計算機アーキテクチャ 60－3
（1985. 12. 6）

ベクトル計算機を用いた3次元隠れ面消去処理

北井 克佳、新實 治男、富田 眞治、萩原 宏
（京都大学・工学部）

1．はじめに

近年、人間と機械のコミュニケーションの手段として、コンピュータ・グラフィックスが広く利用されるようになってきている。応用分野も、従来の2次元グラフィックスによるCAD／CAM、フライト・シミュレーション等に加え、3次元グラフィックスによるCAD／CAM、3次元アニメーション等、大きな広がりを見せ、従来にも増してより高品質な画像をより高速に表示することが求められている。

図形表示処理の高速化、表示画像の高品質化を図るため、1960年代以降表示アルゴリズムを中心に精力的に研究がなされてきた。ハードウェア面からも、最近のハードウェア技術の進歩を背景にして、パイプライン方式やマルチプロセッサ方式による種々のグラフィックス専用マシンが研究・開発されている。その一方で、近年、汎用でかつ高速のスーパコンピュータが広く使われるようになり、種々の分野で大きな成果をあげている。スーパコンピュータの出現により、高速性を要するグラフィックス処理を汎用マシンで行なうのが良いのか、専用マシンを開発して行うのが良いのかは、1つの大きな問題点であると思われる。

本研究では、高速な3次元図形表示アルゴリズムの1つであるZバッファ法をベクトル計算機上に実現し、3次元図形表示処理を行った。Zバッファ法を2通りのアルゴリズムで実現し、さらに現実感を表わす処理としてGouraudの方法によるスムーズ・シェイディング処理、および影多角形法とレイキャスティング法の2通りの方法による付影処理を組み込んだ。

2．図形表示アルゴリズムの概要

図形表示処理は大きく初期処理と隠れ面消去処理の2段階からなる。

2．1．初期データ構造

初期データ構造を以下に示す。処理対象は多角形で構成された多面体とし、曲面は複数の多角形で近似することにより処理対象とする。図形データはすべて3次元ワールド座標系で定義する。1つの多角形は頂点の列で表現する。1つの多角形の頂点データは頂点ブロックの連続する領域に格納する。これにより、頂点データへのポインタをベースにして頂点データをインデキシングでき、下に述べる初期処理でのベクトル化を容易にしている。法線ベクトルデータは、曲面を多角形で近似した場合、その多角形の頂点における曲面の法線ベクトルであり、スムーズ・シェイディング処理で用いる。光源光は平行光線とする。

（i）多角形ブロック
- 頂点データへのポインタ
- 頂点数
- 面の色情報
- フラグ

（ii）頂点ブロック
- 頂点の座標(x,y,z)
- 法線ベクトルデータへのポインタ

（iii）法線ベクトルブロック
- 法線ベクトル(x,y,z)

（iv）制御ブロック
- 視点の位置(x,y,z)
- 視線の方向(x,y,z)
- 光源光の方向(x,y,z)
- スクリーンの位置と大きさ
- 背景色と環境の輝度

2．2．初期処理

a）座標変換処理：3次元ワールド座標系で定義されている図形を、視野変換および透視変換により、表示スクリーンを基準とするスクリーン座標系へ変換する。スクリーン座標系はスクリーンの奥行き方向をZ軸に、スクリーン面をZ＝0平面とする座標系である。

b）輝度計算：陽面（光源光のあたる面）と陰面（光源光のあたらない面）を判定し、次の式により面の輝度を求める。

陽面：$I = Ia + k \times (-\mathbb{N},\mathbb{L})/(|\mathbb{N}|\cdot|\mathbb{L}|)$、陰面：$I = Ia$

但し、$\mathbb{N}$:面の法線ベクトル、$\mathbb{L}$:光線ベクトル、Ia:環境の輝度、k:比例定数

c）後方面の除去処理、クリッピング処理[1]：視野変換後の視点座標系において、視点の反対方向を向いている面（後方面）を処理対象からはずし、視点ピラミッドの外側の頂点のクリッピング処理を行う。

2．3．Zバッファ法による隠れ面消去処理

画像記憶とは別に各画素ごとの面の奥行きを表わすZ値を記憶するZバッファを用いて、視点から見える面を決定し、その面の色情報を画像記憶に書き込む。アルゴリズムは次の通りである。

①すべての画素(x,y)に対して、Zバッファ(x,y)をZ値の最大値で初期化する。

②すべての面上のすべての点(x,y,z)に対して、z＜Zバッファ(x,y) ならば、

Zバッファ(x,y)←z、画像記憶(x,y)←面の色情報　を行う。

２．３．１．Ｚバッファ法のベクトル化にあたっての留意点

Ｚバッファ法による３次元図形表示処理をベクトル計算機を用いて行う場合、次の点に留意する必要がある。

1）アルゴリズム

ａ）大きさ、位置が任意の多角形データを均質化し、処理のベクトル化を容易にする。

ｂ）最も処理時間を要する隠れ面消去処理でのベクトル処理をどのデータに対して行うかを決定する。

2）データ構造

ａ）ベクトル計算機は倍精度の浮動小数点数の演算を高速に実行することを目的に設計されているため、４バイトデータに対する配慮は余りなされていない。Ｚバッファ法は、整数型データの処理量が比較的多いので、主記憶のバンク・コンフリクトを起こさないようにデータにアクセスするなどの配慮が必要である。

２．３．２．本研究で用いたアルゴリズムの概要

本研究で用いた２つのアルゴリズムは、ベクトル化を行うために、Ｚバッファ法をスキャンラインごとの処理に分解して隠れ面消去処理を行ったものである。アルゴリズム１は、各スキャンラインにおける隠れ面消去処理を順々に行っていくアルゴリズムであり、アルゴリズム２は、セグメントの分割により処理データの均質化を図り、スキャンライン数をベクトル長とするようにしたベクトル計算機向きのアルゴリズムである。

1）Ｚバッファを用いて隠れ面消去処理を行う前に次の処理を行う。以下にデータ構造を示す。

ａ）すべての処理対象多角形に対して、図１に示す手順で頂点データから隠れ面消去処理において処理単位となるセグメント・ブロックを作成する。

ｂ）セグメント・ブロックをｙの初期値によって、ylistをヘッダとする線形リストにつなぐ。

（ｉ）セグメント・ブロック

ｙの最終値
多角形ブロックへのポインタ
セグメントの左の稜線ブロックへのポインタ
セグメントの右の稜線ブロックへのポインタ
線形リストでの次のセグメント・ブロックへのポインタ
ｄｚ/ｄｘ

（ⅱ）稜線ブロック

ｘの初期値
ｄｘ/ｄｙ
ｚの初期値
ｄｚ/ｄｙ
輝度ｉの初期値
ｄｉ/ｄｙ

2）隠れ面消去処理

（ｉ）アルゴリズム１

各スキャンラインにおける隠れ面消去処理を、１スキャンライン分のＺバッファを用意することにより行う。なお、レイキャスティング法では１フレーム分のＺバッファを用意し、最初にＺバッファ全体をまとめて初期化している。

◎データ構造

xlist：現スキャンラインと交差するセグメント・ブロックへのポインタを入れる配列。

◎アルゴリズム

すべてのスキャンラインｙについて、

①Ｙの最終値＜ｙであるセグメント(退出セグメント)をxlistから除去する。

②ylist(ｙ)につながれているセグメント(進入セグメント)を新たにxlistにつなぐ。

③隠れ面消去処理：

Ｚバッファ(ｘ)←Ｚ値の最大値

すべてのxlist中のセグメントについて

セグメントの左端から右端に対して、Ｚバッファ(ｘ)を用いた隠れ面消去処理を行う。

④セグメントを構成する左稜線、右稜線のｘ値、ｚ値をdx/dy,dz/dyを用いて更新する。

◎アルゴリズムの長所・短所

長所：①Ｚバッファの容量が１スキャンライン分でよく、主記憶の節約が図れる。

②スキャンライン間の処理を順次的に行うため、稜線のスキャンラインとの交差情報等を保持しやすい。

短所：①処理量の大半を占める隠れ面消去処理でのベクトル長がセグメントのｘ方向の長さとなるため、表示図形が細かくなると、ベクトル化による処理効率の向上を期待できない。

（ⅱ）アルゴリズム２

アルゴリズム１の短所である隠れ面消去処理でのベクトル長の短かさの改善を図ったアルゴリズムである。各スキャンラインにおいて処理可能なセグメントを最大１つずつ選び、このセグメント数をベクトル長として隠れ面消去処理を行う。なお、スキャンラインＹにおける処理可能なセグメントとは、ylist(Ｙ)につながれているセグメントまたはスキャンライン(Ｙ－１)までの処理が終了しているセグメントをいう。

◎データ構造

slist：各スキャンラインの処理可能なセグメント・ブロックへのポインタを入れる配列。

◎アルゴリズム

①セグメントの大きさをできるだけ均等にするために、ｘ，ｙ各方向について、セグメントの最長部が全セグメントの平均長より長いセグメントを分割する。

②Ｚバッファ(ｘ，ｙ)←Ｚ値の最大値

㋐処理可能セグメント数＞０の間、以下の処理を繰り返す。

①slistから退出セグメントを除去する。

②slist(y)=null かつ ylist(y)≠nullの時、slist(y)にylist(y)の先頭セグメントをつなぐ。

③隠れ面消去処理：slist(y)≠nullの要素を収集した後、slist中のセグメントを図２に示す手順で隠れ面消去処理を行う。

④稜線のｘ値、ｚ値の更新、slistの更新(slist(i+1)←slist(i),slist(0)←null)を行う。

◎アルゴリズムの長所・短所

長所：㋐処理量の大半を占める隠れ面消去処理を、ベクトル長をスキャンライン数として実行できる。

㋑複数のセグメントを並列に処理するにもかかわらず、各スキャンラインから１つのセグメントを選ぶことによりＺバッファの回帰的参照を回避できる。

短所：㋐１回のループでの処理セグメント数がスキャンライン数で抑えられるため、処理データ量が増えるとループ回数が増大し、更新処理の比率が高まる。

㋑セグメントの分割により、データ量が増大する。

２．４．スムーズ・シェイディング処理

多角形で近似した曲面を、隣り合う画素の輝度をなめらかに変化させることにより、曲面らしく見せる処理である。本研究では、多角形の頂点における曲面の法線ベクトルと光源光ベクトルより各頂点の輝度を求め、次に、頂点の輝度を線形補間することにより各点の輝度を求めるGouraudの方法[2]を用いた。

２．５．付影処理[3]

本研究では、代表的な付影処理アルゴリズムである、影多角形法およびレイキャスティング法を用いた。

１）影多角形法[4]

影多角形は、陽面と陰面の境界である輪郭稜線と光源光ベクトルとにより定義できる。各図形ごとに影多角形は影多角柱を構成する。このアルゴリズムは、影はこの影多角柱の中に含まれる陽面に落ちるという性質を用いる。

◎アルゴリズム

ａ）初期データとして、㋐地面を表わす多角形ブロックへのポインタ、および、㋑図形の稜線を構成する２つの多角形ブロックと２つの頂点ブロックへのポインタを与える。

ｂ）初期処理における視野変換後、影多角形を生成する。さらに、前方面/後方面の判定を行う。

ｃ）Ｚバッファ法による隠れ面消去処理後、各画素の影指数を記憶する影バッファを用いて付影処理を行う。アルゴリズムは次の通りである。

①すべての画素(ｘ,ｙ)に対して、影バッファ(ｘ,ｙ)を０で初期化する。

②すべての影多角形上のすべての点(ｘ,ｙ,ｚ)に対して、ｚ＜Ｚバッファ(ｘ,ｙ) ならば、

・影多角形が前方面であるならば、影バッファ(ｘ,ｙ)←影バッファ(ｘ,ｙ)＋１

・影多角形が後方面であるならば、影バッファ(ｘ,ｙ)←影バッファ(ｘ,ｙ)－１　を行う。

③影バッファの値が正の画素は影多角柱の中に入っていることになるので、この画素の輝度を落とす。

◎アルゴリズムの長所・短所

ａ）長所：影バッファに対する処理以外は、影多角形を実多角形と同等に扱える。

ｂ）短所：影多角形の処理面積は実多角形に比べ圧倒的に大きいため、付影処理の時間をかなり要する。

２）レイキャスティング法

このアルゴリズムは、視点からは見えるが、光源からは見えない面には影が落ちるという性質を用いる。

◎アルゴリズム

ａ）視点を中心とする座標系で、Ｚバッファ法による隠れ面消去処理を行い、画像記憶に色情報を書き込む。

ｂ）光源を中心とする座標系で、Ｚバッファとは別の影バッファを用いて、Ｚバッファ法による隠れ面消去処理を行う。

ｃ）Ｚバッファの値がＺ値の最大値でないすべての画素点(Xe,Ye,Ze)を、光源を中心とする座標系での座標(Xl,Yl,Zl)に変換する。(Zl－影バッファ(Xl,Yl))＞εならば、画素(Xe,Ye)の輝度を落とす。

◎アルゴリズムの長所・短所

ａ）長所：基本的に実多角形に対する隠れ面消去処理を２回行えばよく既存のルーチンを使える。

ｂ）短所：視点座標系と光源座標系の格子系の違いにより、光源座標系に変換された視点座標系の各画素は光源座標系の格子点からはずれる。このため、何らかの補正を行わないと、影の輪郭線にギザギザを生じる。

３．結果

３．１．処理の仮定

ａ）画像記憶、Ｚバッファ、影バッファはすべて主記憶上に設定する。

ｂ）処理時間には初期データのディスクからのロード時間および画像記憶のディスクへの格納時間は含まない。

ｃ）スクリーンの大きさは、５１２×５１２画素である。

d）１画素のR(赤)G(緑)B(青)I(輝度)は、それぞれ３，３，３，７ビットである。
e）演算はすべて倍精度で行っている。
使用した計算機は京都大学大型計算機センターのＦＡＣＯＭ　ＶＰ１００，ＶＰ２００である。ベクトル化コンパイラはＶＰ１００はＶ１０Ｌ１０，ＶＰ２００はＶ１０Ｌ２０である。Ｌ１０とＬ２０の最大の違いは、収集・拡散手続きのベクトル化の有無である。

３．２．処理で用いたデータの属性

本研究で用いたデータは立方体と球である。表１にデータの属性を示す。

３．３．処理結果

①処理時間は、ＣＰＵ時間を計測するシステム・サブルーチンclockを用いて計測した値からサブルーチンコールによるオーバヘッド２９マイクロ秒(実測値)を引いた時間である。
②表の時間の単位はミリ秒である。
③ベクトル処理時間はベクトル・ユニットが使われたＣＰＵ時間である。
④ベクトル化率は、(ベクトル化された部分のスカラでの処理時間)／(スカラでの全処理時間)による。

１）付影処理を行わない場合（結果を表２、表３に示す）
①立方体のように、図形がスクリーン上全体に表示される場合は、アルゴリズム２の利点が生かされるが、sp8のように、表示データが狭い範囲に限定されるとアルゴリズム２の性能は落ち、逆に処理データのないスキャンラインの処理をバイパスできるアルゴリズム１が有効になる。
②両アルゴリズムでの処理時間３の違いは、セグメントの分割処理の有無による。
③アルゴリズム１で、立方体の数が少ない程、ベクトル化率が良いのは地面の処理の比率が高いためである。

２）影多角形法による付影処理を行った場合（結果を表４、表５に示す。立方体３００個の場合の隠れ面消去処理の詳細データを図３に示す）
①図３より、アルゴリズム２はアルゴリズム１に比べ、隠れ面消去処理での処理速度は約２．４倍になっているが、更新処理は逆に約半分になっており、ベクトル化を図るための処理時間が無視できないものとなっている。
②アルゴリズム２の更新処理２のＶＰ１００とＶＰ２００の処理時間差は収集手続きのベクトル化による。

３）レイキャスティング法による付影処理を行った場合（結果を表６、表７に示す）
①アルゴリズム１において、Ｚバッファを１フレーム分持たせたことにより、表２に示す結果と比べ、処理時間１は大幅に短縮されている。これは、㋐各スキャンラインの処理の開始時にＺバッファの初期化を行うのではなく、最初にＺバッファ全体を初期化するため、および、㋑Ｚバッファを１フレーム分持たせたことにより、Ｚバッファの配列のインデックスが画像記憶と等しくなり、インデックスの計算時間が短縮されたと思われるため、である。
②アルゴリズム１において、処理時間１より２が短かいのは、スムーズ・シェイディング処理を行うか否かの判定処理を後者は含まないためである。

４．考察

４．１．２つの隠れ面消去アルゴリズムに対する考察

１）アルゴリズム１
（ｉ）ベクトル化による効果
ベクトル化による隠れ面消去処理の処理速度の向上率は約１．５～２倍である。処理速度の向上に寄与しているのは、主として稜線の更新処理である。
（ii）ＶＰ１００とＶＰ２００の比較
処理時間を要する隠れ面消去処理では、Ｚバッファに対するアクセスのベクトル長(平均セグメント長)が短かいため、ハードウェア量が２倍に増えても、処理時間の改善はあまりなかった。
付影処理を行わない場合には、ベクトル処理時間はＶＰ２００の方が短かくなっているにもかかわらず、隠れ面消去処理の時間は増大している。このことより、ＶＰ２００の方がベクトル処理の準備時間が長くなっているのではないかと思われる。
（iii）考察
アルゴリズム１はスカラ向きのアルゴリズムといえるが、稜線の更新処理、座標変換処理などのベクトル化による処理速度の向上により、全体として、２～３倍処理速度が上がった。また、レイキャスティング法の場合のように、Ｚバッファを１フレーム分持つことにより処理速度は更に向上する。

２）アルゴリズム２
（ｉ）ベクトル化による効果
隠れ面消去処理での処理速度はスカラに対して７～９倍に向上しているが、アルゴリズム１と比較すると隠れ面消去処理での処理速度の向上分は、以下の２つのオーバヘッドに吸収されてしまい、所期に期待した効果は得られなかった。
①ベクトル計算機向きにアルゴリズム１を改善したのに伴い、アルゴリズム１に比べ、次の２つの原因によりスカラ

での処理時間が大幅(約３倍)に増大している。

・隠れ面消去処理の制御が複雑化している（セグメントによるＸ方向の長さの違いを制御するため、if文のネスティングが１重増えているなど）

・Ｚバッファへのアクセスが順アクセスからリストベクトルによる間接アクセスになっている。

②セグメントの分割に伴うデータ量の増大もあり、並列性を上げるために必要な更新処理の割合が大きくなってしまっている。

（ii）ＶＰ１００とＶＰ２００の比較

①隠れ面消去処理は、㋐ベクトル長が比較的長いこと、㋑１つのループ内の演算数もかなりあること、のため、ハードウェア量が２倍に増えたのに伴い、処理速度も１．５～１．９倍になっている。マスク・パイプラインが２本以上あれば、更に効率が上がるものと思われる。

②slistの更新処理はベクトル化されるが演算器を余り用いないため処理速度の向上は高々４倍にとどまる。

以上より、

①１回のループでの処理セグメント数が５１２個に制限されるため、ベクトル計算機で実行した場合、データ量の増大に伴い更新処理の占める割合が大きくなり、処理速度の向上率が下がってしまう。

②処理データが細かくなった場合にも、１回のループでの隠れ面消去処理の比率は下がってしまうため上記のことがあてはまる。

（iii）考察

強力なリストベクトル処理機能、マスク・パイプラインの多重化、およびハードウェア量の増加により隠れ面消去処理での処理時間の短縮は期待できるが、処理全体を短縮化するためには、並列性を出すために必要な更新処理でのアルゴリズムの改善が必要である。

４．２．２つの付影処理アルゴリズムに対する考察

１）影多角形法

・処理面積が実多角形に比べ大きい分がそのまま付影処理に伴うコストの増大として現われている。

・基本的に、影バッファの処理はＺバッファの処理と同じであるため、４．１の考察がそのまま影多角形法による付影処理にもあてはまる。従って、影多角形法による付影処理は影多角形の処理面積の多さを考えると、Ｚバッファ法と組み合わせて用いるのは有効であるとは言い難い。

２）レイキャスティング法

・視点系と光源系の２つの隠れ面消去処理は、影多角形を用いる場合と異なり、実多角形のみに対する処理であるので、付影処理を行わない場合の約２倍ですんでいる。

・この方法で問題となるのは、視点系の各画素点の光源系への変換処理である。スカラ処理の場合、アルゴリズム１では、処理時間の過半数をこの座標変換処理が占めてしまい、付影処理に伴うコストの増大はかなり大きなものとなる。しかし、ベクトル処理の場合には、行列の乗算を中心とする演算を全画素に対して行うため、演算数、ベクトル長共に十分大きく、ベクトル化による効果は大きい。従って、ベクトル計算機向きのアルゴリズムといえる。

５．おわりに

本研究では、Ｚバッファ法による３次元図形表示処理をベクトル計算機を用いて行ない、更に、２種類の付影アルゴリズムを組み込んだ。レイキャスティング法による付影処理を行った場合以外は、必ずしもベクトル計算機の性能を生かすことはできなかった。今後は、他のアルゴリズムによる３次元図形表示処理をもベクトル計算機を用いて行い、ベクトル計算機との適合性を調べていきたい。

６．参考文献

[1] I.E.Sutherland et al：Reentrant Polygon Clipping,CACM,Jan.,1974.

[2] H.Gouraud：Continuous Shading of Curved Surfaces,SJCC,1970.

[3] F.C.Crow：SHADOW ALGORITHMS FOR COMPUTER GRAPHICS,Computer Graphics,No.2,1977.

[4] J.Bouknight,K.Kelley：An algorithm for producing half-tone computer graphics presentations with shadows and movable light sources,IEEE Trans. on Comp.,Jun.,1971

表１．使用データの属性

データの種類		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
初期多角形数	実	1801	1501	1201	901	601	301	2701	2001	1354
	影	1824	1514	1214	912	606	304	594	220	215
処理多角形数	実	888	739	593	444	298	148	1251	842	599
	影	1762	1467	1167	881	585	294	594	220	215
処理ｾｸﾞﾒﾝﾄ数	実	2416	2005	1627	1217	827	422	3306	1813	1105
	影	4856	4054	3232	2425	1610	809	1794	587	524
平均ｾｸﾞﾒﾝﾄ長（X方向）	実	8	8	8	8	8	9	8	6	5
	影	6	6	6	6	6	6	9	4	3
平均ｾｸﾞﾒﾝﾄ長（Y方向）	実	5	5	5	6	6	6	5	3	3
	影	48	49	48	51	52	49	33	52	52
処理画素数	実	114134	96831	80274	62136	42777	23719	164958	44657	21288
	影	1623361	1378958	1102972	866564	572190	274166	592670	148480	94554
平均面積/多角形	実	128.5	131.0	135.4	139.9	143.5	160.3	131.9	53.0	35.5
	影	921.3	940.0	945.1	983.6	978.1	932.5	997.8	674.9	439.8

註．①データの種類：ｃは立方体、spは球を、数は個数を表わす。
立方体はランダムな大きさであり、ランダムな位置に置かれている。
sp２７は１０×１０個の多角形で近似した球を３×３×３個積み重ねたデータである。
sp８は表示図形のスクリーンに占める割合が小さく、全スキャンラインの約３０％ではデータがない。
②処理多角形数：クリッピング処理、後方面の除去処理後の、実際の処理対象となるデータ数を表わす。
③セグメント数、画素数はスクリーン座標系における値であり、視線、光線に依存する。
④平均面積：処理画素数／処理多角形数で１多角形当りの平均面積を求めている。
⑤実は実多角形、影は影多角形法で用いる影多角形のデータである。
⑥各データはスクリーン上での大きさが５１２×１７９の地面を表わす多角形を含む。

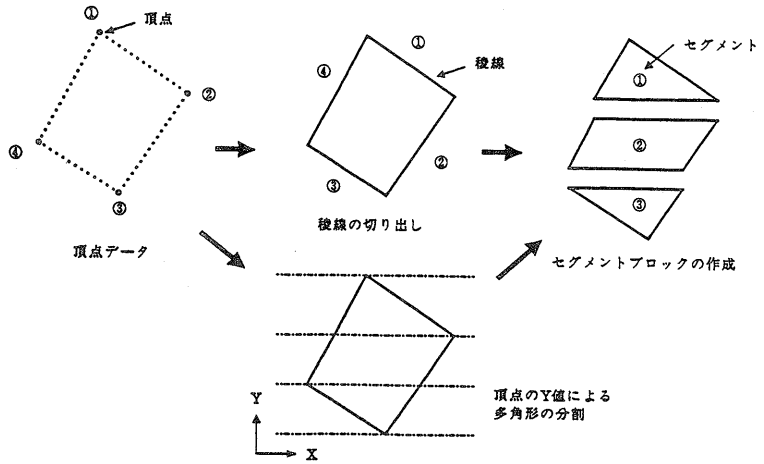


図１．セグメントブロックの作成

表６．アルゴリズム１（レイキャスティング法）

スカラ(VP200)		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	793	680	571	456	339	219	1539	828	544
	2	622	544	467	390	311	233	823	548	403
	3	1764	1705	1621	1540	1445	1330	1998	1506	1305
	4	150	146	143	140	137	133	177	165	154
	合計	3329	3075	2802	2526	2231	1914	4536	3047	2405

ＶＰ２００		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	307	263	222	178	133	84	483	243	155
	2	221	188	156	124	91	58	296	181	120
	3	101	98	94	90	85	79	122	90	79
	4	3	3	3	3	3	3	3	3	3
	合計	632	553	474	395	312	224	904	517	357
ｽｶﾗ/VP200		5.3	5.6	5.9	6.4	7.2	8.6	5.0	5.9	6.7
ﾍﾞｸﾄﾙ処理時間		360	320	280	250	210	160	540	300	220
ﾍﾞｸﾄﾙ化率		91.8%	92.4%	93.1%	94.3%	95.4%	96.7%	92.0%	92.9%	94.3%
処理画素数/秒		180677	175217	169340	157338	137172	105972	182404	86446	59694
処理多角形数/秒		1406	1337	1251	1124	956	661	1383	1630	1680
初期多角形数/秒		2851	2716	2534	2281	1927	1345	2987	3873	3797

註．①処理時間の１、２、３、４はそれぞれ視点座標系での隠れ面消去処理、光源座標系での隠れ面消去処理、視点座標系から光源座標系への座標変換処理、画像記憶・Ｚバッファ・影バッファの初期化処理等の時間である。

表７．アルゴリズム２（レイキャスティング法）

スカラ(VP200)		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	1750	1520	1300	1037	790	537	2914	1534	1180
	2	1528	1402	1135	1010	890	649	1892	1718	1479
	3	1606	1555	1482	1411	1325	1227	1810	1379	1204
	4	150	146	143	140	137	133	177	165	154
	合計	5035	4623	4061	3598	3141	2546	6793	4797	3997

ＶＰ２００		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	292	261	228	188	150	110	447	265	211
	2	308	282	221	194	169	124	378	340	288
	3	102	99	94	90	85	79	123	90	79
	4	3	3	3	3	3	3	3	3	3
	合計	704	645	546	475	406	316	951	698	580
ｽｶﾗ/VP200		7.1	7.2	7.4	7.6	7.7	8.1	7.1	6.9	6.9
ﾍﾞｸﾄﾙ処理時間		490	460	400	360	330	270	670	500	440
ﾍﾞｸﾄﾙ化率		95.7%	96.0%	96.4%	96.8%	97.6%	98.2%	95.9%	95.9%	96.5%
処理画素数/秒		162048	150197	146910	130828	105428	75043	173451	63992	38681
処理多角形数/秒		1261	1146	1085	935	734	468	1315	1207	1032
初期多角形数/秒		2557	2328	2198	1897	1481	952	2840	2867	2333

註．①処理時間の１、２、３、４はそれぞれ視点座標系での隠れ面消去処理、光源座標系での隠れ面消去処理、視点座標系から光源座標系への座標変換処理、画像記憶・Ｚバッファ・影バッファの初期化処理等の時間である。

表2．アルゴリズム1（付影処理なし）

スカラ(VP100)		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	201	168	135	102	69	35	450	322	225
	2	537	474	414	348	280	199	1198	530	306
	3	166	144	126	105	85	65	286	197	144
	合計	904	787	674	555	434	299	1934	1049	675

VP100		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	49	41	33	25	18	10	80	56	40
	2	278	240	205	167	124	71	430	170	93
	3	88	75	62	48	34	20	122	78	53
	合計	415	356	301	240	177	101	631	305	186
ﾍﾞｸﾄﾙ処理時間		210	180	150	120	90	60	320	160	100
ﾍﾞｸﾄﾙ化率		77.4%	77.6%	77.5%	78.3%	80.0%	86.2%	83.9%	86.2%	87.3%
ｽｶﾗ/VP100		2.2	2.2	2.2	2.3	2.5	2.9	3.1	3.4	3.6

VP200		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	43	36	29	22	15	8	70	50	35
	2	293	252	214	172	128	76	385	154	82
	3	75	63	51	38	27	15	105	66	42
	合計	411	351	293	233	170	99	560	270	160
ﾍﾞｸﾄﾙ処理時間		170	140	120	100	80	50	260	120	80
ﾍﾞｸﾄﾙ化率		73.4%	73.2%	74.3%	76.1%	79.3%	83.5%	84.5%	85.7%	88.2%
ｽｶﾗ/VP200		2.2	2.2	2.3	2.4	2.6	3.0	3.5	3.9	4.2
処理画素数/秒		277885	276105	273745	266784	251713	239182	294444	185599	133255
処理多角形数/秒		2162	2107	2022	1906	1754	1492	2233	3122	3749
初期多角形数/秒		4385	4280	4096	3868	3537	3035	4821	7420	8475

註．①処理時間の1、2、3はそれぞれ初期処理、隠れ面消去処理、セグメント・ブロックの作成処理等の時間である。

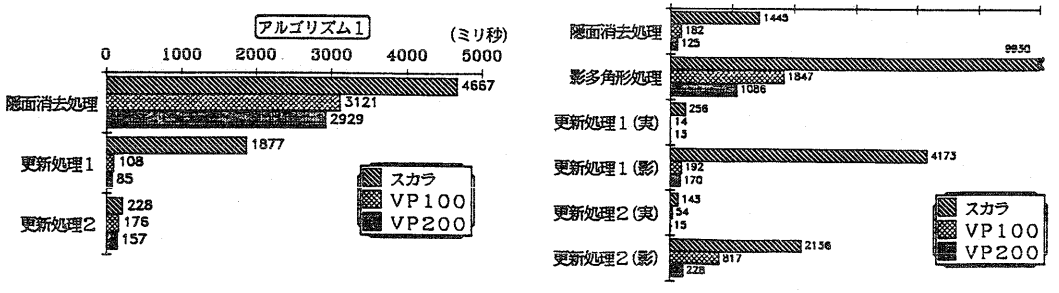
表3．アルゴリズム2（付影処理なし）

スカラ(VP100)		c300	c250	c200	c150	c100	c50	sp5	sp8
処理時間	1	201	168	135	101	68	35	324	224
	2	1813	1599	1403	1173	946	709	1300	1042
	3	362	322	282	233	194	151	413	320
	合計	2377	2089	1820	1506	1208	895	2036	1587

VP100		c300	c250	c200	c150	c100	c50	sp5	sp8
処理時間	1	49	41	34	26	18	11	58	40
	2	251	227	205	177	148	119	183	167
	3	116	98	82	62	46	28	115	83
	合計	415	366	321	265	212	158	355	290
ﾍﾞｸﾄﾙ処理時間		270	240	210	180	150	120	240	200
ﾍﾞｸﾄﾙ化率		93.9%	94.0%	93.9%	94.3%	94.9%	95.8%	94.3%	94.3%
ｽｶﾗ/VP100		5.7	5.7	5.7	5.7	5.7	5.7	5.7	5.5

VP200		c300	c250	c200	c150	c100	c50	sp5	sp8
処理時間	1	41	34	28	21	14	8	48	34
	2	152	139	127	110	94	77	116	106
	3	100	85	71	54	39	23	100	73
	合計	293	259	226	185	147	108	265	212
ﾍﾞｸﾄﾙ処理時間		180	170	150	130	110	90	170	140
ﾍﾞｸﾄﾙ化率		95.2%	95.8%	95.8%	96.4%	96.9%	98.0%	95.3%	95.5%
ｽｶﾗ/VP200		8.1	8.1	8.1	8.2	8.2	8.3	7.7	7.5
処理画素数/秒		388984	374554	355914	336462	290125	219472	168603	100457
処理多角形数/秒		3026	2859	2629	2404	2021	1369	3179	2827
初期多角形数/秒		6138	5806	5325	4879	4076	2785	7555	6389

註．①処理時間の1、2、3はそれぞれ初期処理、隠れ面消去処理、セグメント・ブロックの作成処理等の時間である。



註．①アルゴリズム1の隠れ面消去処理は影多角形処理を含む。
②更新処理1は、稜線の更新処理である。
③更新処理2は、セグメントの進入、退出処理である。

図3．影多角形法（立方体300個）

表4．アルゴリズム1（影多角形法）

スカラ(VP100)		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	566	473	382	283	188	95	579	370	270
	2	6773	5844	4648	3694	2467	1304	2989	1314	931
	3	417	351	290	227	167	105	376	229	172
	合計	7756	6668	5320	4204	2822	1504	3944	1914	1373

VP100		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	147	121	100	72	50	25	117	71	53
	2	3405	2898	2318	1873	1303	657	1224	578	442
	3	260	215	174	131	94	51	185	100	71
	合計	3811	3234	2591	2076	1447	733	1526	749	566
ﾍﾞｸﾄﾙ処理時間		1320	1120	890	710	500	260	620	300	230
ﾍﾞｸﾄﾙ化率		67.9%	68.3%	68.0%	67.5%	66.4%	68.5%	77.0%	76.5%	75.5%
ｽｶﾗ/VP100		2.0	2.1	2.1	2.0	2.0	2.1	2.6	2.6	2.4

VP200		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	129	107	86	65	44	23	103	63	47
	2	3171	2702	2159	1744	1202	601	1146	545	423
	3	227	190	152	114	77	39	163	85	59
	合計	3526	2999	2397	1923	1322	664	1413	693	528
ﾍﾞｸﾄﾙ処理時間		1200	1020	810	650	450	230	570	280	210
ﾍﾞｸﾄﾙ化率		70.0%	70.3%	70.2%	69.7%	69.1%	71.2%	78.6%	78.4%	76.8%
ｽｶﾗ/VP200		2.2	2.2	2.2	2.2	2.1	2.3	2.8	2.8	2.6
処理画素数(実)/秒		32366	32291	33489	32315	32345	35743	116765	64479	40311
処理多角形数(実)/秒		252	246	247	231	225	223	886	1216	1134
処理画素数(実+影)/秒		492723	492145	493632	482981	465003	448890	536287	278862	219356
処理多角形数(実+影)/秒		751	736	734	689	668	666	1306	1533	1541
初期多角形数(実)/秒		511	501	501	469	454	454	1912	2889	2564

註．①処理時間の1、2、3はそれぞれ初期処理、隠れ面消去処理・影多角形処理、セグメント・ブロックの作成処理等の時間である。

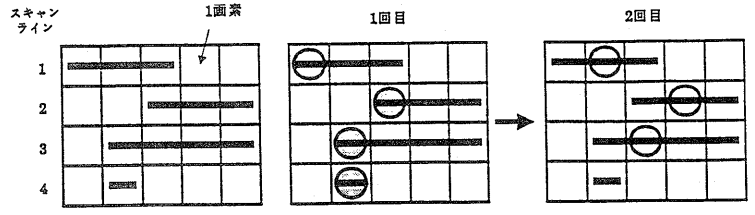


図2．アルゴリズム2

表5．アルゴリズム2（影多角形法）

スカラ(VP100)		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	563	471	375	283	187	94	576	370	269
	2	1843	1658	1423	1194	956	714	2865	1297	1045
	3	16231	13663	10845	8454	5508	2602	4305	1651	1200
	4	994	859	725	579	445	308	850	549	446
	合計	19631	16651	13368	10509	7096	3718	8596	3867	2960

VP100		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	144	120	96	73	49	26	116	72	54
	2	250	226	205	176	148	119	383	184	168
	3	2857	2463	1948	1545	1036	536	959	345	273
	4	360	304	248	188	131	75	265	153	117
	合計	3611	3113	2497	1982	1364	755	1723	754	612
ﾍﾞｸﾄﾙ処理時間		2620	2260	1820	1460	1020	560	1250	490	390
ﾍﾞｸﾄﾙ化率		95.0%	94.9%	94.9%	95.0%	95.1%	94.8%	94.5%	93.2%	92.5%
ｽｶﾗ/VP100		5.4	5.3	5.4	5.3	5.2	4.9	5.0	5.1	4.8

VP200		c300	c250	c200	c150	c100	c50	sp27	sp5	sp8
処理時間	1	126	105	84	63	42	21	102	62	46
	2	152	139	127	110	94	77	233	116	106
	3	1475	1272	1013	806	543	281	504	175	134
	4	306	258	210	159	110	61	224	130	99
	合計	2059	1774	1433	1138	789	440	1063	483	385
ﾍﾞｸﾄﾙ処理時間		1710	1480	1200	960	670	380	840	370	300
ﾍﾞｸﾄﾙ化率		98.2%	98.2%	98.3%	98.3%	98.3%	98.4%	97.4%	97.1%	97.1%
ｽｶﾗ/VP200		9.5	9.4	9.3	9.2	9.0	8.4	8.1	8.0	7.7
処理画素数(実)/秒		55424	54571	56011	54622	54212	53851	155209	92520	55358
処理多角形数(実)/秒		431	416	414	390	378	336	1177	1744	1558
処理画素数(実+影)/秒		843730	831707	825609	816397	779359	676296	712851	400137	301236
処理多角形数(実+影)/秒		1287	1243	1228	1165	1119	1003	1736	2200	2117
初期多角形数(実)/秒		875	846	838	792	762	683	2541	4146	3521

註．①処理時間の1、2、3、4はそれぞれ初期処理、隠れ面消去処理、影多角形処理、セグメント・ブロックの作成処理等の時間である。