

(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2015-108703

(P2015-108703A)

(43) 公開日 平成27年6月11日(2015.6.11)

(51) Int.Cl.
G10K 15/00 (2006.01)

F I
G10K 15/00 L

テーマコード (参考)

審査請求 未請求 請求項の数 23 O L 外国語出願 (全 32 頁)

(21) 出願番号 特願2013-250903 (P2013-250903)
(22) 出願日 平成25年12月4日 (2013.12.4)

(71) 出願人 513299605
アカデミア ゴルニーチョーフトニーチャ
アイエム. スタニスラワ スタスイジャ
ダブリュ クラコフイ
Akademia Gomiczo-Hu
tnicza im. Stanisla
wa Staszica w Krako
wie
ポーランド国、 30-059 クラコフ
、 エイエム. ミキウィツカ 30
Al. Mickiewicza 30,
30-059 Krakow, Pol
and
(74) 代理人 100114764
弁理士 小林 正樹

最終頁に続く

(54) 【発明の名称】 ビームトレーシングで可視オブジェクトの近似セットを決定するシステム及び方法

(57) 【要約】

【課題】ビームトレーシングのコンピュータ実行方法を提供する。

【解決手段】ビームトレーシングのコンピュータ実行方法であり、一式のビームは物理的波動現象を表すものであり、その方法は以下のステップを含む：ビーム及び交差する可能性のあるオブジェクトに関する情報を受け取る、ビーム・トライアングル交差テストを実行する、交差テストがネガティブである要素を放棄する、ビームをより小さい部分ビームに分割する、並列プロセスのためにデータを提供する、部分ビームに関して三角形を放棄する、部分ビームをより小さい部分ビームに分割する、より小さい部分ビームを光線に近似させる、各光線に一番近い三角形が存在する場合はその三角形を探す、区切られたより小さい部分ビームを生成する、以前区切られたより小さい部分ビームに選択的に合流させる。

【選択図】 図 1

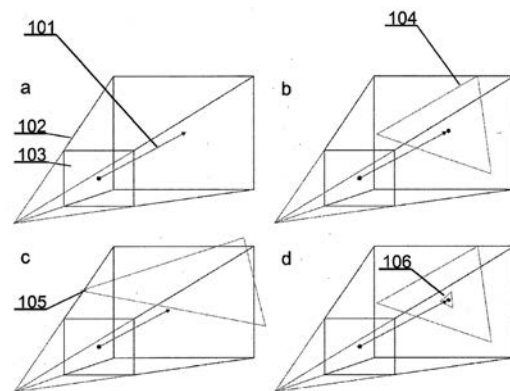


図 1

【特許請求の範囲】

【請求項 1】

ビームトレーシングのコンピュータ実行方法であり、一式のビームは物理的な波動現象を表すものであり、その方法は以下のステップを含む：

ビーム及び交差する可能性のあるオブジェクトに関する情報を受け取る (3 0 1)

ビーム・トライアングル交差テストを実行する (3 0 2)

交差テストがネガティブである要素を放棄する (3 0 3)

ビームをより小さい部分ビームに分割する (3 0 4)

並列プロセスのためにデータを提供する (3 0 5)

部分ビームに関して三角形を放棄する (3 0 6)

部分ビームをより小さい部分ビームに分割する (3 0 7)

より小さい部分ビームを光線に近似させる (3 0 8)

各光線に一番近い三角形が存在する場合はその三角形を探す (3 0 9)

区切られたより小さい部分ビームを生成する (3 1 0)

以前区切られたより小さい部分ビームに選択的に合併させる (3 1 1)。

10

【請求項 2】

交差する可能性のあるオブジェクトに関する情報は、好ましくはシーンツリーなど、オブジェクトにより構成されるシーンの階層あるいは階層的ではない表現として入力される、請求項 1 に記載の方法。

【請求項 3】

部分ビーム及びそれらの部分ビームに関連するそれぞれの三角形の並列プロセスのためのデータを提供すること (3 0 5) を更に含む、請求項 1 に記載の方法。

20

【請求項 4】

部分ビームに関する三角形を放棄すること (3 0 6) は、部分ビームと交差するかに関するテストの結果に基づいて実行され、交差テストがネガティブの場合は実行されない、請求項 1 に記載の方法。

【請求項 5】

区切られたより小さな部分ビームを生成するステップ (3 1 0) は、この特定のより小さな部分ビームに近似する光線の始まりに一番近い位置にあるオブジェクトでビームを区切ることを含む、請求項 1 に記載の方法。

30

【請求項 6】

この特定のより小さい部分ビームに近似する光線の始まりに一番近い位置にあるオブジェクトは、レイ・トライアングル交差テストに基づいて決定される、請求項 5 に記載の方法。

【請求項 7】

レイ・トライアングル交差テストは In g o W a l d 氏によるテストである、請求項 6 に記載の方法。

【請求項 8】

レイ・トライアングル交差テストは、S S E あるいは A V X の指示を使って並行に行われる、請求項 6 に記載の方法。

40

【請求項 9】

より小さい部分ビームの合流 (3 1 1) は、三角形を一致させることで区切られるより小さい部分ビームに行われる、請求項 1 に記載の方法。

【請求項 10】

一致する三角形は、同じ面に位置し同じ材料の同一あるいは二つの三角形あるいは同様の面及び同様の特徴を持つ吸音材の面に位置する三角形である、請求項 1 に記載の方法。

【請求項 11】

一式のビームは音波を表すものである、請求項 1 に記載の方法。

【請求項 12】

一式のビームが物理的波動現象を表すビームトレーシングのシステムであり、そのシス

50

テムは以下を含む：

シーン表現を記憶する階層モデルメモリ（５０７）

レイ・トライアングル及びビームトライアングルタイプの交差テストを行うための専用のシグナルプロセッサ（５０５）

以下のステップを実行するように設定されたバスによりプロセッサ（５０５）に接続されるコントローラ（５０２）；

I/Oコントローラ（５０６）により、ビーム及び交差する可能性のあるオブジェクトに関する情報を受け取る（３０１）

ビーム・トライアングル交差テストを実行する（３０２）

交差テストがネガティブである要素を放棄する（３０３）

ビームをより小さい部分ビームに分割する（３０４）

並列プロセスのためにデータを提供する（３０５）

部分ビームに関して三角形を放棄する（３０６）

部分ビームをより小さい部分ビームに分割する（３０７）

より小さい部分ビームを光線に近似させる（３０８）

各光線に一番近い三角形が存在する場合はその三角形を探す（３０９）

区切られたより小さい部分ビームを生成する（３１０）

以前区切られたより小さい部分ビーム（３１０）に選択的に合併させる（３１１）

【請求項１３】

交差する可能性のあるオブジェクトに関する情報は、好ましくはシーンツリーなど、オブジェクトにより構成されるシーンの階層あるいは階層的ではない表現として入力される、請求項１２に記載のシステム。

【請求項１４】

部分ビーム及びそれらの部分ビームに関連するそれぞれの三角形の並列プロセスのためのデータを提供すること（３０５）を更に含む、請求項１２に記載のシステム。

【請求項１５】

部分ビームに関する三角形を放棄すること（３０６）は、部分ビームと交差するかに関するテストの結果に基づいて実行され、交差テストがネガティブの場合は実行されない、請求項１２に記載のシステム。

【請求項１６】

区切られたより小さな部分ビームを生成するステップ（３１０）は、この特定のより小さな部分ビームに近似する光線の始まりに一番近い位置にあるオブジェクトでビームを区切ることを含む、請求項１２に記載のシステム。

【請求項１７】

この特定のより小さい部分ビームに近似する光線の始まりに一番近い位置にあるオブジェクトは、レイ・トライアングル交差テストに基づいて決定される、請求項１６に記載のシステム。

【請求項１８】

レイ・トライアングル交差テストはIngo Wald氏によるテストである、請求項１７に記載のシステム。

【請求項１９】

レイ・トライアングル交差テストは、SSEあるいはAVXの指示を使って並行に行われる、請求項１７に記載のシステム。

【請求項２０】

より小さい部分ビームの合流（３１１）は、三角形を一致させることで区切られるより小さい部分ビームに行われる、請求項１２に記載のシステム。

【請求項２１】

一致する三角形は、同じ面に位置し同じ材料の同一あるいは二つの三角形あるいは同様の面及び同様の特徴を持つ吸音材をの面に位置する三角形である、請求項１２に記載のシステム。

10

20

30

40

50

【請求項 2 2】

一式のビームは音波を表すものである、請求項 1 2 に記載のシステム。

【請求項 2 3】

コンピュータで実行された場合、請求項 1 に記載のコンピュータ実行方法の全てのステップを実行する、コンピュータで実行可能な指示を記憶するコンピュータで可読の固定記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明の実施形態は、概ねビームトレーシングの分野に関するものであり、特に例えばリアルタイムで音響効果をシミュレートするサウンドトレーシングに関するものである。 10

【背景技術】

【0002】

現在、上記の解決法は、リアルタイムレジームを必要としないため計算に時間がかかるアニメ映画及び/又は漫画(カートゥーン)に主に使われている。同じくビデオゲーム業界においては、通常、音の多様な変化のアクティベーショントリガーを構成するジオメトリ領域が存在する。

【0003】

「適応フラスタム(Adaptive Frustum)」と題する文献において説明される先行技術であるアルゴリズム <http://gamma.cs.unc.edu/SOUND/data/vis2008.pdf>; (AD-Frustum: Adaptive Frustum Tracing for Interactive Sound Propagation; Anish Candak, Christian Lauterbach, Micah Taylor, Zhimin Ren, Dinesh Manocha, Member, IEEE) 及び <http://hal.inria.fr/docs/00/50/99/81/PDF/CG98.pdf> (A Beam Tracing Method with Precise Antialiasing for Polyhedral シーン; Djamchid GHAZANFARPOUR and Jean-Marc HASENFRATZ; Laboratoire MSI - Universite de Limoges) は、多面体を探知するために適応ビームのサブディビジョンを使用する。理論上、それらの方法によりビームを制限するオブジェクト(他のものに遮断されないビームの範囲のオブジェクト)を素早く見つけることが可能になるが、実際、アルゴリズムに使用される構造はリアルタイムの解決法として使うには高額すぎる(それらを使用するには時間がかかる)。特定のケース、ツリー構造、及び限られた正確さ及び遅いビーム合流アルゴリズムを含む高額な交差テストは、現代的なプロセッサを最大限に利用しない。これらの問題によりアルゴリズムを実施するのは難しく活用の拡大の可能性も低くなる。 20 30

【0004】

「リアルタイム音響合成における光線トレーシングの使用(Using ray tracing for real time audio synthesis)」と題するもう一つの先行技術文献米国特許第 8 1 3 9 7 8 0 B 2 号公報は、サウンドエンジンにより、三次元のシーン内で音波をエミュレートすることで聞き手の位置で最終的な音を測定する可能性を開示するものである。サウンドエンジンは、音の起こる位置から光線を出して三次元のシーンを通してその光線をトレースすることで音波をエミュレートしてもよい。光線は、音の変更の要因がある三次元シーン内でオブジェクトと交差しても良い。聞き手が聞いた最終的な音を測定するために、音の変更の要因及び他の要因(例えば、光線が移動した距離、オブジェクトとの交差の角度等)がサウンドイベントに適用されることもある。 40

【0005】

「IEEE TRANSACTIONS ON VISUALIZATION AND 50

COMPUTER GRAPHICS, VOL. 13, NO. 6, NOVEMBER/DECEMBER 2007年」において出版されたChristian Lauterbach及びDinesh Manochaによる「Interactive Sound Rendering in Complex and Dynamic scenes using Frustum Tracing」と題する別の先行技術文献は、動的なソース及びオブジェクトの複雑及び仮想的なシーンにおけるリアルタイムのサウンドレンダリングの方法を開示する。同方法は、インタラクティブな光線トレーシングの有効性と立体表現のトレーシングの正確さを組み合わせたものである。その方法では、凸状の四角錐台が使用され、レイパケットトレーシングを使ってクリッピング及び交差テストを行う。単純で能率的な方式を使用して二次的な角錐台を計算し階層トラバースを行う。しかしながら、アルゴリズムはリアルタイムのシミュレーション及びハイエンドなPCでレンダリングを行えない。この方法は、とても複雑であるため現在のハードウェアの解決法でのゲームにおいてリアルタイムで実行するには不可能である。

10

【先行技術文献】

【特許文献】

【0006】

【特許文献1】米国特許第8139780B2号公報

【非特許文献】

【0007】

【非特許文献1】AD - Frustum: Adaptive Frustum Tracing for Interactive Sound Propagation; Anish Chandak, Christian Lauterbach, Micah Taylor, Zhimin Ren, Dinesh Manocha, Member, IEEE

20

【非特許文献2】A Beam Tracing Method with Precise Antialiasing for Polyhedral Scenes; Djamchid GHAZANFARPOUR and Jean-Marc HASENFRAZ; Laboratoire MSI - Université de Limoges

【非特許文献3】IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, VOL. 13, NO. 6, NOVEMBER/DECEMBER 2007において出版された「Interactive Sound Rendering in Complex and Dynamic scenes using Frustum Tracing」 by Christian Lauterbach, Anish Chandak, and Dinesh Manocha

30

【発明の概要】

【発明が解決しようとする課題】

【0008】

本発明の目的は、リソースを計算することに関してより能率的であり、必要な結果を得る時間を短くすることでリアルタイムのサウンドトレーサーを作成することを可能にする、ビームトレーシングで可視性を決定するコンピュータ実行方法を提供することである。

40

【課題を解決するための手段】

【0009】

本発明の対象は、ビームトレーシングのコンピュータ実行方法であり、一連のビームは物理的な波動現象を表すものであり、その方法は以下のステップを含む、

ビーム及び交差する可能性のあるオブジェクトに関する情報を受け取る、

ビーム・トライアングル交差テストを実行する、

交差テストがネガティブである要素を放棄する、

ビームをより小さい部分ビームに分割する、

50

並列プロセスのためにデータを提供する、
部分ビームに関して三角形を放棄する、
部分ビームをより小さい部分ビームに分割する、
より小さい部分ビームを光線に近似させる、
各光線に一番近い三角形が存在する場合はその三角形を探す、
区切られたより小さい部分ビームを生成する、
以前区切られたより小さい部分ビームに選択的に合併させる。

【 0 0 1 0 】

本発明の他の対象は、一連のビームが物理的波動現象を表すビームトレーシングのシステムであり、そのシステムは以下を含む、

10

シーン表現を記憶する階層モデルメモリ、

レイ・トライアングル及びビームトライアングルタイプの交差テストを行うための専用のシグナルプロセッサ、

以下のステップを実行するように設定されたバスによりプロセッサに接続されたコントローラ、

I/Oコントローラを介して、ビーム及び交差する可能性のあるオブジェクトに関する情報を受け取る、

ビーム・トライアングル交差テストを実行する、

交差テストがネガティブである要素を放棄する、

ビームをより小さい部分ビームに分割する、

20

並列プロセスのためにデータを提供する、

部分ビームに関して三角形を放棄する、

部分ビームをより小さい部分ビームに分割する、

より小さい部分ビームを光線に近似させる、

各光線に一番近い三角形が存在する場合はその三角形を探す、

区切られたより小さい部分ビームを生成する、

以前区切られたより小さい部分ビームに選択的に合併させる。

【 0 0 1 1 】

本発明のもう一つの目的は、コンピュータで実行された場合、本発明に係るコンピュータ実行方法の全てのステップを実行する、コンピュータで実行可能な指示を記憶するコンピュータで読みとり可能な固定記憶媒体を提供することである。

30

【 0 0 1 2 】

好ましくは、交差する可能性のあるオブジェクトに関する情報は、好ましくはシーンツリーなど、オブジェクトにより構成されるシーンの階層あるいは階層的ではない表現として入力される。

【 0 0 1 3 】

好ましくは、同方法は、更に部分ビーム及びそれらの部分ビームに関連するそれぞれの三角形の並列プロセスのためのデータを提供する。

【 0 0 1 4 】

好ましくは、部分ビームに関する三角形を放棄することは、部分ビームと交差するかに関するテストの結果に基づいて実行され、交差テストがネガティブの場合は実行されない。

40

【 0 0 1 5 】

好ましくは、区切られたより小さな部分ビームを生成するステップは、この特定のより小さな部分ビームに近似する光線の始まりに一番近い位置にあるオブジェクトでビームを区切ることを含む。

【 0 0 1 6 】

好ましくは、この特定のより小さい部分ビームに近似する光線の始まりに一番近い位置にあるオブジェクトは、レイ・トライアングル交差テストに基づいて決定される。

【 0 0 1 7 】

50

好ましくは、レイ・トライアングル交差テストは、Ingo Wald氏によるテストである。

【0018】

好ましくは、レイ・トライアングル交差テストは、SSEあるいはAVXの指示を使って並行に行われる。

【0019】

好ましくは、より小さい部分ビームの合流は、三角形を一致させることで区切られるより小さい部分ビームに行われる。

【0020】

好ましくは、一致する三角形は、同じ面に位置し同じ材料の同一あるいは二つの三角形あるいは類似又は同様の特性を持つ吸音材を有する面に位置する三角形である。

10

【0021】

好ましくは、一連のビームは音波を表すものである。

【図面の簡単な説明】

【0022】

本発明の対象は、以下の図面の例示的な実施形態において説明される。

【図1】図1は、レイ・トライアングル方法の使用法を表す4つの例を示す図である。

【図2】図2は、ビームの予め決められた数のセクションへの分割と、それぞれのセクションの別の光線との近似を表す図である。

【図3】図3は、適応レイ・フラスタムのプロセスを示す図である。

20

【図4A】図4Aは、例示的なシーンにおける本発明に係る方法を示す図である。

【図4B】図4Bは、例示的なシーンにおける本発明に係る方法を示す図である。

【図4C】図4Cは、例示的なシーンにおける本発明に係る方法を示す図である。

【図4D】図4Dは、例示的なシーンにおける本発明に係る方法を示す図である。

【図4E】図4Eは、例示的なシーンにおける本発明に係る方法を示す図である。

【図4F】図4Fは、例示的なシーンにおける本発明に係る方法を示す図である。

【図4G】図4Gは、例示的なシーンにおける本発明に係る方法を示す図である。

【図4H】図4Hは、例示的なシーンにおける本発明に係る方法を示す図である。

【図4I】図4Iは、例示的なシーンにおける本発明に係る方法を示す図である。

【図5】図5は、本発明に係るシステムのブロック図である。

30

【発明を実施するための形態】

【0023】

表記と名称

【0024】

以下の詳細な説明の幾つかの部分は、データ処理手順、ステップあるいはコンピュータメモリ上で実行可能なデータビットの操作の他の象徴として示されている。従って、コンピュータはそれらの論理的ステップを実行するため物理量の物理的な操作が必要になる。

【0025】

通常、それらの物理量は、コンピュータシステムで保存、転送、組み合わせ、比較及び操作することが可能な電気あるいは磁気信号という形をとる。一般的な使用法の理由で、これらの信号は、ビット、パケット、メッセージ、値、要素、記号、文字、用語、数字等と称される。

40

【0026】

また、それら及び同様の用語の全ては、適当な物理量と関連付けられ、それらの物理量に適用される便利な呼び名にすぎない。「処理」、「作成」、「転送」、「実行」、「測定」、「検出」、「取得」、「選択」、「計算」、「生成」、あるいは同様の用語は、コンピュータのレジスタ及びメモリ内に物理（電子的）量として表されるデータを操作し、同様にメモリ、レジスタあるいは他の情報記憶装置内などに物理量として表される他のデータに変換するコンピュータシステムの処置およびプロセスのことを示す。

【0027】

50

ここにビームは多角形の底面を持つ三次元のオブジェクト（しばしば面に位置する）であり、定められた範囲及び底面の頂点から生じるエッジベクトルを有する（それらの全てのベクトルは、半面に向けられているはずである）。

【0028】

エッジベクトルの延長が特定の点で交差する場合、その点は頂点と呼ばれる。例えば、四角形の底面を持つビームの場合、4つのうち2つのベクトルが一つの点で交差し、他の2つが2つ目の点で交差することがある。そのような場合、ビームは実質的にはエッジにおいて始まる。

【0029】

最適な底面は三角形及び四角形であるが、本方法は如何なる多角形にも使用できる。四角形の底面を持つビームはしばしば角錐台と呼ばれる。

【0030】

本発明は、適応レイフラスタムと呼ばれる新たな方法であり、適応フラスタムアルゴリズムを拡張するための反復的な試み及び拡張が性能にどう影響を与えるかについてテストを行った後考案された。

【0031】

この方法は、少ない数の複雑なテストをより多い数のより単純なテストに代えることが有利であるとする見解に基づく。実際、適応フラスタムアルゴリズムにおいてビーム細分を管理するのに使用される四分木構造の大部分は少ない数の三角形を足すと最大レベルの細分に達する。多くの場合、適応レイフラスタムにおいて使用された単純なテストの数は、適応フラスタム(Adaptive Ray-Frustum)において使用された複雑なテストの数を下回ることができることが判明した。

【0032】

本発明に係る方法は、十分に小さいビームは光線で近似することが可能であるという仮定に基づく。例えば、上記のように音のシミュレーションに使用されたシーンにおけるオブジェクトは、シーンにおける相当するグラフィックオブジェクトに比べてずっと単純である。それは、小さなサイズのビームが光線で近似された場合、エラーの可能性が低いことを意味する。また、ビームのほとんどは、完全に遮断されているか、まったく遮断されていない。

【0033】

ビームとの交差に関してテスト済みの三角形が、ビームの全ての部分について交差テストを実行する必要があるほど十分に大きいと仮定する場合（ビームのk回目のクリッピングの後、部分の数は 4^k ）、 $4^{k+1} - 1$ 回のテストが行われる。

【0034】

本発明は、 4^k までのより単純なテスト及び $4^p - 1$ までの複雑なビーム・トライアングルテストを導入する。pおよびkのパラメータの適当な選択により処理速度をかなり上げることができるため実行時間が大幅に短くなる。

【0035】

ビームを光線で近似

【0036】

ビームを幾つかに分ける細分化は、三角形がビームの適当に小さい部分と交差するか否かのバイナリ判定を可能にする。交差している場合、ビームが交差する三角形の一部はビームを完全に遮断すると推測される。これは、とても小さいビームだけに当てはまる。

【0037】

そのような単純化された方法が利用されることが決定したため、ビーム及び三角形間で高額な交差テストを利用する必要はないと見てよい。この高額なテストは、ビームの中心を通った光線が三角形と交差したか否かの単純な確認法と置き換えても良い。

【0038】

その一方で、より大きいビームが光線で近似されるとエラーが多発し適切な三角形が拒絶される場合がある。以下は解決法の個々の要素の説明である。

10

20

30

40

50

【 0 0 3 9 】

大きいビームと小さいビームの違いは、所定のシーンを考慮した適切な分割にある。多数の三角形を感知するのを可能にする分割からは小さいビームが得られ、多数のオブジェクトの感知を可能にしない分割では大きいビームが得られる。これは定量的アプローチである。例えば、音に一点の源がある場合、その源の六角形の異なる側面から起因する6つのビームが放射される。

【 0 0 4 0 】

ビームが部分ビームに分割されない場合、それぞれに1つの側面が感知され、音は多くても6つの異なる側面から反射される。各ビームが一回分割される場合、24の部分ビームが発生し、それは音が反射するオブジェクトの数は多くて24であることを意味する。2回の分割の場合、最大96個のオブジェクトが感知され、3回の分割の場合には最大384個のオブジェクトがあり、4回の分割の場合には最大1536個のオブジェクトが感知される。

10

【 0 0 4 1 】

従って、大きいビームは、シーンに関して十分に分割されていなく多数のオブジェクト(三角形)を感知することができないビームを意味する。この問題は、ビームが分割されていないうえ光線が三角形を外したため大きい三角形が全く感知されなかった図1Cに部分的に示されている。図1Dに示されるように、小さい三角形が大きい三角形を遮断し、ビームの分割が不十分であった場合(大きいビーム)、より大きな三角形は感知されない。

20

【 0 0 4 2 】

以上の記述定義は、方程式により定量的に定義されてもよい。即ち、この方法は、 h は所定のオブジェクトへの距離であり p は分割の数である場合、 $(2^{(p-1)})/h$ より小さいオブジェクトは感知しない。例えば、分割が4回行われて、オブジェクトが10メートル離れている場合、同方法による解答は1.25メートルであり、この数字よりも大きいオブジェクトしか確実に感知されない。より小さいオブジェクトが感知される可能性はある程度だけあるが、確実性はない。

【 0 0 4 3 】

適応フラスタムにおけるレイ・トライアングル交差テストの使用

【 0 0 4 4 】

適応フラスタムアルゴリズムの性能テストによると、同テストの計算的に最も高額な部分は交差テストである。そのテストは頻繁に実行され、毎回大量の計算が必要なだけでなく所定のサブビームに関する情報をメモリにロードすることを必要とする。

30

【 0 0 4 5 】

交差アルゴリズムをスピードアップさせる望ましい方法は、上記のテスト実行の必要性をなくすことである。しかしながら、全ての三角形について可能ではない。交差テストを簡素化することでビームの処理を大幅にスピードアップさせることができる。更に改善するには、「ビーム・トライアングル」交差テストを「レイ・トライアングル」テストに置き換える。

【 0 0 4 6 】

残念ながら、ビームを光線で近似することは、大きいビームには不正確である。この場合、場合によっては制限されるため、ビームの中心を通る光線と交差するオブジェクトだけが考慮される。この方法は、大きいビームには無効である。そのような簡素化テストを適切なレベルのビーム細分化と合わせて大きいビームに使用するとアルゴリズムの正確さは大幅に下がる。

40

【 0 0 4 7 】

図1は、レイ・トライアングル方法を使用する4つの例を示す。例(a)は、ビームのスタート面103から始まるビーム102を近似する光線101を示す。

【 0 0 4 8 】

例(b)は、三角形104と交差する光線を示す。三角形104は、そのためビームと

50

交差すると分類される。

【0049】

例(c)は、ビームと交差する大きい三角形105を示すが、ビームを近似する光線101は三角形105と交差しない。その結果、三角形105は、ビームと交差しないと誤って分類される。

【0050】

例(d)は、ビーム102と交差する2つの三角形104及び106を示す。近似する光線101は、両方の三角形と交差し、小さいほうの三角形106は、光線の始まりにより近い。より小さい三角形106は、ビーム102の全ての面をカバーすると分類されるが、大きい三角形104のほうがよい対象である。

10

【0051】

光線のグリッドを利用するアルゴリズム

【0052】

大きいビームを光線で近似する問題を解決するには、図2に示されるようにビームを予め決められた数の区分に均一に分割し、それらの各区分を別々の光線と近似する。

【0053】

図2の例(a)は、ビームが4つの光線と近似される可能性を示す。この方法により4倍の正確さが可能となる。分割がない場合、ビームと交差する三角形のうち1つの三角形が探知される可能性を意味する。4つの部分ビームに分割することで4つの三角形の探知が可能になる。更に分割することで正確さが向上するが計算力も増大する。

20

【0054】

4つの光線のテストはSSE(Streaming SIMD Extensions)指示を使用することで効果的に早められる。例(b)は、ビームの三回の分割に相当する、16の光線とのビームの近似を示す。

【0055】

また、AVX(Advanced Vector Extensions)の指示を使用することで、2回のテストで所定の三角形で全ての光線の交差テストを実行することが可能になり(ベクトル指示なしでの16回に代わる)、更にスピードアップされる(使用された指示及びメモリへのアクセス数によって3から4倍の範囲)。

【0056】

SSEを使用することで4回で1つの三角形と16の光線の交差テストを行うことを可能になり、4倍まで速度が上がる。AVXがSSEと主に異なるのは、AVXはデータ処理中に2倍の計算を実行することである。

30

【0057】

光線のグリッドを作成した後、光線の始まりから三角形の面の方向ベクトルの距離が最短である三角形が各光線について見い出されなければならない。

【0058】

それは処理されるビームの部分を制限する三角形であるとみなされる。区切るオブジェクトからの反射により作成されるビームの数を減少するためには、一致する三角形により制限されるビームの部分を合流する追加的ステップが導入される(一致する三角形は、同じ三角形、あるいは同じ面に位置し同じ材料の2つの異なる三角形、および同様の垂直及び面接触、及び同様の特性の音響材料を持つ面に位置する三角形である)。

40

【0059】

このアプローチにおいて、制限する三角形を見出すステップの間、アルゴリズムの適合性は処理の効果的に最終段階に押される。

【0060】

上記の解決法は適応フラスタムアルゴリズムにおいて提案されるものよりもかなり単純である。上記の解決法は、ツリー構造を必要とせず、保持されたビームの部分の半分までを除去し、ビーム部分の説明に必要なデータのサイズを大幅に減らす。

【0061】

50

実際、（全ての三角形をスキャンして光線との交差テストを実行することで）各光線に一番近い三角形が見つかった場合、アルゴリズムのその部分におけるメモリ使用量を大幅に減らすことが可能である。

【0062】

ここに記載のアプローチの問題は、テストされたレイ・トライアングルの交差点が多数あることである。ビームと交差する可能性のある三角形の数は多いが、これらの三角形の多くは最終的にはビームを制限しないことがわかる。それでも、小さい値の最大のビーム細分化には提示されるこの方法が適応フラスタムに基づくアプローチよりも効率的である。

【0063】

ビーム・トライアングル及びレイ・トライアングルを組み合わせた解決法

【0064】

ビームに直接作用するアルゴリズム及び光線だけに作用するアルゴリズムには重大な欠点がある。一つ目のアルゴリズムは、コストが高いテストの条件によりされるが、階層的処理が早い三角形の排除を可能にする。二つ目のアルゴリズムでは一つの三角形をより早く処理することができるが、確認された三角形の数が多すぎる。

【0065】

この問題の解決法は、両方のアルゴリズムの利点を組み合わせることである。「ビーム・トライアングル」交差テストは大きいビームに適するため、ビームと確実に交差しない三角形を粗く除外する速く不正確なテストを効率をよくするため使用しても良い。ビーム交差テストにより減らされた所定のシーンにおいて処理される三角形の数は、単純な「レイ・トライアングル」交差テストの数を大きく減らす。これにより性能が良くなる。

【0066】

上記の方法を改良するもう一つの方法は、最大数の三角形を破棄する「ビーム・トライアングル」テストの応用でビームを p 倍分割する方法である。このステップを実行した後、所定のビームの区分が $k - p$ 倍分割され、得られた区分は光線と近似される。

【0067】

実用的なテストは最良の結果を示す p の値は、適用された「レイ・トライアングル」交差テスト及び実行するにあたって使用されたベクトル指示によって $k / 4$ から $k / 2$ の間であることを示す。 $k / 4$ から $k / 2$ の間の値の選択はシーンによるため、所定のシーンのテストを行うのが最も良い。デフォルトの値が適用されてもよい。

【0068】

Ingo Wald テスト

【0069】

このテストは光線の三角形の面との交差点の重心座標を計算するための事前に計算された値の使用に基づくものである。このアルゴリズムは、一つの三角形との交差があるかに関していくつかの光線が同時にテストされる、ベクトル指示を使った実施によく適している。

【0070】

図4は、部分ビームが特定の数の光線と近似することを示す。その後、所定の部分ビームと交差する各三角形には、それらの部分ビームと近似する光線との交差が実行される。この時点でIngo Waldテストなどの交差テストが必要となる。テストは、Moller-Trumboreに定義される座標(Plucker coordinates)テストあるいは以下の他の如何なる交差テストでもよい：

- Moller Tomas i Trumbore Ben. Fast, minimum storage ray-triangle intersection. Journal of Graphics Tools 2(1). 1997, s. 21-28;

- Shevtsov Maxim, Soupikov Alexei i Kapus Alexander; Ray-Triangle Intersection Al

10

20

30

40

50

gorithm for Modern CPU Architectures. Proceedings of GraphiCon'2007 conference, 2007;

- Wald Ingo; realtime Ray Tracing and Interactive Global Illumination. PhD Thesis. Saarbrücken: Max-Planck-Institut für Informatik, 2004.

【0071】

アルゴリズムの実施には3つのバージョンが考えられる。

ベクトル指示を使わない実施、

SS E指示を使う実施、これは1つの三角形に対し、一度に4つの光線の交差テストを行うことが可能である、

AV X指示を使う実施、これは1つの三角形に対し、一度に8つの光線の交差テストを行うことが可能である。

【0072】

また、1つの光線での複数の三角形の交差、及び1つの三角形での1つの光線の交差の促進が実施される可能性もある。AV X及び/又はSS Eの使用、及び一度における複数の光線のテストが通常のPCにおいて最も効率的であると証明されたが、他のバージョンが他のプラットフォームではより効率的、あるいは特定のハードウェアにおいて実施しやすい可能性がある。

【0073】

適応レイフラスタムアルゴリズム

【0074】

適応レイフラスタムアルゴリズムは、前の段階で作られた「レイ・トライアングル」テストの数を減らすために2段階の三角形処理を使用する。同処理の最初の段階においてビームあるいはその一部と確実に交差しない三角形を排除する。また、ビームと交差する近似する一連の三角形を提供するためにシーン・ツリーが使用される。それらのオブジェクトは、階層あるいは非階層的構造と定義され、シーンの空間分割を提供することが可能である。

【0075】

アルゴリズムのこの部分の分割の最大のデプスはpに設定されている。ビームと交差している可能性のある三角形のリストから、「ビーム・トライアングル」テストが肯定的な結果を返すものが選択される。次に、そのリストはビームの4つの部分のそれぞれで同様に処理される。そして、pステップの後、結果として生じるビームを制限する可能性のある三角形のリストはアルゴリズムの2つ目の部分に渡される。

【0076】

適応レイフラスタムアルゴリズムの2段階目の処理において、ビームは区分にk-pフォール度に分割され、ビームの各区分は光線に近似される。最初の段階で得られたそれぞれの三角形にはビームのその部分を近似する全ての光線との交差テストが実行される。

【0077】

それぞれの光線の状態はその光線と交差する三角形及び交差点への距離により説明される。テストされた三角形が、光線の現在の状態よりで記録されたものよりも近く、交差テストが肯定的である場合、光線の状態は更新される。

【0078】

アルゴリズムの最後のステップは、アルゴリズムに記憶されるビームの数を減らすために一致するビームの合流を試みることである。

【0079】

上記の方法は疑似コードとして以下の方法で公式化される：

【0080】

「適応レイフラスタム方法」

10

20

30

40

50

【数1】

```
Adaptive Ray-Frustum(beam, maximum_beam_division,  
                      pre_processing_depth, scene_tree)  
{  
1  Triangles = find_triangles(scene_tree)  
2  If Triangles.count > 0:  
    {  
4   grid_of_rays = Create_Grid_Of_Rays(beam) 10  
5   Process_The_Beam(beam, maximum_beam_division,  
                     pre_processing_depth, Triangles,  
                     grid_of_rays)  
6   Merge_partial_beams(grid_of_rays, maximum_beam_division)  
    }  
}
```

数1

10

20

【0081】

「ビーム処理方法」

【数 2】

```
Process_The_Beam(beam, maximum_beam_division,  
                 pre_processing_depth, Triangles_To_Process,  
                 grid_of_rays)  
{  
1  For each of partial beams:  
   {  
2   Triangles = Triangles_Intersecting_With_The_Beam(  
                   beam_part, Triangles_To_Process) 10  
3   If pre_processing_depth > 0:  
     {  
4     Process_Beam(partial_beam, maximum_beam_division - 1,  
                   pre_processing_depth - 1, Triangles, grid_of_rays)  
     }  
5   Else: 20  
     {  
6     Process_Ray_Set(partial_beam,  
                       maximum_beam_division - 1,  
                       Triangles, grid_of_rays)  
     }  
   }  
}
```

数2

30

【0082】

「一連の光線を処理する方法」

【数3】

```

Process_Beams_Set(partial_beam,
                  maximum_beam_division - 1,
                  Triangles, grid_of_rays)
{
1  Rays_Count = maximum_beam_division * 2
2  For Each Ray Of The partial_beam:
   {
3   Triangle = Closest_Triangle_Intersecting_With_The_Ray(
                   Triangles)
4   Store the Ray and the Triangle in the grid_of_rays
   }
}

```

10

数3

20

【0083】

一番近い三角形を見出しプロセスは、オブジェクトのリストをスキャンし、光線と交差し、光線の源から交差点の距離が最短であるものを選択するほど単純でもよい（その時点でレイ・トライアングル交差テストが適用される）。

【0084】

代わりに、ビームのソース面から1つの頂点の距離により三角形を分類してもよい（時々、まだスキャンされていない三角形の最短距離よりも近い一番近い三角形が見つかった場合、処理は止められてもよく、一番近い三角形の探知を早く完了する）。

【0085】

ビームを合流するアルゴリズム

30

【0086】

ビームを幾つかの部分に均一に分割することで一致する部分の合流をかなり単純化することができる。一致する部分は、隣接し、同一の三角形あるいは同じ物質で同じ面に位置する三角形により制限される部分である。スイープアルゴリズムは、部分ビームのグリッドを検索し、一致する三角形により制限される領域を探すために使用される（一致する部分ビームは一致する三角形により制限される）。そして、それらの領域は、ビームトレーシングのためのアルゴリズムで処理されるビームの数を減らすために合流される。

【0087】

ビーム合流は、ビームは k 回分割されると 4^k 部分になり、反射されたビームが 4^k 生成されることを意味し、有利である。反射されたそれぞれのビームは、 4^k 部分に k 回分割され、最大 $4^k * 4^k = 4^{2k}$ のビームが得られる。例えば、反射だけが可能なサウンドトレーシングエンジンにおいて源が6つのビームを放射すると仮定すると、アルゴリズムにおいて生成されるビームの最終の最大数は、 $6 * (1 + 4^k + 4^{2k} + 4^{4k})$ である。 $k = 0$ の場合、ビームの数は最大24、 $k = 1$ の場合、ビームの最大数は1662、 $k = 2$ の場合、ビームの最大数は394854、そして $k = 3$ の場合、ビームの最大数は100688262である。それぞれのビームが64部分に分割されると1億以上になり容認できない数となる。

40

【0088】

ビームの数を制限することは、これらのビームが更に処理されるため、一度目及び2度目の反射においては特に有利である。例えば、例示的テストモデルにおいて通常のアルゴ

50

リズムにおいて1.35秒で1425383のビームが生成されたのに対し、ビームを合流させる新しいアルゴリズムを使用した後は、その数は4559に減り、実行時間は0.04秒だった。

【0089】

図3は、適応レイフラスタムのプロセスを示す。ステップ301において音波ビーム及び交差し得るオブジェクトに関する情報が受け取られる(シーン・ツリーを使用)。疑似コードとして以前示された適応レイフラスタム方法において、このステップは1行目に相当する。そして、ステップ302において、「ビーム・トライアングル」交差テストが実行される。以前に疑似コードにおいて示されたビーム処理方法において、このステップは2行目に相当する。交差テストが否定的であるこれらの要素は、ステップ303破棄される。

10

【0090】

同方法の最初の大部分が完了した後、ステップ304においてビームをより小さい部分ビームに分割する手順に進む。疑似コードとして以前示されたビーム処理方法において、このステップは3-4行目に相当する。続いて、ステップ305において、例えば、2つの部分ビーム及び部分ビームに関して更に処理されるそれぞれの三角形など、並列処理のためのデータが提供される。以前、疑似コードとして示されたビーム処理方法において、このステップは3-4行目に相当する。

【0091】

次に、ステップ306において、部分ビームに対して三角形の破棄が実行される。シーン要素は、部分ビームと交差するか確認される。交差テストが否定的であるものは破棄される。以前、疑似コードにおいて示されたビーム処理方法において、このステップは2行目に相当する。

20

【0092】

次に、ステップ307において、部分ビームは、より小さい部分ビームに分割される。以前、疑似コードとして示された一連の光線の処理方法において、このステップは一行目に相当する。その後、ステップ308において、ステップ307の部分ビームは光線で近似される。以前、疑似コードにおいて示された一連の光線の処理方法において、このステップは2行目に相当する。次に、ステップ309において、存在する場合、それぞれの光線に一番近い三角形が探知される。以前、疑似コードにおいて示された一連の光線の処理方法において、このステップは3行目に相当する。その後、ステップ310において、区切られた部分ビームが生成される。前のステップにおけるそれぞれの光線は、一つの部分t系ビームに相当する。その部分ビームは、ステップ307のこの特定の部分ビームを近似する光線の始めに一番近く位置するオブジェクトにより区切られる。以前疑似コードにおいて示された一連の光線を処理する方法において、このステップは、4行目に相当する。

30

【0093】

方法311の最終ステップでは、以前区切られた部分ビームに選択的に合流を行う。このステップは、処理の結果を制限するために実行される。疑似コードにおいて以前示された適応レイフラスタムにおいて、このステップは6行目に相当する。

40

【0094】

図4Aから4Iは、例示的シーンにおいて本発明に係る方法を示す。シーンは、一つのビームを構成する図4Aに示すように、1つのビーム及び5つの障害物P1からP5を構成する。図4Bは、ステップ302及び303を実行した結果を示すものであり、P5の全体とP1オブジェクトの一部が、初期メインビームと交差しないとして破棄されている。図4Cは、例としてビームが2つの部分ビームに分割された方法のステップ304の結果を示す。図4Dは、同方法のステップ305及び306の結果を示す。左の部分ビーム及び右の部分ビームの並列処理のデータが提供され、以前に定義された部分ビームに関しオブジェクトが破棄される。

【0095】

50

図 4 E は、同方法のステップ 3 0 7 に従った左の部分ビーム及び右の部分ビームの更なる分割を示す。図 4 F は、部分ビームが光線と近似された方法のステップ 3 0 8 を示す。図 4 G は、それぞれのビームに一番近い三角形が見つけれられる同方法のステップ 3 0 9 を示す。図 4 H は、同方法のステップ 3 1 0 及び結果として区切られた部分ビームを示す（この段階では、部分ビームの処理が再開されているため、光線ではない）。

【 0 0 9 6 】

最後に、図 4 I は、部分ビームの合流が行われたステップ 3 1 1 の最終結果を示すものである。この場合、P 3 オブジェクトにより区切られた 2 つの部分ビームを合流することが可能であった。部分ビームを合流する P S オブジェクトの側面が異なる面に位置するため、P 2 の部分ビームを合流することはできない。その結果を最適な結果と比較することで、係数の適当な選択により正確な結果の許容範囲の近似が得られることを確定することが可能である。部分ビームを合流する段階は、この場合、部分ビームの最終的な数を 8 から 7 に減らすことができる。実際、合流アルゴリズムを使用した後、ビームの最終的な数が何度も減る状況が頻繁に起こる。

10

【 0 0 9 7 】

図 4 における全ての例は、二次元で描写されている。表示されているオブジェクトは、三次元のスペースにおいてはピラーである（図 4 のように、上面図ではなく正面図）。ピラーの斜視図のそれぞれの縁は、三角形により三次元に表されている。

【 0 0 9 8 】

図 5 は本発明に係るシステムのブロック図である。本発明に係るシステム 5 0 1 は既存の G P U ユニットと同様の役割を果たす専用のハードウェアモジュールとして作られても良い。そのようなモジュールは、ビームとシーンツリーを保存する階層モデルメモリ 5 0 7 及び交差テスト（レイ・トライアングル及びビーム・トライアングルの両方）を行う専用の信号プロセッサ 5 0 5 の 2 つの中心的部分に分けられる。モジュールは、シーン表現を使うことでソース及びレシーバーの位置により G P U （グラフィック処理装置）が画像をレンダリングするように使用されるのと同様の方法で音を生成するリアルタイムサウンドトレーシングを行うハードウェアの一部でもよい。

20

【 0 0 9 9 】

また、I / O コントローラ 5 0 6 及びメモリ 5 0 3 と協力してリアルタイムサウンドトレーシングを行う上記の方法のステップを実行する多目的コントローラ 5 0 2 がある。

30

【 0 1 0 0 】

内部回路は上記のアルゴリズムのハードウェア実装から成る。ソフトウェアからハードウェアへの変換を行うために V H D L 言語などの既存の業界標準のソリューションを使用することができる。

【 0 1 0 1 】

本発明は、好ましくは拡張カードなど、オブジェクトを感知する専用のハードウェアモジュールの形で実装されてもよい。この専用のハードウェアモジュールは完全なビームトレーシングあるいはサウンドレンダリングモジュールを作成するために他のモジュールと接続されてもよい。

【 0 1 0 2 】

そのような拡張カードのプロセッサは、ベクトル命令を利用して独立した一連のビームトライアングル交差テストを並行に処理されるマルチスレッドの論理ユニットを構成する。

40

【 0 1 0 3 】

また、そのようなプロセッサは、ベクトル命令を利用して独立した一連のレイ・トライアングル交差テストを並行に処理するマルチスレッドの論理ユニットを構成する。

【 0 1 0 4 】

拡張カードはアイドル状態の論理ユニットに一連のテストを割り当てる専用のディスパッチャ回路も構成する。

50

【0105】

拡張カードは、更にシーンモデルを保存する階層モデルを有するメモリを構成する。階層モデルメモリは、バスを介してプロセッサに接続されている。読み取り書き込みバッファは正規化された方向ベクトルの形で光線を構成してもよい。

【0106】

シーンツリーは、初期化の段階で階層モデルメモリにロードされる。拡張カードを使用するソフトウェアは、見えるオブジェクトが見つけれなければ成らない一連のビームをカードに送る。それに応じて、ソフトウェアはビームが分割された異なる光線と交差する三角形を受け取る。このような拡張カードは System on Chip (SoC) 技術に実装され、ビデオゲーム機あるいはコンピュータ及び他のマルチメディア及び波動現象シミュレーション用の専用のハードウェアであってもよい。

10

【0107】

ハードウェア実装は効率を上げるため、パフォーマンスが常に優先されるゲームの環境において有利である。

【0108】

説明されたアルゴリズムは音の伝播をシミュレートするビームトレーシングのアプリケーション及びビームトレーシングが使用される他のアプリケーションの処理を大幅にスピードアップする。ビーム処理及び光線処理のハイブリッドの組み合わせを使用することで適応フラスタムアルゴリズムの欠点が除去される。ビームを制限するオブジェクトを感知する結果的な方法は実装するのがより簡単で、ビームを合流するより効率の良いアルゴリズムの使用を可能にする。効率性においてかなりの改良があった一方でそれと同時に出力ビームの数が減った。

20

【0109】

適応フラスタムトレーシング (Adaptive Frustum Tracing) 及びユニフォームフラスタムトレーシング (Uniform Frustum Tracing) が混合され両方が適切に応用されたものである。1つ目のアルゴリズムは範囲外の三角形を早い破棄を可能にし、他のものは単純なテストを使用する。この解決法は複雑な破棄テストの実行を可能にし、光線に一連の簡単なテストとして実行される。

【0110】

当該方法は、マルチパス構造のため、最初にシーンツリーからビームと交差するセルを抽出し、続いてそれらのセルから所定の時間実行される破棄及びビーム分割アルゴリズムにパスされ、最後に部分ビームは光線と近似され、残った三角形はそれらの光線の交差される。必要がある場合、例えばセルをシーンツリーから抽出してそれらから三角形を得るなど、その後の段階は組み合わせられても良い。

30

【0111】

本発明の異なる実施例は、ソース・エミッタビームから全方向に放射する代わりに予め決められた数の光線を放射し、完全な三角形から反射を実行する（それぞれの三角形につき一回）：

最初に、光線が全方向に放射され、それらが交差する三角形が見つけれ、

次に、それらの全ての三角形に反射ビームが生成され、続いて通常の処理が行われる。

40

【0112】

当業者であれば、上記のサウンドトレーシングの方法は1つ以上のコンピュータプログラムにより実行及び/又は制御されることが可能であることは簡単に認識できる。そのようなコンピュータプログラムは、通常、パーソナルコンピュータ、パーソナルデジタルアシスタント、携帯電話、デジタルテレビのレシーバー及びデコーダーあるいは同様のものなどのコンピューティング装置におけるコンピューティングリソースを利用することで実行される。アプリケーションは固定媒体に保存される。固定媒体の例は、例えば、RAMなどのフラッシュメモリあるいは揮発性メモリなどの不揮発性メモリである。コンピュータの指示はプロセッサにより実行される。これらのメモリは本文において開示される技術的概念に係るコンピュータ実行方法の全てのステップを実行するコンピュータ実行可能な

50

指示を構成するコンピュータプログラムを保存する例示的記録媒体である。

【0113】

上述の発明は、特定の好ましい実施例に関して記載、説明及び定義されたが、本明細書における実施の参考及び例は、本発明を制限するものではない。当該技術的概念のより幅広い範囲から逸脱しないように多様な修正及び変更を行うことが可能なことは明らかである。上記の好ましい実施例は、単に例示的なものであり、上記の技術的概念の範囲を完全に示すものではない。

【0114】

当業者であれば、上記のビーム及び特にサウンドレーシングレーシングの方法は、光波、電波及び衝撃波などの他の物理的な波動現象をシミュレートするために実行されてもよいことは簡単に認識される。その結果、本方法はサウンドレーサー以外のマルチメディア及びシミュレーションシステムにおいて使用できる。

10

【0115】

従って、保護される範囲は、本明細書において説明される好ましい実施例に限られるものではないが、以下のクレームによってのみ限定される。

【图 1】

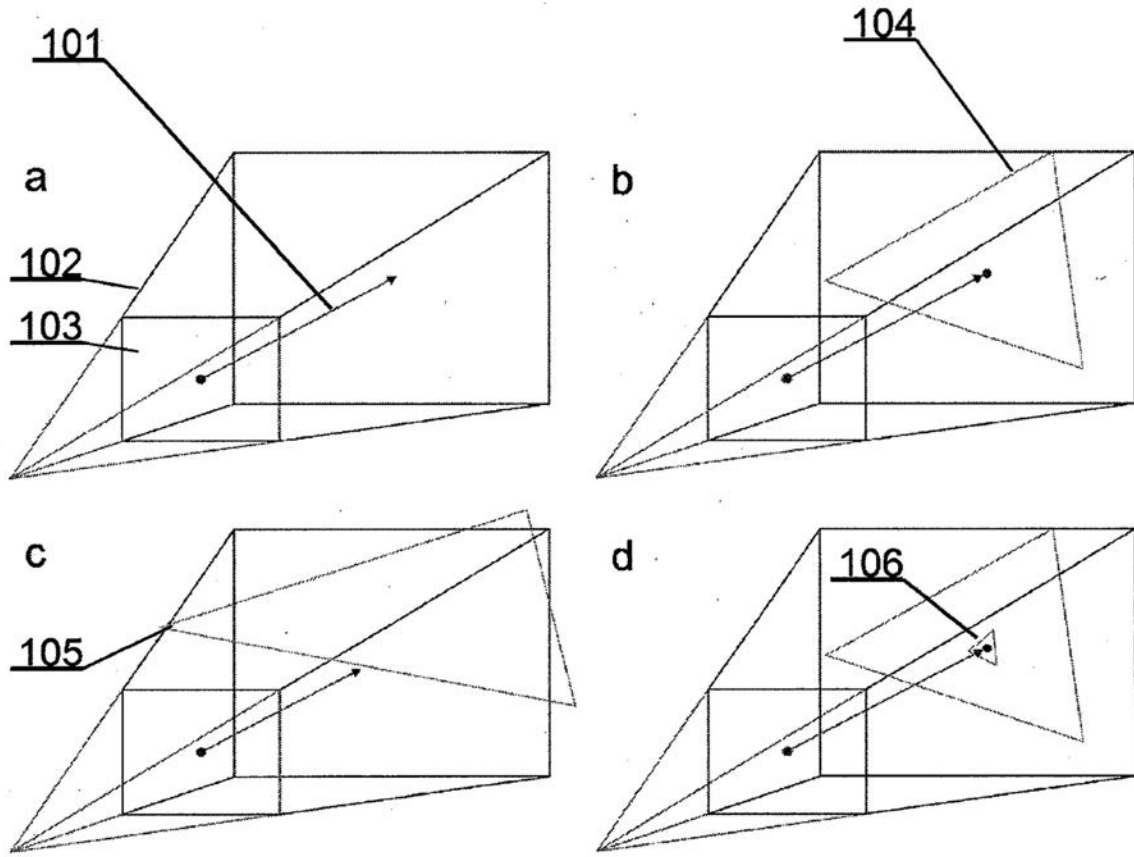


图 1

【图 2】

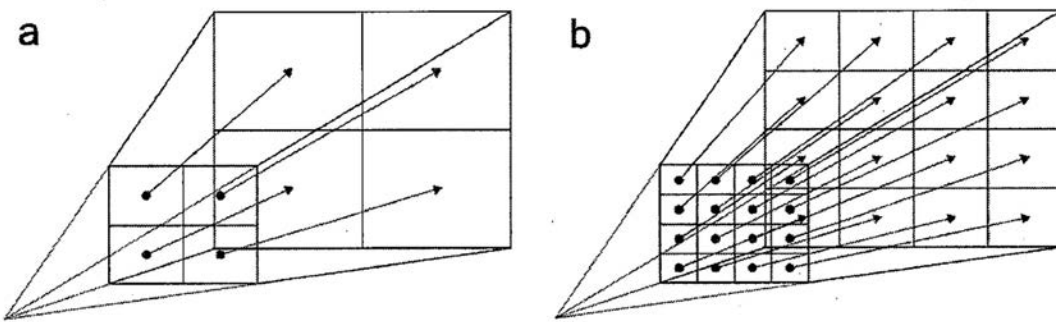


图 2

【図3】

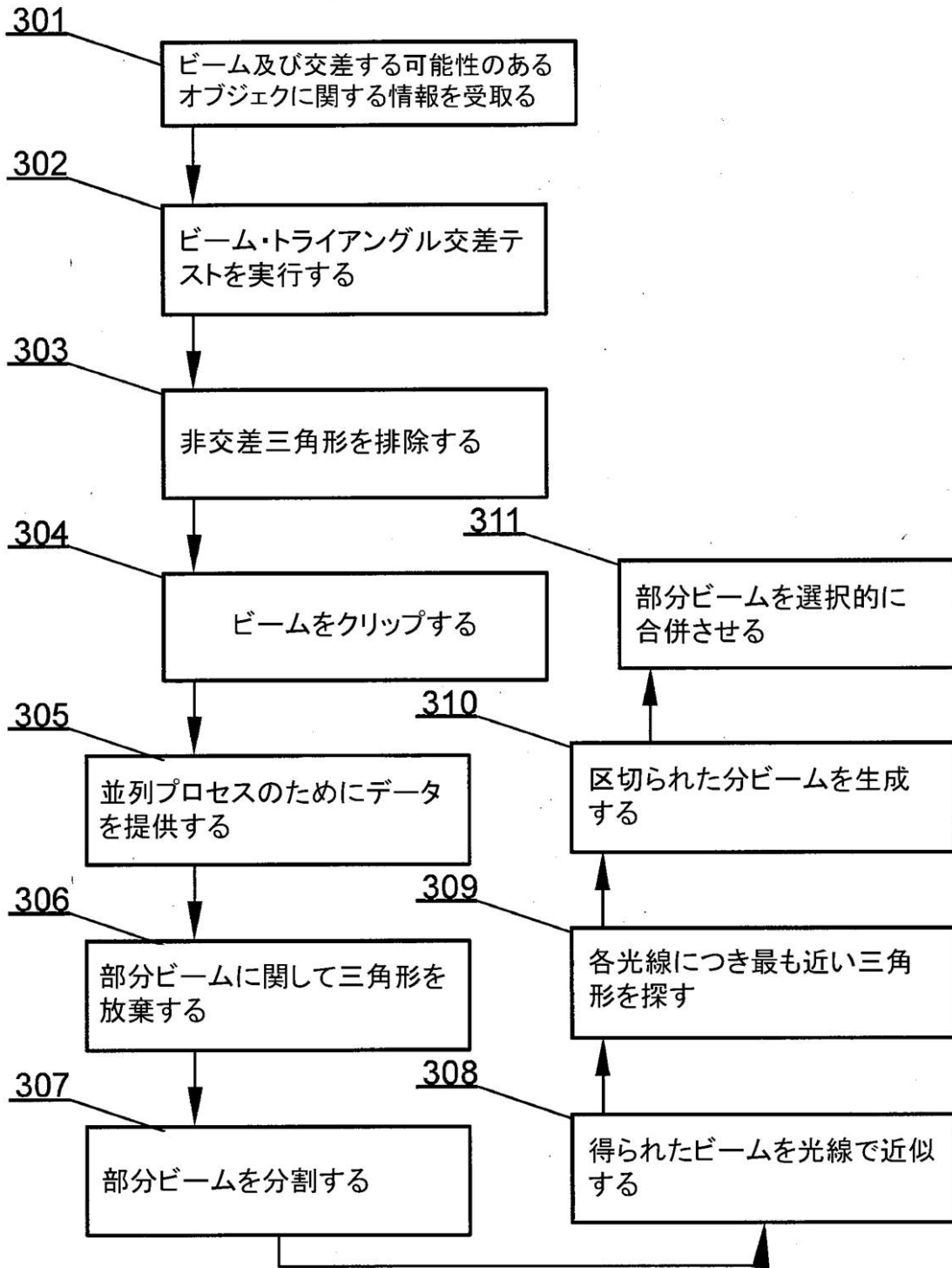


図3

【 図 4 A 】

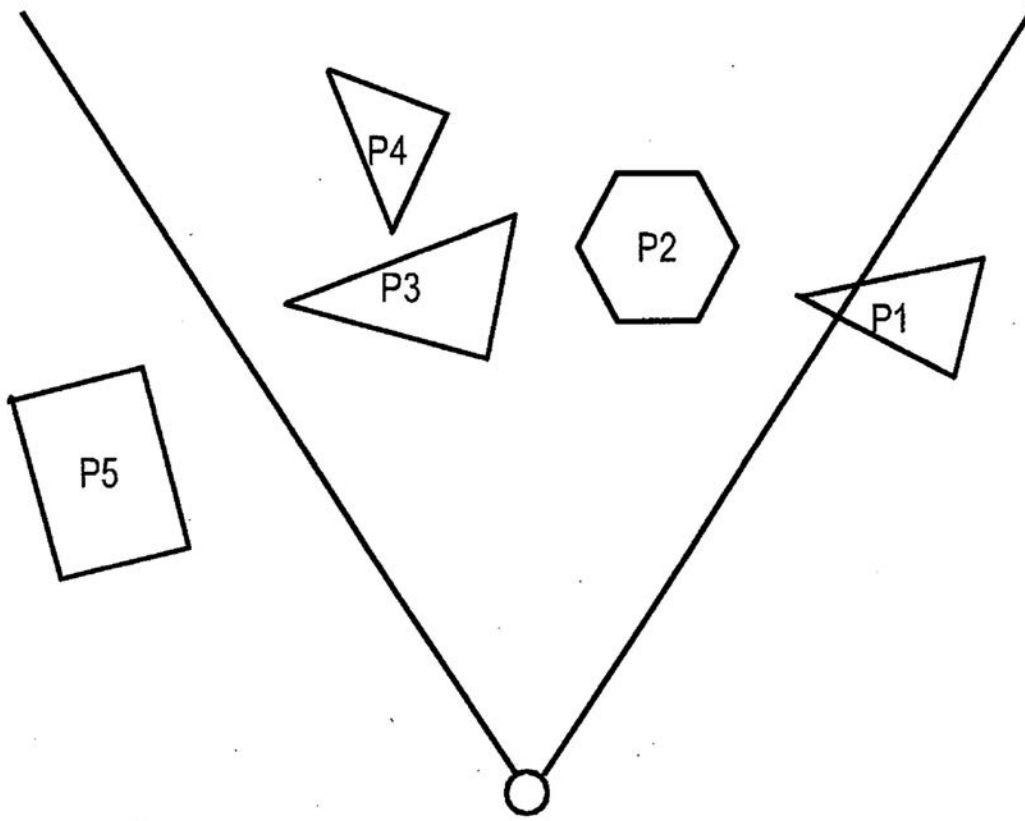


図 4A

【 図 4 B 】

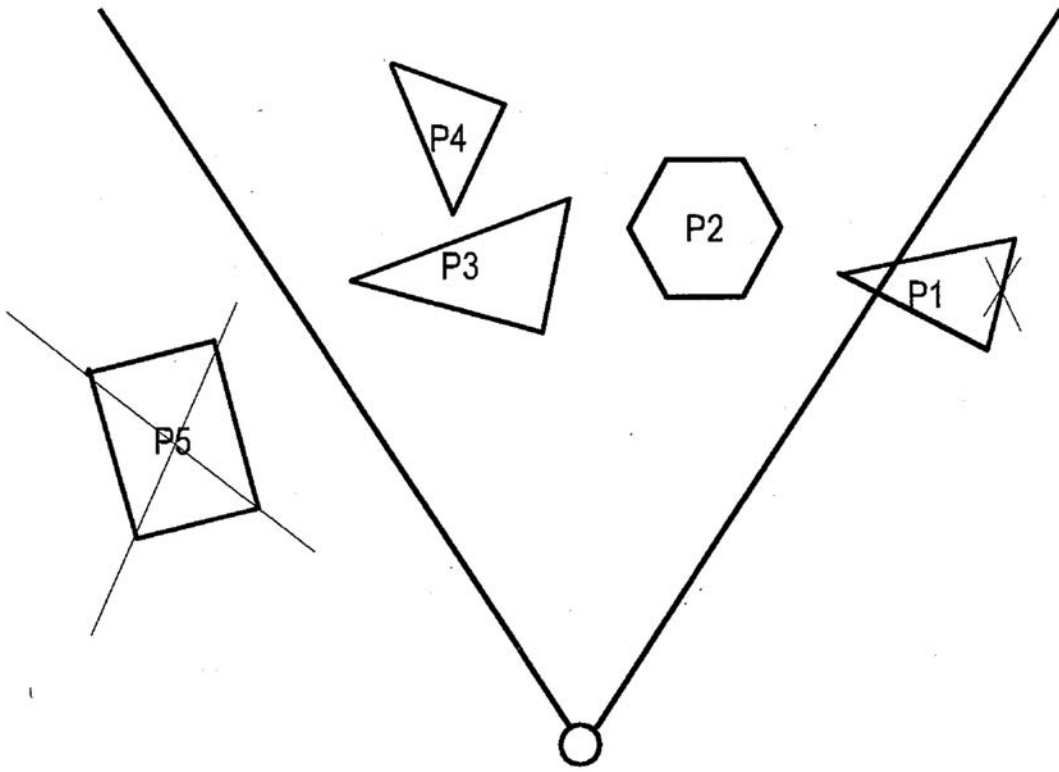


図4B

【 図 4 C 】

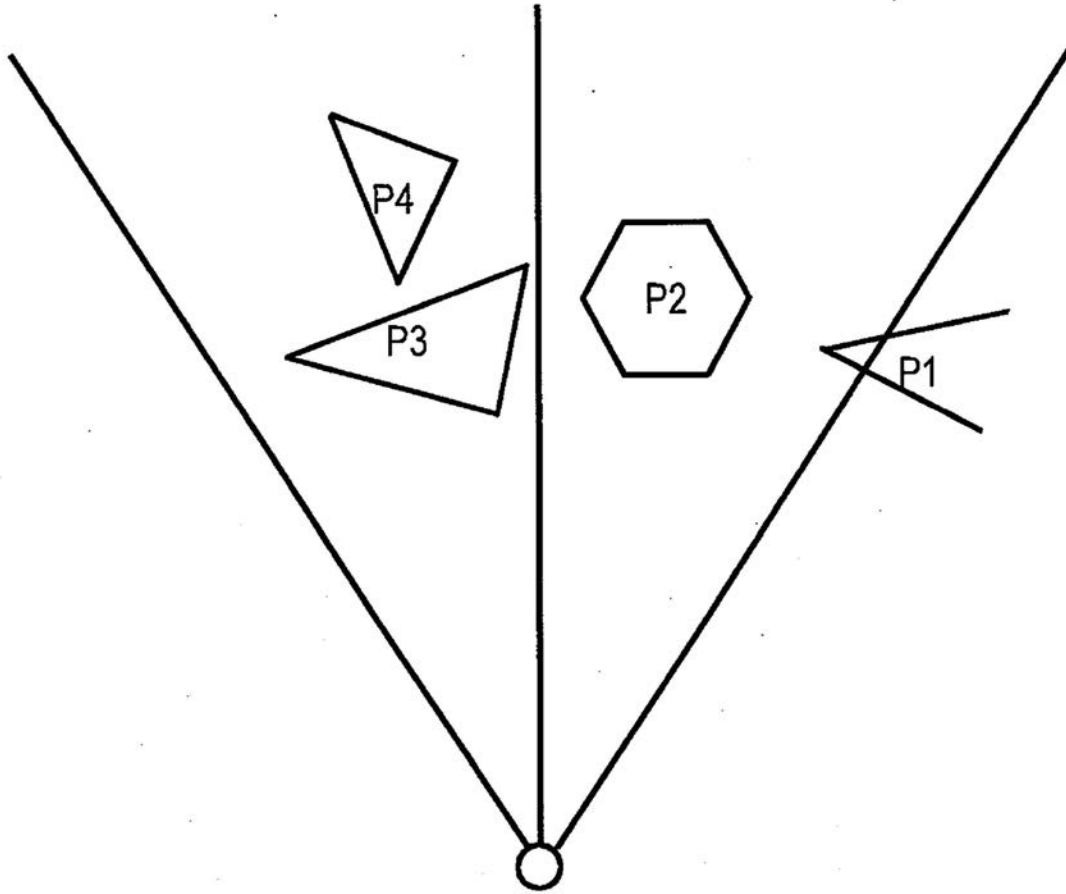


図4C

【 図 4 D 】

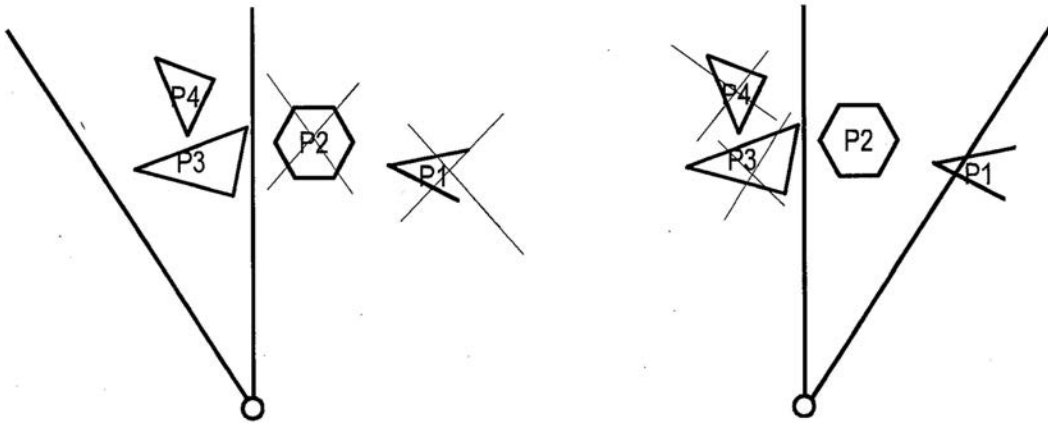


図4D

【 図 4 E 】

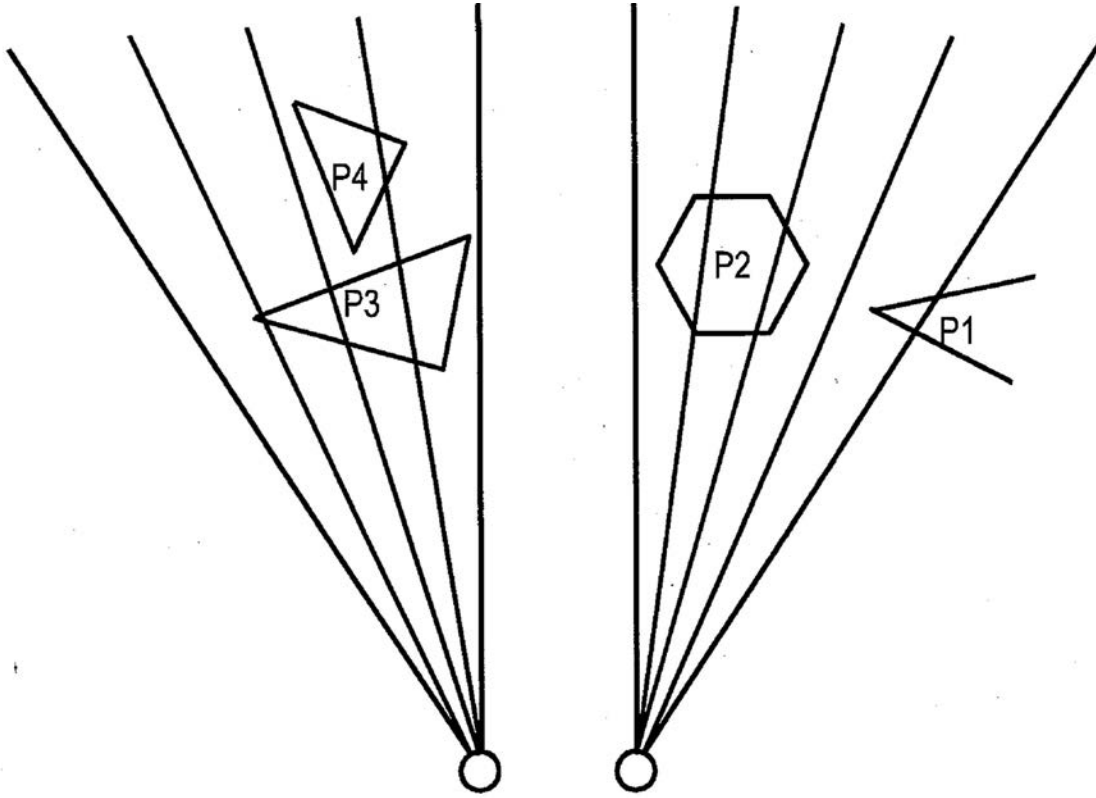


図4E

【 図 4 F 】

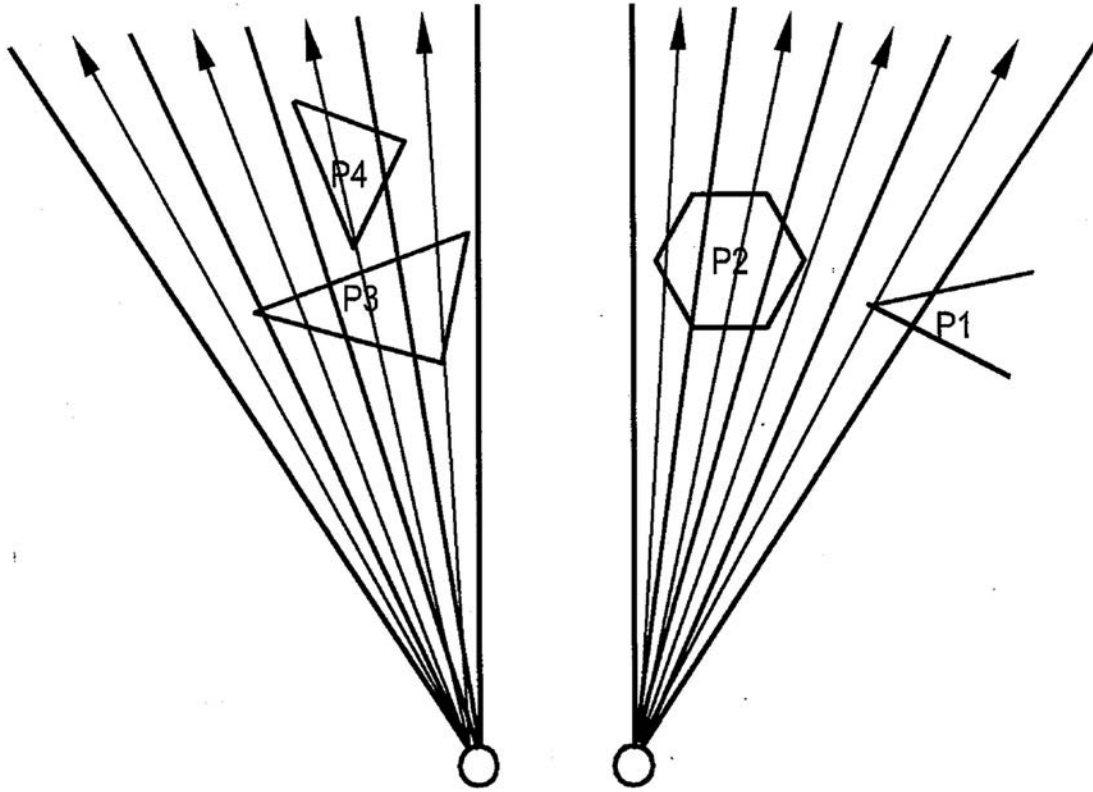


図4F

【 図 4 G 】

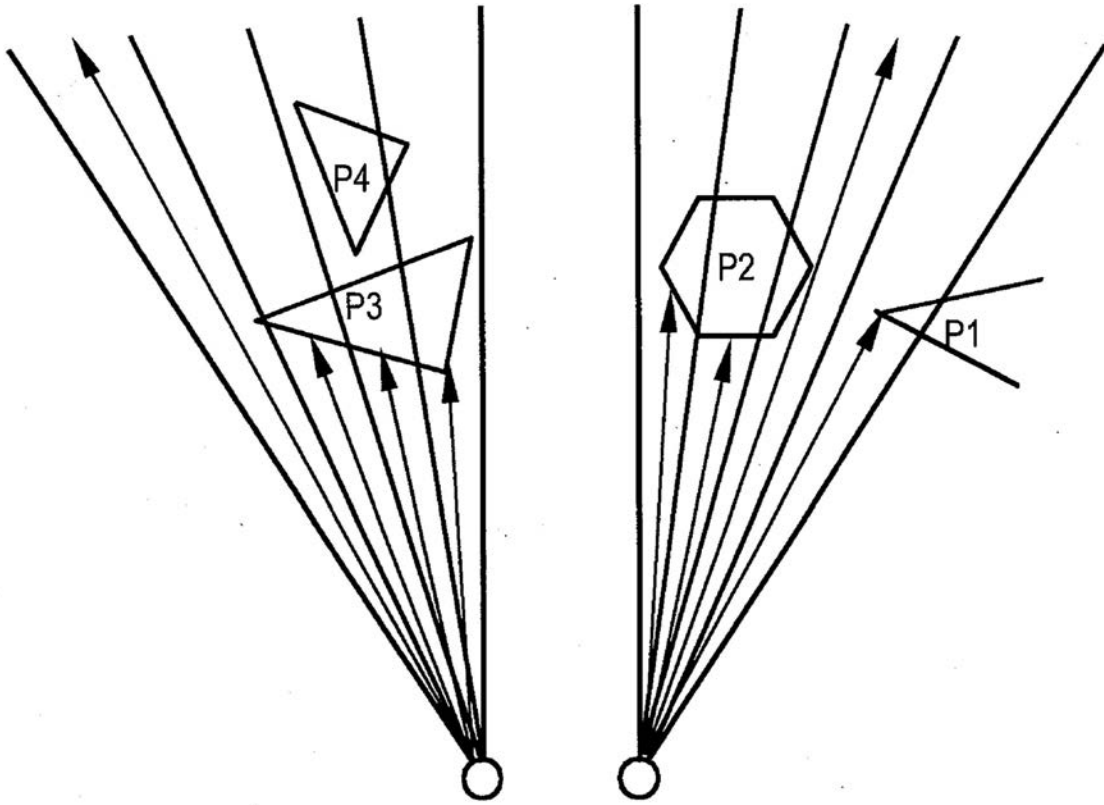


図4G

【 図 4 H 】

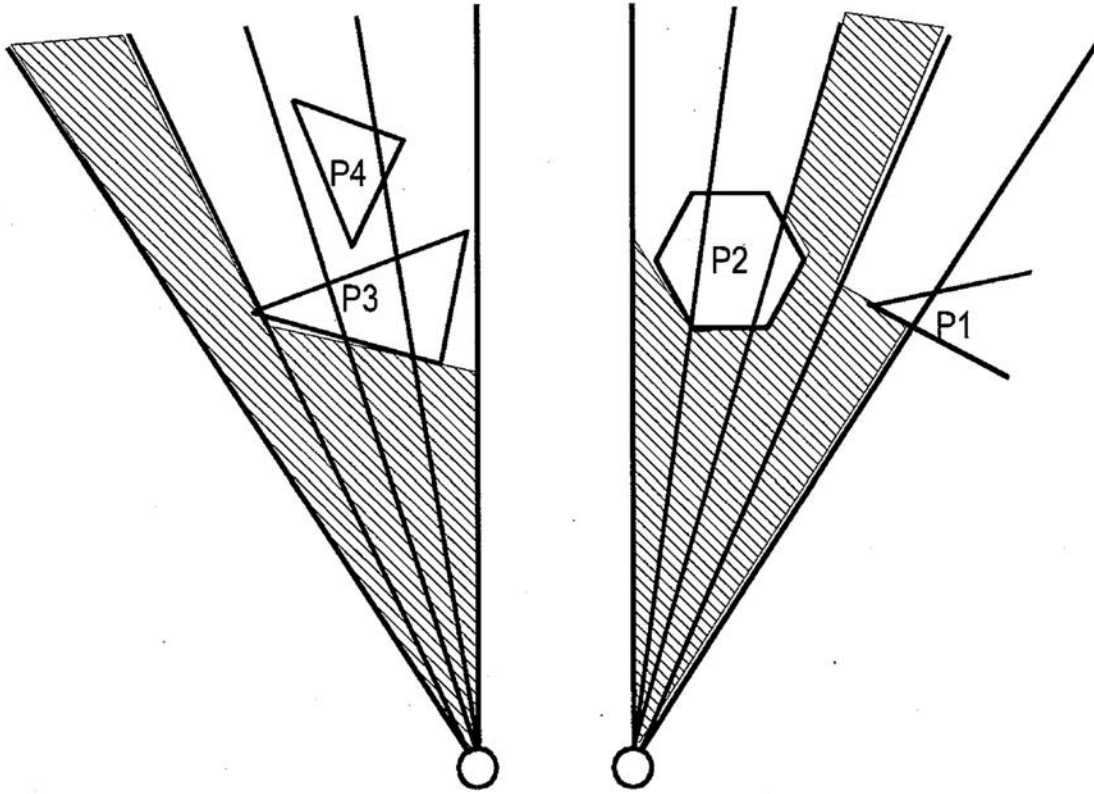


図4H

【 図 4 I 】

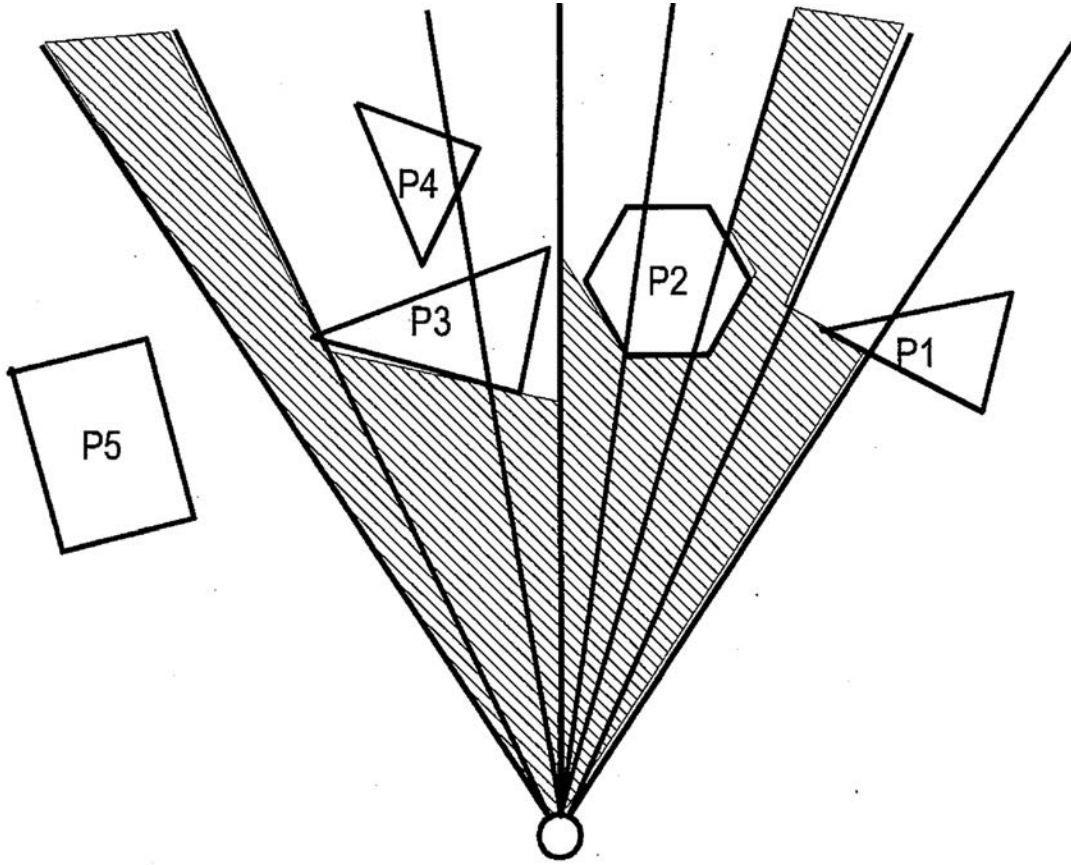


図 4I

【図5】

501

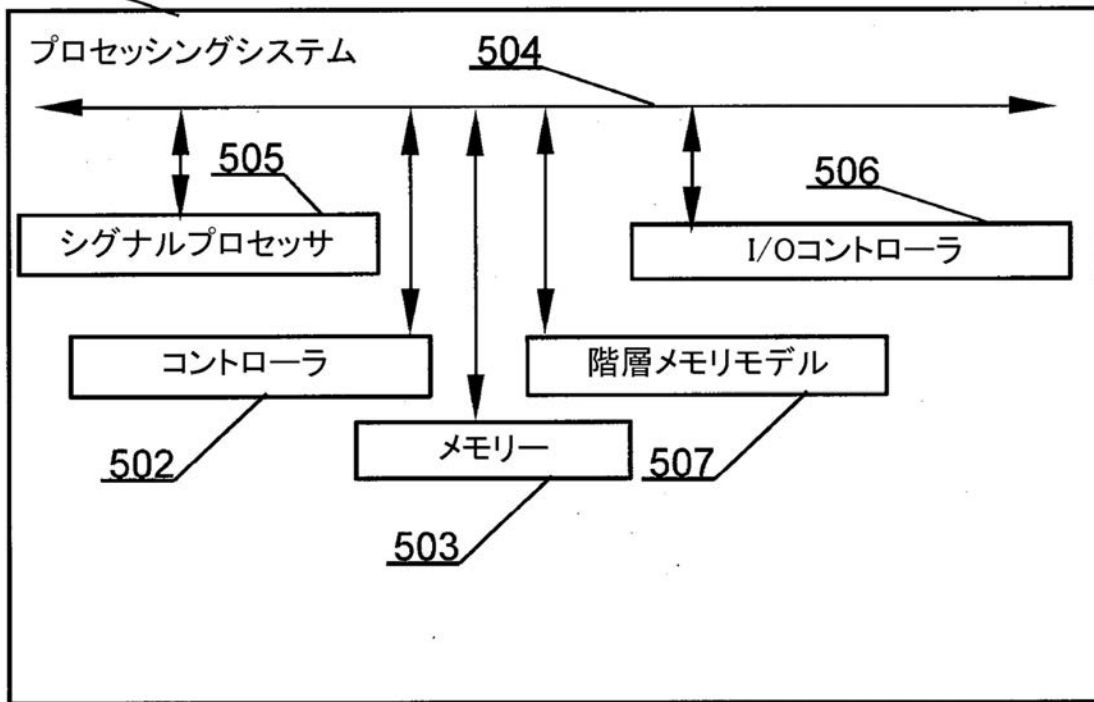


図5

フロントページの続き

(74)代理人 100099874

弁理士 黒瀬 靖久

(72)発明者 シモン パルカ

ポーランド国、 34 - 450 クロチェンコ、 ユーエル・ナボタルスカ 53

(72)発明者 トマシュ ペジマツ

ポーランド国、 31 - 315 クラコフ、 ユーエル・ラジコワスキエゴ 100F / 74

(72)発明者 パルトス ジオルコ

ポーランド国、 30 - 364 クラコフ、 ユーエル・ブホウィツカ 18G m. 20

【外国語明細書】

2015108703000001.pdf