるので、GPU12は、システムバスを介してシステムメモリ10から値を読み取り、またはシステムメモリ10に値を書き込むよりも迅速にグラフィックスメモリ40から値を読み取り、またはグラフィックスメモリ40に値を書き込むことができる場合がある。グラフィックスメモリ40は、シーンプリミティブ(たとえば、三角形)を階層的にグループ化するADS41を記憶し得る。いくつかの例では、ADS41は、システムメモリ10内、シェーダユニット46のシェーダメモリ(図示せず)内、または共有システム/グラフィックスメモリ(図示せず)内に記憶され得る。

## 【0032】

20

本開示の態様によれば、GPU12は、ADS41内に複数のプリミティブを編成し得る。GPU12は、ADS41のノードに関連する1つまたは複数のバウンディングボリュームを、グラフィックスメモリ40に記憶されたオフスクリーンレンダリングターゲットにラスタライズし得る。GPU12は、ADS41の横断を開始するためのADS41内の開始ノードとしてADS41の非ルートノードを決定し得る。GPU12は、レイが交差するプリミティブを決定するために開始ノードから開始してADS41を横断し得る。

## 【0033】

30

図3は、GPU12がレイトレーシングを実行し得る例示的なシーンを示すブロック図である。図3に示すように、シーン50は、プリミティブ52A～52E(以下、「プリミティブ52」)を含む2Dシーンまたは3Dシーンである場合がある。レイトレーシング処理の一部として、GPU12は、2Dレイまたは3Dレイである場合がある、ソース55から発するレイ54に対して、レイ54が交差するシーン50の複数のプリミティブ52内のいくつかのプリミティブを決定し得る。

## 【0034】

30

GPU12は、シーン50を分割し、シーン50の分割された部分を階層的に配置し、シーン50の分割された部分の階層を再帰的に横断することによってレイ54が交差する複数のプリミティブ52内の1つのプリミティブを系統的に決定し得る。図4は、シーン50をバウンディングボリュームに区分することを示すブロック図である。図4に示すように、GPU12は、プリミティブ52をバウンディングボリューム56A～56E(「バウンディングボリューム56」)に概念的に区分し得る。バウンディングボリューム56は、囲まれたプリミティブのすべての点が存在し得る最小領域を有するバウンディングボックスである場合がある、軸平行なバウンディングボックス(AABB)である場合がある。バウンディングボックスは、これらのボックスの辺が座標軸(たとえば、x軸、y軸、およびz軸)に平行であり得るように軸平行であり得る。

40

## 【0035】

バウンディングボリューム56Aは、シーン50のすべてのプリミティブ52に境界をつけるバウンディングボックスである場合がある。バウンディングボリューム56Bおよび56Cは、バウンディングボリューム56Aによって縛られたシーン50の一部分のサブセットに境界をつけるという点で、バウンディングボリューム56Aのサブセットであり得る。バウンディングボリューム56Bは、プリミティブ52Aおよび52Bに境界をつける場合があり、バウンディングボリューム56Cは、プリミティブ52C、52D、および52Eに境界をつける場合がある。

50

バウンディングボリューム56Dおよび56Eは、バウンディングボリューム56Cのサブセットである場合があり、バウンディングボリューム56Bによって縛られたシーン50の一部分のサブセットに境界をつける場合がある。バウンディングボリューム56Dは、プリミティブ52Cおよび52Dに境界をつける場合があり、バウンディングボリューム56Eは、プリミティブ52Eに境界をつける場合がある。

#### 【 0 0 3 6 】

図4に示す例では、GPU12は、プリミティブ52を5つのバウンディングボリューム56に区分し得る。GPU12は、5つよりも多いか、または少ないバウンディングボリューム56を使用し得る。たとえば、GPU12は、プリミティブ52Aおよび52Bにそれぞれ個別に境界をつけるためにバウンディングボリューム56Bのサブセットとして追加のバウンディングボリュームを形成し得る。いくつかの例では、CPU6は、プリミティブ52をバウンディングボリューム56に区分するように構成される場合もある。

#### 【 0 0 3 7 】

図5は、図4に示すバウンディングボリューム56およびプリミティブ52に関連付けられたノードを有する例示的な階層データ構造体を示す。上記で説明したように、シーンのシーンプリミティブは、ADS41内に編成される場合があり、ADS41は、レイプリミティブ交点を決定するために横断される場合がある。GPU12は、グラフィックスメモリ40内にADS41を記憶し得る。図5に示すように、バウンディングボリューム56およびプリミティブ52は、シーン50に関連するバウンディングボリューム階層(BVH: bounding volume hierarchy)ツリー-60などのADS41のノードに関連付けられる場合がある。GPU12またはCPU6は、BVHツリー-60がシーン50のバウンディングボリューム56およびプリミティブ52に関連付けられたノード62A～62E(「ノード62」)を含むようにBVHツリー-60を構成し得る。より具体的には、ルートノード62A、内部ノード62C、ならびにリーフノード62B、62D、および62Eは各々、複数のバウンディングボリューム56のうちの1つのバウンディングボリュームに関連付けられる場合がある。ルートノード62A以外のBVHツリー-60のノードは、BVHツリー-60の非ルートノードと呼ばれる場合がある。たとえば、内部ノード62C、ならびにリーフノード62B、62D、および62Eは、BVHツリー-60の非ルートノードと呼ばれる場合がある。リーフノード62B、62D、および62Eは各々、複数のプリミティブ52のうちの少なくとも1つのプリミティブにリンクされ得る。たとえば、リーフノード62Bは、プリミティブ52Aおよび52Bにリンクされる場合があり、リーフノード62Dは、プリミティブ52Cおよび52Dにリンクされる場合があり、リーフノード62Eは、プリミティブ52Eにリンクされる場合がある。BVHツリー-60は、不平衡なバイナリツリーと見なされる場合があるが、その理由は、階層データ構造体60の各非リーフノードが多くて2つの子ノードを有し、リーフノード62B、62D、および62Eが等しくない深さを有する場合があるからである。

#### 【 0 0 3 8 】

バウンディングボリューム56Aがシーン50のすべてのプリミティブ52に境界をつけるので、GPU12は、ルートノード62Aをバウンディングボリューム56Aに関連付け得る。GPU12は、ルートノード62Aの子ノードであるノード62Bおよび62Cを、それぞれバウンディングボリューム56Bおよび56Cに関連付け得るが、その理由は、バウンディングボリューム56Bおよび56Cが各々、バウンディングボリューム56Aのサブセットであるからである。GPU12は、プリミティブ52Aおよび52Bをノード62Bにリンクさせる場合があるが、その理由は、ノード62Bに関連するバウンディングボリューム56Bがプリミティブ52Aおよび52Bに境界をつけ、バウンディングボリューム56Bのサブセットである追加のバウンディングボリュームが存在しないからである。

#### 【 0 0 3 9 】

GPU12は、バウンディングボリューム56Dおよび56Eを、ノード62Cの子ノードであるノード62Dおよび62Eに関連付け得るが、その理由は、バウンディングボリューム56Dおよび56Eが各々、ノード62Cに関連するバウンディングボリューム56Cのサブセットであるからである。GPU12は、プリミティブ52Cおよび52Dをノード62Dにリンクさせる場合があるが、その理由は、ノード62Dに関連するバウンディングボリューム56Dがプリミティブ52Cおよび52D

10

20

30

40

50

に境界をつけ、バウンディングボリューム56Dのサブセットである追加のバウンディングボリュームが存在しないからである。GPU12は、プリミティブ52Eをノード62Eにリンクさせる場合もあるが、その理由は、ノード62Eに関連するバウンディングボリューム56Eがプリミティブ52Eに境界をつけ、バウンディングボリューム56Eのサブセットである追加のバウンディングボリュームが存在しないからである。

#### 【0040】

GPU12は、BVHツリー-60のノード62に関連するバウンディングボリューム56に対してレイボックス交差テストを実行することによってレイ54に交差する複数のプリミティブ52のうちの1つのプリミティブを決定するようにBVHツリー-60を横断し得る。GPU12は、ルートノード62Aに関連するバウンディングボリューム56Aに対してレイボックス交差テストを実行することによってBVHツリー-60の横断を開始する場合がある。レイ54がバウンディングボリューム56Aに交差するとGPU12が判断した場合、GPU12は、ノード62BまでBVHツリー-60を横断し続ける場合があり、GPU12は、ノード62Bに関連するバウンディングボリューム56Bに対してレイボックス交差テストを実行する場合がある。レイ54がバウンディングボリューム56Bに交差しないとGPU12が判断した場合、GPU12は、ノード62Aおよびさらに下にノード62CまでBVHツリーを再帰的に横断する場合があり、GPU12は、ノード62Cに関連するバウンディングボリューム56Cに対してレイボックス交差テストを実行する場合がある。レイ54がバウンディングボリューム56Cに交差するとGPU12が判断した場合、GPU12は、ノード62DまでBVHツリー-60を横断し続ける場合があり、GPU12は、ノード62Dに関連するバウンディングボリューム56Dに対してレイボックス交差テストを実行する場合がある。レイ54がバウンディングボリューム56Dに交差するとGPUが判断した場合、GPU12は、ノード62Dにリンクされたプリミティブに対してレイプリミティブ交差テストを実行する場合がある。したがって、GPU12は、プリミティブ52Cに対してレイプリミティブ交差テストを実行する場合があり、プリミティブ52Dに対してレイプリミティブ交差テストを実行する場合もある。GPU12は、プリミティブ52Dに対するレイプリミティブ交差テストから、レイ54がプリミティブ52Dに交差すると判断し得る。レイ54がプリミティブ52Dに交差するとGPU12が判断した後、GPU12は、ノード62Cおよびさらに下に62EまでBVHツリー-60を再帰的に横断し続ける場合があり、GPU12は、ノード62Eに関連するバウンディングボリューム56Eに対してレイボックス交差テストを実行する場合がある。GPU12は、レイボックス交差テストに基づいて、レイ54がバウンディングボリューム56Eに交差しないと判断し、それによって、レイ54に関するBVHツリー-60の横断を終了する場合がある。

#### 【0041】

内部ノード62Cまたはリーフノード62B、62D、および62Eのうちの1つなどの非ルートノードから開始してBVHツリー-60を横断することができる場合、GPU12は、ルートノード62AからBVHツリー-60の横断を開始することと比較して、GPU12が実行するレイ交差テストの数を低減し、それによって、レイが交差するプリミティブを決定する効率を増大させる場合がある。GPU12は、レイ54がBVHツリー-60の非ルートノードに関連するバウンディングボリュームに交差すると判断することによってBVHツリー-60の非ルートノードからBVHツリー-60の横断を開始する場合がある。GPU12は、バウンディングボリューム56の少なくとも1つのサブセットを、グラフィックスメモリ40内のオフスクリーンレンダリングターゲットにラスタライズし得る。GPU12は、複数のバウンディングボリューム56をオフスクリーンレンダリングターゲットにラスタライズすることに基づいて、BVHツリー-60の横断を開始するためのBVHツリー-60内の開始ノードとしてBVHツリー-60の非ルートノードを決定し得る。GPU12は、レイ54が交差するプリミティブを決定するために開始ノードから開始してBVHツリー-60を横断し得る。

#### 【0042】

図6は、例示的なオフスクリーンレンダリングターゲットにラスタライズされたバウンディングボリュームの例示的な表示を示すプロック図である。GPU12は、BVHツリーの上位レベルの指定された数に関して、BVHツリーの指定された上位レベルのノードに関連するバウンディングボリュームの表示をグラフィックスメモリ40内のオフスクリーンレンダリ

ングターゲットにラスタライズするために、そのグラフィックスパイプラインのピクセルシェーダおよび/または頂点シェーダを使用する場合がある。たとえば、16レベルBVHツリーを仮定すれば、GPU12は、BVHツリーの上位5~6レベルに関連するバウンディングボリュームの表示をラスタライズし得る。したがって、GPU12は、BVHツリーのレベルのすべてよりも少ないレベルに関連するバウンディングボリュームの表示をラスタライズすることを選択し得る。

#### 【 0 0 4 3 】

図6に示すように、GPU12は、レンダリングターゲット64内のBVHツリー60のノード62B~62Eに関連するバウンディングボリューム56B~56Eの表示66A~66D(「表示66」)をレンダリングし得る。GPU12は、グラフィックスメモリ40内にレンダリングターゲット64を記憶し得る。GPU12は、バウンディングボリューム56B~56Eを投影する場合があり、GPU12のシェーダユニット46は、BVHツリー60の非ルートノード62B~62Eに関連するバウンディングボリューム56B~56Eの表示を、多角形、立方体などの、2次元表示または3次元表示としてラスタライズする場合がある。たとえば、GPU12のハードウェアラスタライザは、バウンディングボリューム56の各々を、レンダリングターゲット64内のピクセル内にスキャンコンバートし得る。一例では、GPU12は、バウンディングボリューム56B~56Eの表示66をフラットシェーディングされた立方体としてラスタライズし得る。GPU12は、表示66の各々の出力色がBVHツリー60内の関連のノードを示すノードインデックスを表し得るように、異なる色でバウンディングボリューム56B~56Eの表示66をラスタライズする場合もある。GPU12は、関連のバウンディングボリュームの形状をマッチングさせるために内部ノードの表示の各々をスケーリングし変換する場合がある。いくつかの例では、GPU12は、シーン50内のバウンディングボリューム56B~56Eよりも比較的低い解像度でバウンディングボリューム56B~56Eの表示66をラスタライズし得る。このように、GPU12は、レイプリミティブ交点を決定する際に、その性能をさらに増大させる場合がある。  
10

#### 【 0 0 4 4 】

GPU12は、投影されたバウンディングボリューム56B~56Eの標準深さテストを実行することによって、また、投影されたバウンディングボリューム56B~56Eの様々な深さを表す表示66のピクセルに様々な出力色を割り当てることによって表示66の出力色を決定し得る。GPU12は、表示66のピクセルに割り当てられた出力色をBVHツリー60のノード62に関連付け得る。このように、GPU12は、ピクセルの出力色をサンプリングすることによってレンダリングターゲット64内のピクセルに関連する、BVHツリー60内のノードを決定し得る。  
30 表示66をレンダリングターゲット64にラスタライズする一環として、GPU12は、レンダリングターゲット64内のピクセルに関して、GPU12がピクセルに交差するレイ54などの1つまたは複数の1次レイをマッピングし得るように、レンダリングターゲット64内のピクセルへの1次レイのマッピングを決定し得る。いくつかの例では、レンダリングターゲット64は、ピクセルと1次レイとの間の1対1マッピングを有し得る。他の例では、ターゲット64をレンダリングするために(対応するバウンディングボリューム56の解像度と比較して)比較的低い解像度で表示66がラスタライズされる場合、ピクセルは、複数の1次レイにマッピングされ得る。GPU12は、表示66Cがレイ54のソースに最も近い表示であること、および表示66Cがレイ54にマッピングされたピクセルを含むことを判定し得る。GPU12は、ピクセルの色を決定するためにレイ54にマッピングされた表示66C内のピクセルをサンプリングし得る。GPU12は、表示66C内のピクセルのサンプリングされた色がノード62Dのノードインデックスに関連付けられると判定する場合があり、それによって、表示66Cがノード62Dに関連付けられると判定する場合がある。したがって、GPU12は、BVHツリー60を横断するための開始ノードとしてノード62Dを設定する場合があり、ノード62DからBVHツリー60の横断を開始する場合がある。  
40

#### 【 0 0 4 5 】

GPU12がノード62Dから開始してBVHツリー60を横断するとき、GPU12は、ノード62Dに関連するバウンディングボリューム56Dに対してレイボックス交差テストを最初に実行し得る。レイ54がバウンディングボリューム56Dに交差するとGPUが判断した場合、GPU12は、  
50

ノード62Dにリンクされたプリミティブ52Cに対してレイプリミティブ交差テストを実行する場合がある。レイ54がプリミティブ52Cに交差しないとGPUが判断した場合、GPU12は、ノード62DまでBVHツリー60を再帰的に横断する場合があり、ノード62Dにリンクされたプリミティブ52Dに対してレイプリミティブ交差テストを実行する場合がある。GPU12は、プリミティブ52Dに対するレイプリミティブ交差テストから、レイ54がプリミティブ52Dに交差すると判断し得る。GPU12は、ノード62CまでBVHツリー60を再帰的に横断し、それによつて、レイ54に関するBVHツリー60の横断を終了する場合がある。わかるように、レイ54に交差するプリミティブを見つけるためのBVHツリー60の横断は、GPU12がルートノードから開始してBVHツリー60を横断するように要求される場合とは対照的に、比較的少ないレイボックス交差テストを実行することを含み得る。

10

#### 【 0 0 4 6 】

いくつかの例では、視野角に応じて、任意の2つのノードのバウンディングボリュームの表示が、スクリーン上に投影されるとき、重複する場合がある。この場合、BVHツリー60の横断は、2つの重複するバウンディングボリュームの最も低い共通の先祖(lower common ancestor)から開始する場合がある。バウンディングボリュームの2つの表示が重複する場合、かつ2つのバウンディングボリュームの重複した領域内でレイが交差する場合、GPU12は、バウンディングボリュームに関連するノードのうちの最も低い共通の先祖ノードを決定する場合があり、2つの重複した表示によって表示されたバウンディングボリュームに関連するノードのうちの最も低い共通の先祖ノードからBVHツリー60の横断を開始する場合がある。たとえば、GPU12は、表示66をレンダリングターゲット64にラスタライズしながら、表示66Cおよび表示66Dがエリア69内で重複すると判定し得る。GPU12は、ノード62Dおよび62Eのうちの最も低い共通の先祖ノードに関連する出力色を決定する場合があり、エリア69(すなわち、充填(fill)エリア69)内のピクセルの出力色を、ノード62Dおよび62Eのうちの最も低い共通の先祖ノードに関連する同じ出力色に設定する場合がある。この例では、GPU12は、ノード62Cがノード62Dおよび62Eのうちの最も低い共通の先祖ノードであると判定する場合があり、エリア69内のピクセルの出力色を、ノード62Cに関連付けられた表示66Bの出力色に設定する場合がある。

20

#### 【 0 0 4 7 】

たとえば、ソース55から発するレイ68がエリア69に交差する場合、GPU12は、ノード62Cから横断を開始することを決定する場合がある。ノード62Cがレンダリングターゲット64内の表示66Bによって表示されるバウンディングボリューム56Cに関連付けられるので、GPU12は、ノード62Cに関連するバウンディングボリューム56Cに対してレイボックス交差テストを実行し得る。レイ68がバウンディングボリューム56Cに交差するとGPU12が判断した場合、GPU12は、ノード62DまでBVHツリー60を横断する場合がある。GPU12は、ノード62Dに関連するバウンディングボリューム56Dに対してレイボックス交差テストを実行する場合がある。レイ54がバウンディングボリューム56Dに交差するとGPUが判断した場合、GPU12は、ノード62Dにリンクされたプリミティブ52Cに対してレイプリミティブ交差テストを実行する場合がある。レイ54がプリミティブ52Cに交差しないとGPUが判断した場合、GPU12は、ノード62DまでBVHツリー60を再帰的に横断する場合があり、ノード62Dにリンクされたプリミティブ52Dに対してレイプリミティブ交差テストを実行する場合がある。GPU12は、プリミティブ52Dに対するレイプリミティブ交差テストから、レイ54がプリミティブ52Dに交差すると判断し得る。しかしながら、GPU12は、ノード62Cおよびさらに下にノード62EまでBVHツリー60を再帰的に横断し続ける場合があり、ノード62Eに関連するバウンディングボリューム56Eに対してレイボックス交差テストを実行する場合がある。GPU12は、レイ54がバウンディングボリューム56Eに交差しないと判断し、それによつて、レイ54に関するBVHツリー60の横断を終了する場合がある。

30

#### 【 0 0 4 8 】

図7は、レイプリミティブ交点を見つけるために例示的な階層ツリー構造体を横断するための開始ノードを決定するための例示的な処理を示すフローチャートである。図7に示すように、本処理は、CPU6またはGPU12などの少なくとも1つのプロセッサによって、階層

40

50

データ構造体内に複数のプリミティブを編成することを含み得る(702)。本処理は、CPU6またはGPU12によって、階層データ構造体の非ルートノードに関連する複数のバウンディングボリュームをオフスクリーンレンダリングターゲットにラスタライズすることをさらに含み得る(704)。本処理は、CPU6またはGPU12によって、また複数のバウンディングボリュームをオフスクリーンレンダリングターゲットにラスタライズすることに基づいて、階層データ構造体の横断を開始するための階層データ構造体内の開始ノードとして階層データ構造体の非ルートノードを決定することをさらに含み得る(706)。本処理は、CPU6またはGPU12によって、レイが交差するプリミティブを決定するために開始ノードから開始して階層データ構造体を横断することをさらに含み得る(708)。

#### 【0049】

10

いくつかの例では、本処理は、GPU12によって、複数のバウンディングボリュームの複数の表示をオフスクリーンレンダリングターゲットにラスタライズすることをさらに含み得る。本処理は、CPU6またはGPU12によって、表示の出力色に少なくとも部分的に基づいて複数のバウンディングボリュームの複数の表示のうちの1つの表示に関連するノードを決定することをさらに含み得る。本処理は、CPU6またはGPU12によって、複数の表示のうちの1つまたは複数内の1つまたは複数のピクセルがレイにマッピングされることを決定することと、CPU6またはGPU12によって、開始ノードとして1つまたは複数のピクセルに関連する階層データ構造体のノードを決定することをさらに含み得る。

#### 【0050】

20

いくつかの例では、本処理は、CPU6またはGPU12によって、複数の表示のうちの2つ以上の表示が重複すると判定することと、GPU12によって、最も低い共通の先祖の表示の出力色を有する2つ以上の表示の重複の領域を充填することと、CPU6またはGPU12によって、開始ノードとして、最も低い共通の先祖の出力色に関連付けられた階層データ構造体のノードを決定することをさらに含み得る。いくつかの例では、本処理は、GPU12によって、複数のバウンディングボリュームの複数の表示を、複数のバウンディングボリュームよりも比較的低い解像度でオフスクリーンレンダリングターゲットにラスタライズすることをさらに含み得る。

#### 【0051】

30

1つまたは複数の例では、説明した機能は、ハードウェア、ソフトウェア、ファームウェア、またはそれらの任意の組合せで実装され得る。ソフトウェアで実現される場合、機能は、1つまたは複数の命令またはコードとして、コンピュータ可読媒体上に記憶されるか、またはコンピュータ可読媒体を介して送信される場合がある。コンピュータ可読媒体は、コンピュータデータ記憶媒体、またはコンピュータプログラムの1つの場所から別の場所への転送を容易にする任意の媒体を含む通信媒体を含み得る。データ記憶媒体は、本開示で説明する技法を実装するための命令、コードおよび/またはデータ構造を取り出すために1つもしくは複数のコンピュータまたは1つもしくは複数のプロセッサによってアクセスされ得る任意の利用可能な媒体であり得る。限定ではなく例として、そのようなコンピュータ可読媒体は、RAM、ROM、EEPROM、CD-ROMもしくは他の光ディスクストレージ、磁気ディスクストレージもしくは他の磁気記憶デバイス、または、命令もしくはデータ構造の形態の所望のプログラムコードを搬送もしくは記憶するために使用することができ、コンピュータによってアクセスすることができる、任意の他の媒体を含み得る。また、任意の接続も厳密にはコンピュータ可読媒体と呼ばれる。たとえば、ソフトウェアが、同軸ケーブル、光ファイバーケーブル、ツイストペア、デジタル加入者線(DSL)、または赤外線、無線、およびマイクロ波などのワイヤレス技術を使用して、ウェブサイト、サーバ、または他のリモートソースから送信される場合、同軸ケーブル、光ファイバーケーブル、ツイストペア、DSL、または赤外線、無線、およびマイクロ波などのワイヤレス技術は、媒体の定義に含まれる。ディスク(disk)およびディスク(disc)は、本明細書で使用するとき、コンパクトディスク(CD)、レザーディスク(登録商標)、光ディスク、デジタル多用途ディスク(DVD)、フロッピーディスク、およびブルーレイディスクを含み、ディスク(disk)は通常、データを磁気的に再生し、ディスク(disc)は、レーザーを用いてデータを光学

40

50

的に再生する。上記の組合せもコンピュータ可読媒体の範囲の中に含まれるべきである。

#### 【0052】

コードは、1つもしくは複数のデジタル信号プロセッサ(DSP)、汎用マイクロプロセッサ、特定用途向け集積回路(ASIC)、フィールドプログラマブル論理アレイ(FPGA)、または他の均等な集積論理回路もしくは個別論理回路などの1つまたは複数のプロセッサによって実行され得る。したがって、本明細書で使用する「プロセッサ」および「処理ユニット」という用語は、前述の構造、または本明細書で説明する技法の実装に適した任意の他の構造のいずれかを指す場合がある。加えて、いくつかの態様では、本明細書で説明する機能は、符号化および復号のために構成された専用のハードウェアモジュールおよび/またはソフトウェアモジュール内で提供されてもよく、または複合コーデックに組み込まれてもよい。また、技法は、1つまたは複数の回路または論理要素において完全に実装され得る。

#### 【0053】

本開示の技法は、ワイヤレスハンドセット、集積回路(IC)またはICのセット(すなわち、チップセット)を含む、多種多様なデバイスまたは装置において実装され得る。開示する技法を実行するように構成されたデバイスの機能的態様を強調するために、様々な構成要素、モジュール、またはユニットについて本開示で説明したが、これらの構成要素、モジュール、またはユニットは、必ずしも異なるハードウェアユニットによる実現を必要とするとは限らない。そうではなくて、上で説明されたように、様々なユニットは、コードックハードウェアユニットの中で組み合わされてよく、または適切なソフトウェアおよび/もしくはファームウェアとともに、前述のような1つもしくは複数のプロセッサを含む、相互動作可能なハードウェアユニットの集合によって提供されてよい。

#### 【0054】

様々な例について説明してきた。これらおよび他の例は以下の特許請求の範囲内に入る。

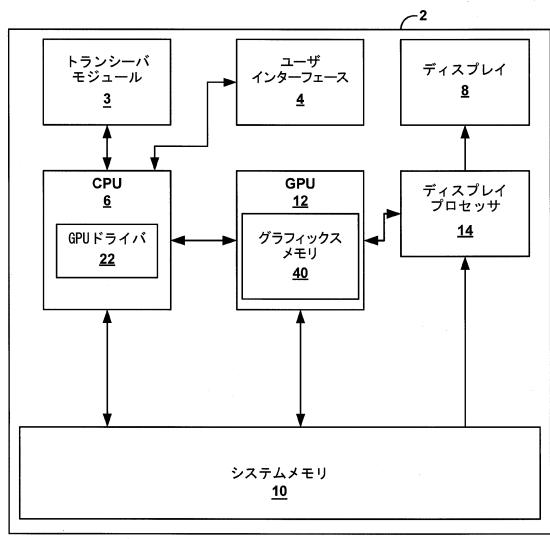
#### 【符号の説明】

#### 【0055】

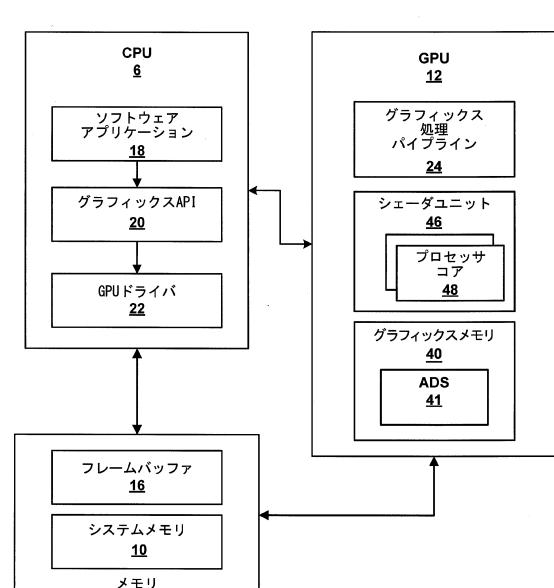
- |                             |  |    |
|-----------------------------|--|----|
| 2 デバイス                      |  |    |
| 3 トランシーバモジュール               |  |    |
| 4 ユーザインターフェース               |  | 30 |
| 6 中央処理ユニット、CPU              |  |    |
| 8 ディスプレイ                    |  |    |
| 10 システムメモリ                  |  |    |
| 12 グラフィカル処理ユニット、GPU         |  |    |
| 14 ディスプレイプロセッサ              |  |    |
| 16 フレームバッファ                 |  |    |
| 18 ソフトウェアアプリケーション           |  |    |
| 20 グラフィックスAPI               |  |    |
| 22 GPUドライバ                  |  |    |
| 24 グラフィックス処理パイプライン          |  | 40 |
| 40 グラフィックスメモリ               |  |    |
| 41 加速データ構造体、ADS             |  |    |
| 46 シェーダユニット                 |  |    |
| 48 プロセッサコア                  |  |    |
| 50 シーン                      |  |    |
| 52 プリミティブ                   |  |    |
| 54 レイ                       |  |    |
| 55 ソース                      |  |    |
| 56 バウンディングボリューム             |  |    |
| 60 バウンディングボリューム階層ツリー、BVHツリー |  | 50 |

- 62A ルートノード
- 62B リーフノード
- 62C 内部ノード
- 62D リーフノード
- 62E リーフノード
- 64 レンダリングターゲット
- 66 表示
- 68 レイ
- 69 エリア

【図1】



【図2】



【図3】

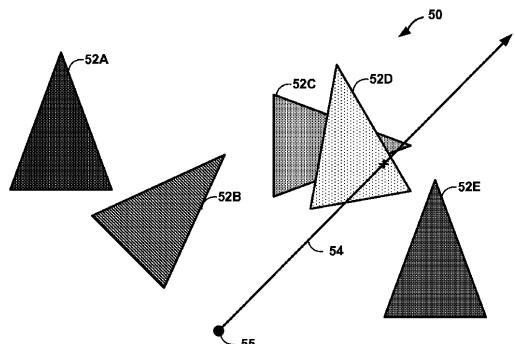


FIG. 3

【図5】

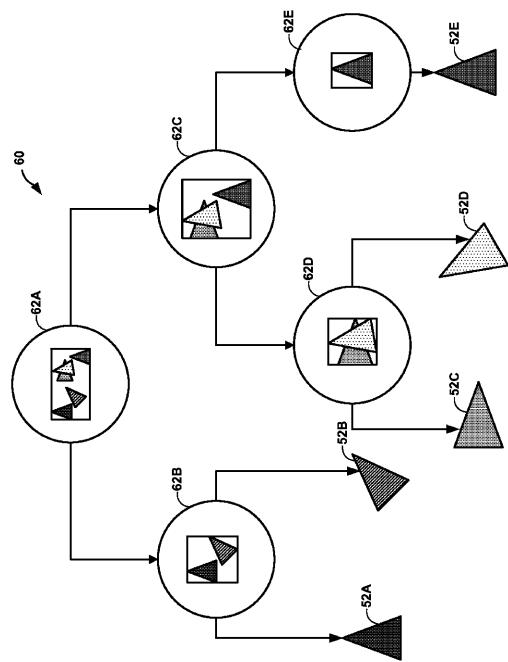


FIG. 5

【図4】

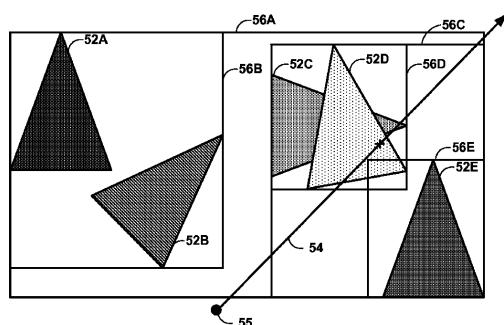


FIG. 4

【図6】

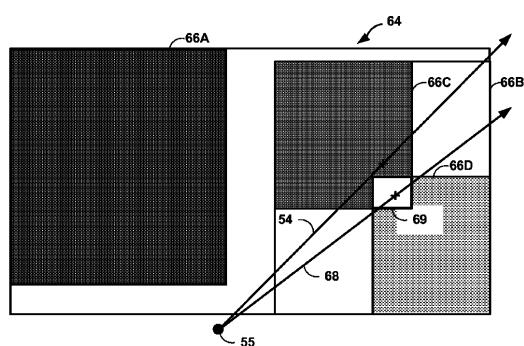
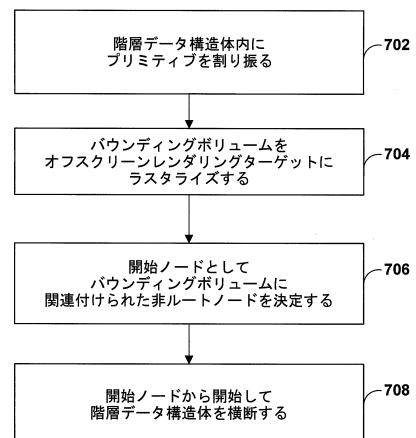


FIG. 6

【図7】



---

フロントページの続き

(72)発明者 ヴィニート・ゴエル

アメリカ合衆国・カリフォルニア・92121-1714・サン・ディエゴ・モアハウス・ドライ  
ヴ・5775

(72)発明者 オウンス・モウリ

アメリカ合衆国・カリフォルニア・92121-1714・サン・ディエゴ・モアハウス・ドライ  
ヴ・5775

審査官 村松 貴士

(56)参考文献 特開2010-134919(JP,A)

特開2008-165760(JP,A)

特開2010-165344(JP,A)

特表2011-515766(JP,A)

米国特許出願公開第2011/0316855(US,A1)

米国特許出願公開第2008/0259075(US,A1)

米国特許出願公開第2010/0289799(US,A1)

(58)調査した分野(Int.Cl., DB名)

G06T 15/00 - 19/20