

# 隠面消去と濃淡表示について

山内 平行・大須賀 節雄

On the Hidden-surface Problem and Half-tone Computer Graphics Representations

By

Hiroyuki YAMAUCHI and Setsuo OSUGA

## 摘 要

図形処理の一分野として、三次元物体の表示に関する問題、特に物体部分の可視性を求める問題がある。

我々はこの問題を多面体の場合について扱い、蓄積型ディスプレイ装置を HITAC-10 小型電子計算機の図形出力装置として用い、結果を濃淡画像で表示するプログラムを作成した。

また、これと関連して機械式画像入力装置からの写真などの画像入力データと上記プログラムによって計算機内部に生成された形状データとを混合して表示するプログラムも作成されているのでこれらの手法、結果について報告する。

## 1. ま え が き

過去数年来、各種の三次元物体の隠線消去あるいは可視面を求めるアルゴリズムが提案されている。これらのアルゴリズムは例外はあるが大きく分けて二つのカテゴリに分類される。すなわち、三次元空間で直接稜線と面との前後関係、面と面との前後関係などを数学的に処理して可視性を求めるアルゴリズムと、物体の透視面上の像空間内での情報に原空間の軸の情報を加えて可視性を求めるアルゴリズムとである。

いずれにしろ可視性を求めるアルゴリズムでは、各種図形入力データの内部形式 (data structure)、時間短縮のためのソーティング (sorting) およびコヒーレンス (coherence) の情報の利用の方法が重要な問題のひとつとなっている。

我々の用いた手法は、上のべたカテゴリの後者に属するものであり、面を物体表現の基本単位とし、面のプライオリティ (priority) の概念を定義し、面分割を行なっているのを特長としている。視線上の任意の二点について視点に近い点を可視性という意味からプライオリティが高いといい、任意の二面についてそれらが凸多角形であれば相貫していない限り、その二面内の任意の点に関するプライオリティは面全体に関して保存される。したがって、任意の二面間の相対的プライオリティをマトリックス形式で記憶する方法を用いた。面が凹多角形の場合にはこれを凸多角形に分割し、相貫が起る場合は相貫線に沿って面分割するこ

とによりすべて凸多角形の問題に帰着させることとした。また、プライオリティ方式をとることによって背景写真などの画像入力データ装置を介して入力される画像情報と、形状を表わす数値データが与えられて計算機内で発生された図形情報とを合成して表示することが容易となった。

## 2. 入力データと処理手順

各種データと手続きとの関係を Fig. 1 に示す。

### 2. 1 入力データ

これは前述のように二種類ある。

- ・形状データ入力 三次元物体を定義するためのデータをデータ・タイプライタから与える。
  - (1) 頂点データ 頂点番号とその三次元座標値
  - (2) 面データ 面番号とその面の境界線データを任意の向きの頂点列で与える。

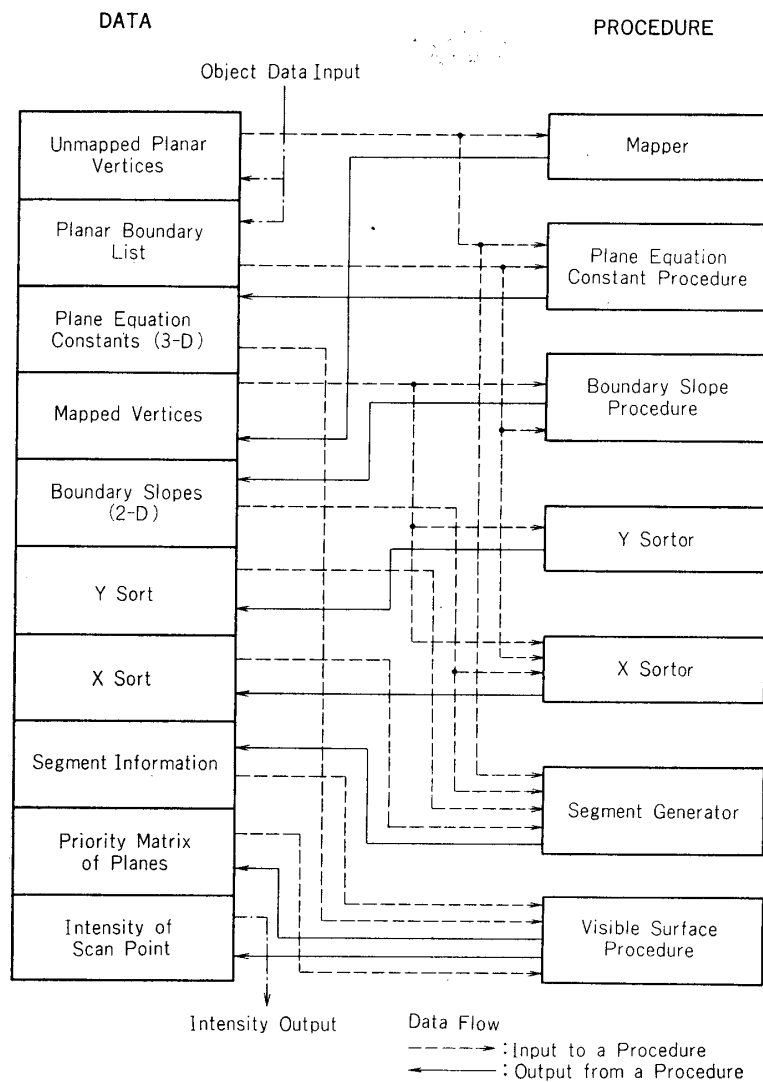


Fig. 1 Data Allocation and Procedures.

ただし、面の中に孔のあるような多重連結面の場合はその外周と内周の向きは互いに逆にし、間に\*（アスタリスク）を挿入して表現する。

(3) 物体 物体番号とその物体を構成する面番号を任意の順序で与える。

これらのデータは後述する一定の形式で記憶されるが同時に面の境界を定義する“辺”が両端の頂点のペアとして定義され、内部名がつけられる。

・パターン・データ入力 写真などの画像データは機械式画像入力装置を用いて入力する。すなわち、写真などを装置の回転ドラムに巻きつけ、それにビーム光をあて反射光を光電管で受け、このアナログ出力をA-D変換し計算機に入力する。計算機側ではプログラムによりこの入力データをさらに、適当な階調度に変換して1語に4データをパックしディスクに格納する。

## 2.2 処 理

・形状データ処理 物体に関する位置情報およびトポロジカルな情報と視点データが与えられると、プログラムはまずFig.1に示すMapperと名づけられたルーチンによって各頂点の空間座標を透視座標に変換する。問題の記述には一般に空間座標系と透視座標系および透視平面座標系の三つの座標系が定義される。したがって透視図を得るためには一般には回転を含む座標変換が必要であるが、ここではFig.2のようにこれら3つの座標系は一致している場合を扱う。するとこれらの座標系による空間座標系から透視平面座標系への変換式は(1)式で表わされる。

$$\begin{aligned} X' &= X_0 + \frac{-Z_0}{Z - Z_0}(x - X_0) \\ Y' &= Y_0 + \frac{-Z_0}{Z - Z_0}(y - Y_0) \end{aligned} \quad (1)$$

ここで $(x, y, z)$ は物体上の点の位置、 $X', Y'$ は透視面上の位置、 $(X_0, Y_0, Z_0)$ は視点の位置である。

また、各平面の方程式および透視面上での各面の辺の方程式が計算され、のちの面分割をするSegment Generatorと可視面を求めるVisible Surface Procedureによって参照される。空間内の平面の一般形はここでは $ax + by + cz + d = 0$ を使い、面が空間内の点 $(x_1, y_1, z_1)$ 、

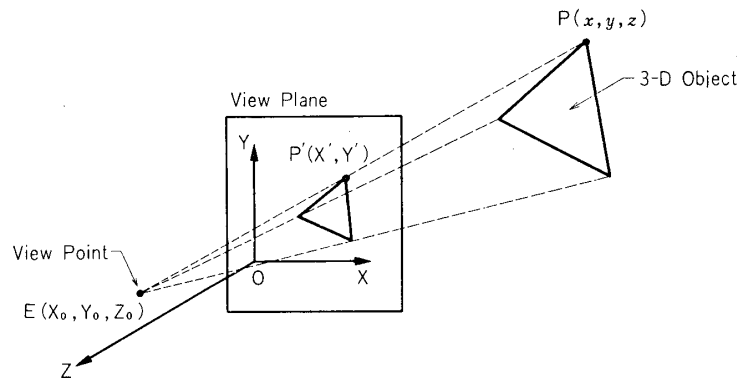


Fig. 2 Co-ordinate System.

$(x_2, y_2, z_2)$  および  $(x_3, y_3, z_3)$  で指定された時, (2)式によって各係数が計算される.

$$\begin{vmatrix} x-x_1 & y-y_1 & z-z_1 \\ x_2-x_1 & y_2-y_1 & z_2-z_1 \\ x_3-x_1 & y_3-y_1 & z_3-z_1 \end{vmatrix} = 0 \tag{2}$$

次に透視変換された各頂点の座標値の座標を小さい順に並べて YSORT テーブルに登録する. プログラムはこのテーブル内の  $y$  座標値をとり,  $x$  軸に平行な直線と面の各辺との交点列を求める. その際, 辺の中間で交ったか, 端点で交ったか, 一致したかによって後の処理が異なるのでそれを判定し, この識別コードと, 交った辺の番号, その交点座標を XSORT テーブルに交点座標の  $x$  の値の小さな順に登録する. 同一の  $x$  の値をもつ交点 (すなわち, 辺の端点で交わる場合) については, 面を左に見て反時計まわりの順序で辿った時に先に生ずる辺を先にする (Fig. 3 参照).

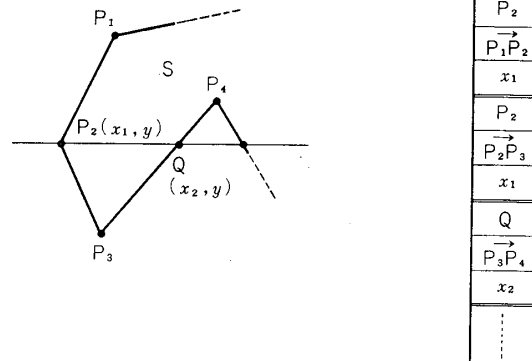


Fig. 3 XSORT.

Segment Generator はこの XSORT テーブルのデータを用いて与えられた入力面に凹頂点があるかどうかを確認する. ここで凹頂点は交わる二辺のなす内角が  $180^\circ$  を越すものを指す. 凹頂点がある場合は求まった交点列のタイプ (Fig. 4 参照) を求め, それに従って面分割する. 詳しいことは次の第節でのべる. ここで凹頂点であるか否かおよび交点のタイプは直接にその角度を計算して求めるのではなく, 各頂点座標間の大小関係および辺の傾きから論理的に求めている.

すべての面が凸多角形に分割され, 各分割された面の情報を表わす Segment Information が作成されたら, Visible Surface Procedure はこれと Priority Matrix から可視面を求め, ディスプレイルーチンに表示データを渡す.

・画像データ処理 写真などの画像データは, 尺度変換や明暗変換によって適当なレベルに量子化し, これをプログラムによって発生された図形データの背景として表示する. この図形データとの重ね合わせの問題は, すでにのべた面のプライオリティという考えをいかし,

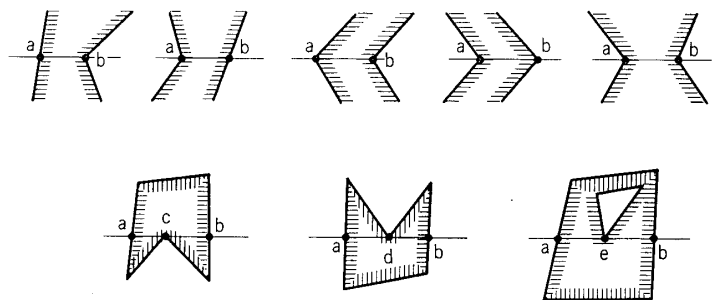


Fig. 4 Type of Cross Points.

背景に最も低いプライオリティを与えることによって容易に行なわれる。

### 3. 面分割とデータ構造

#### 3.1 面分割

面分割は面が凹頂点を含む時、その凹頂点を通る  $x$  軸に平行な直線で切る場合と、面同志の相貫があるとき相貫線に沿って切る場合とがある。まず最初に前者の場合を考えると、一般の凹多角形（多重連結面を含む）は Fig. 5 で示される基本図形の複合形として現われる。 $x$  軸に平行な直線で下から上に順次分割してゆくので、図形の向きにより処理が異なる。こうする理由は表示をしながら処理する方法をとることによって不要な処理を最小限にするためである。これらの単純図形の分割ルールから基本ルールを確立し、一般の図形をこれらの基本ルールを再帰的に使用することにより凸多角形に分割してゆく方法をとる。

図形の分割は、面を構成する頂点群の中に交点を新しい頂点とみなして含め、これを図形を左にみて反時計まわりに現われる順序で並べた頂点列内の交点の分布にもとづいて行なわれる。基本ルールはこのような頂点列に適用され、面の一部を切り取って二つ以上の新しい頂点列に変換する変換ルールであり、図形頂点群の有するシンタクティックな性質から後述する一定の条件のもとで次のように表わされる。

- (a)  $XaAb \rightarrow aAb, abX$
- (b)  $XaAcBb \rightarrow aAc, cBb, abX$
- (b')  $XaAbYd \rightarrow aAb, abX, dbY$
- (c)  $XaAb*Ye \rightarrow aAb, aeYebX$  (3)

ここで小文字の  $a, b, c, d, e$  は Fig. 5 で示された分割線上の交点を表わし、大文字の  $A, B$  は交点を含まない頂点列、 $X, Y$  は任意の頂点列である。右辺の、(カンマ)で区切られた列は、分割された面の頂点列を示す。また \* (アスタリスク) は図形内に内孔がある時外周と内周との境界を示す記号である。

なおここでの記述に直接関係はないが、面分割にこのような手法を用いたのは図形の言語

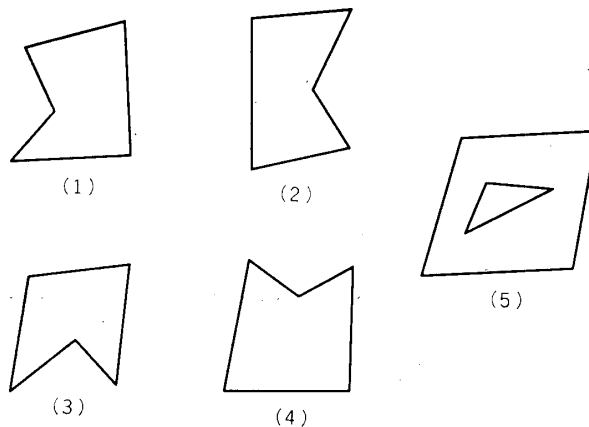
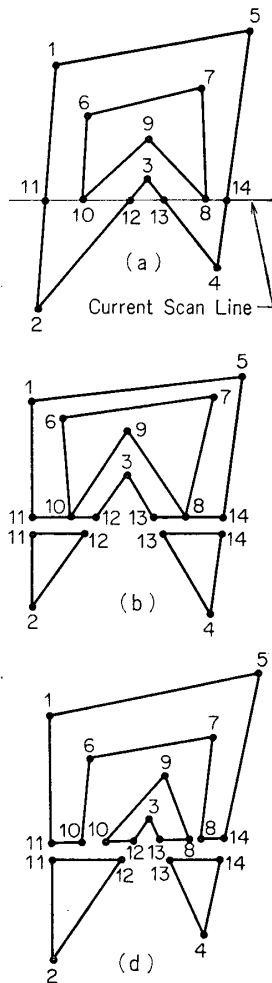


Fig. 5 Simple Polygons.



		a	b	a	b			e	e
PBS	1	11	2	12	3	13	4	14	5
IPL	11	10	12	13	8	14			

(a)

		a	b	a	b			e	e
PBS	1	11	0	12	3	13	0	14	5
IPL	11	10	12,13	8	14				
LWS	11	2	12,13	4	14				

(b)

			b	a				e	e
PBS	0	0	12	3	13	0	0	0	8
IPL	0	10	12	13	8	0			
UPS	1	11	10	6	7	8	14	5	
LWS	11	2	12,13	4	14				

PBS	All Zero
IPL	All Zero
UPS	1 11 10 6 7 8 14 5,12 3 13 8 9 10
LWS	11 2 12,13 4 14

(d)

\*a, b, c, d, e: Type of Cross Points

Fig. 6 An Example of Plane Division.

的な表現と処理の研究の一端として行なわれたもので、それは形状をシンタックスに基づいて一定の標準形式で表わすことを基本とする。本方法は面分割という目的にたいしてこれに近い方法を用いているので、参考のため一般の図形の場合を式で示すと

$$S = \prod_r [AaB(cC)^{ir}b(Dd)^{jr} \prod_s [*(Ee)^{ks}] \prod_t [*(Ff)^{lt}] ] \quad (4)$$

のような形で表現できる。ただし、 $ir, jr, ks, lt$  は 0 もしくは正整数、 $(cC)^{ir}$  等は  $cC$  という形式が繰り返す  $ir$  回生ずることを示し、 $\prod$  等は [ ] 内全体が繰り返されることを示す。

プログラムによる実際の分割の例が Fig. 6 に示してある。すなわち、Segment Generator は XSORT テーブルから分割線との交点列をまず IPL と名づけられるテーブルに転送し、さらに交点を新しい頂点とみなして頂点列に挿入し交点のタイプを示すコードを含めてテーブル PBS に記入する (Fig. 6, (a))。次にこの PBS と IPL を参照し、これらに(3)式を繰り返し適用する。分割線の下側にある分割面 (下位分割面と呼ぶ) を順次切りとって、これをテーブル LWS に登録する (Fig. 6, (b))。この分割面はすべて凸多角形になってい

る。LWS に登録された分割面を定義するために用いられた頂点列は PBS から削除する。ただし、交点番号に関しては 2 度用いることにする。2 度参照されたら PBL および IPL からそれを削除する。このように下位分割面を切りとると、面の残りの部分は必ずしも閉じた形にならないので、分割線の上側にある分割面（上位分割面）にたいする頂点列の標準化が必要である。その方法を Fig. 6 を参照して説明する。以下の記述中（ ）内は Fig. 6 (b) の場合である。IPL を左から見ていき、最初にゼロでない交点番号 (11) を出発点とし、次にゼロでない交点番号 (10) を求め、PBL 内のその交点番号のある場所にとぶ。そのあと、PBL を循環列として右へたどっていき、これ以後交点となっている頂点番号 (8) に最初にあうまで進む。ただし、アスタリスク記号があると循環列はここで分割されるからアスタリスク記号に達したらその循環列の先頭に戻りそこから右へ辿っていく。交点に出会ったら出会った交点番号を IPL 内にさがし、さらに IPL 内でその交点番号の次の（右方の）ゼロでない交点番号 (14) を求め、PBL 内のその交点番号のある場所に再びもどる。以下同じことを繰り返し出発点にもどるまでの頂点列を上位分割面とする。この例では結局上位分割面、 $\overline{11-10-6-7-8-14-5-1-11}$  が得られる。⌈ は IPL 内、⌋ は PBS 内のサーチを示す。これを IPL が空になるまで続ける。作成された上位分割面はテーブル UPS に登録してゆく、このようにして作られた上位分割面のおののに対して分割線を順次求め、いまのべた手続きを繰り返して最終的に与えられた面をすべて凸多角形に分割する。

次に相貫線に沿って面分割する場合を考える。相貫の状態は Fig. 7 に示すように (a), (b) 2 つの場合が考えられる。(a) の場合は次の (5) 式の基本ルールで面分割する。

$$A_p' B_{p''} \longrightarrow p' p'' A, p'' p' B \quad (5)$$

Fig. 7 (a) の例では  $A = p_3 p_4$ ,  $B = p_1 p_2$  である。(b) の場合は、相貫線を延長し、(a) の場合に帰着させる。相貫の判定は面のプライオリティ判定の際同時に行なう。

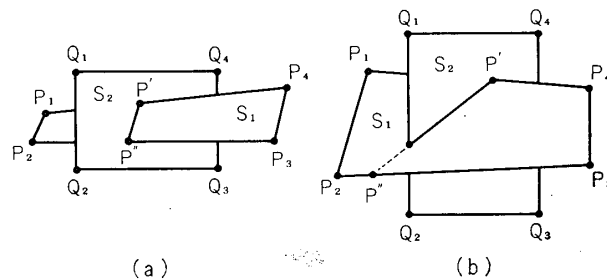


Fig. 7 Type of Intersecting Planes.

### 3. 2 データ構造

以上のべた方法により、与えられた面情報は、Fig. 8 に示すようなデータ構造に記憶される。面の分割が進行すると共に分割された面データは元の面データから Fig. 8 (a) の例のような構造を形成してゆく。この図の第 1 の例に見られるように凸平面  $S = S_0$  をさらに  $S_{00}$ ,  $S_{01}$ ,  $S_{02}$  に分割しているのはデータ記憶形式をととのえるためである。

Fig. 8 (b) はこのデータ・ブロックの内容を示し、A ブロックの  $y$ -min,  $z$ -max は元の

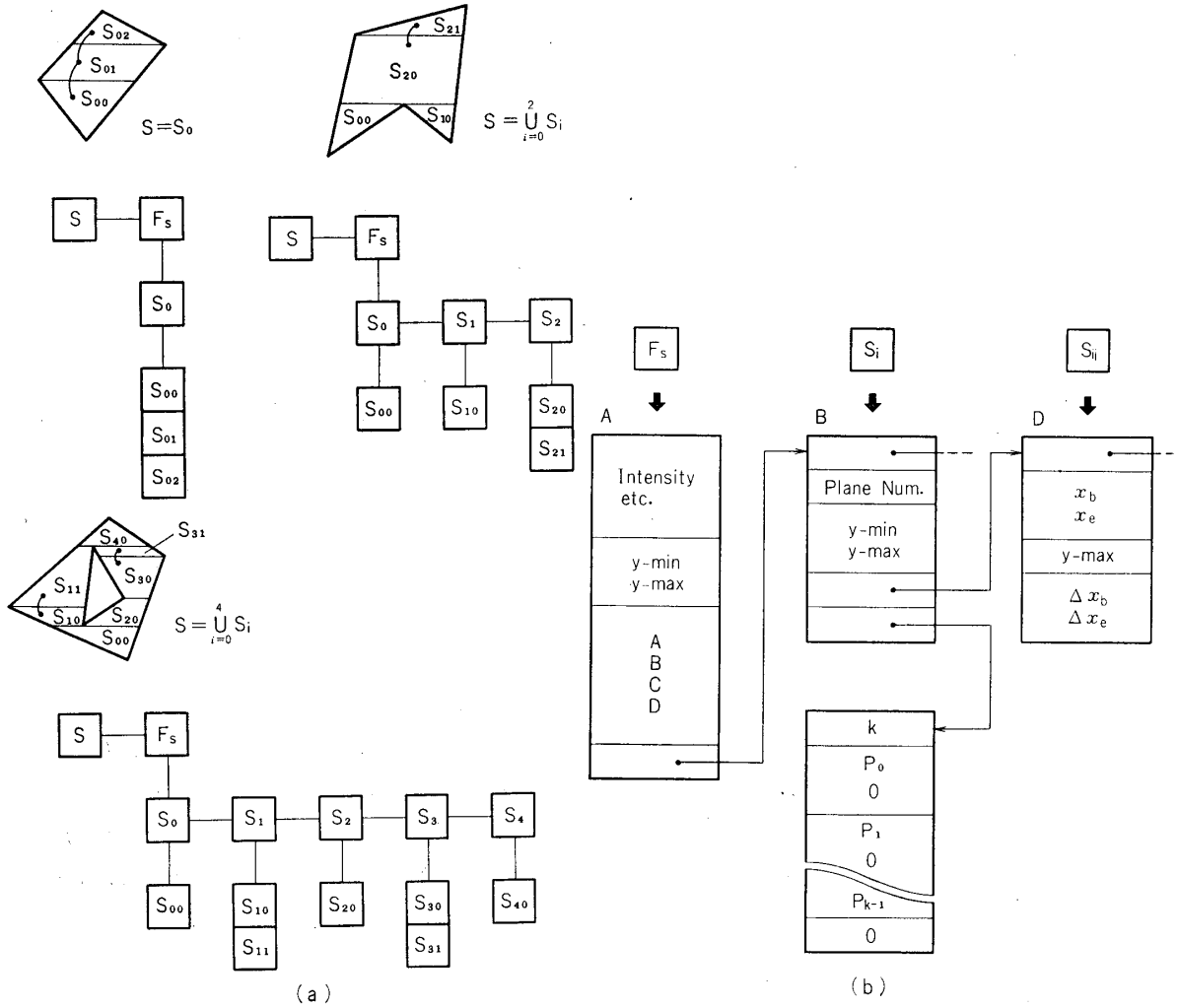


Fig. 8 Data Structure. (Cont.).

(分割前の) 入力面の頂点の透視平面上の  $y$  座標の最小値, 最大値を表わし, A, B, C, D は平面方程式の 4 つの係数を表わす: B ブロックの  $y$ -min,  $y$ -max は各分割面の  $S_i$  ごとくその頂点の最小値, 最大値を表わし, 面番号は分割面番号を表わす. 最小, 最大値を記憶するのは表示やプライオリティの判定の効率をあげるためである. C ブロックは各分割面の頂点番号列を表わす.  $k$  は頂点の数である. D ブロック・チェーンは前述のように分割面をさらにデータ記憶の便宜上細分したもので, スキャン・ラインとの交わりの始まりと終りを示す  $x_b, x_e$  データ, 細分の上限の  $y$  座標  $y$ -max それに, スキャン・ラインが 1 つ進んだときの増分  $\Delta x_b, \Delta x_e$  データが入る.

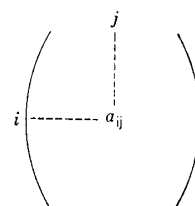
#### 4. 表 示

表示用の図形出力装置は, ソニー・テクトロニクス社製蓄積型 CRT 661 タイプを使用し



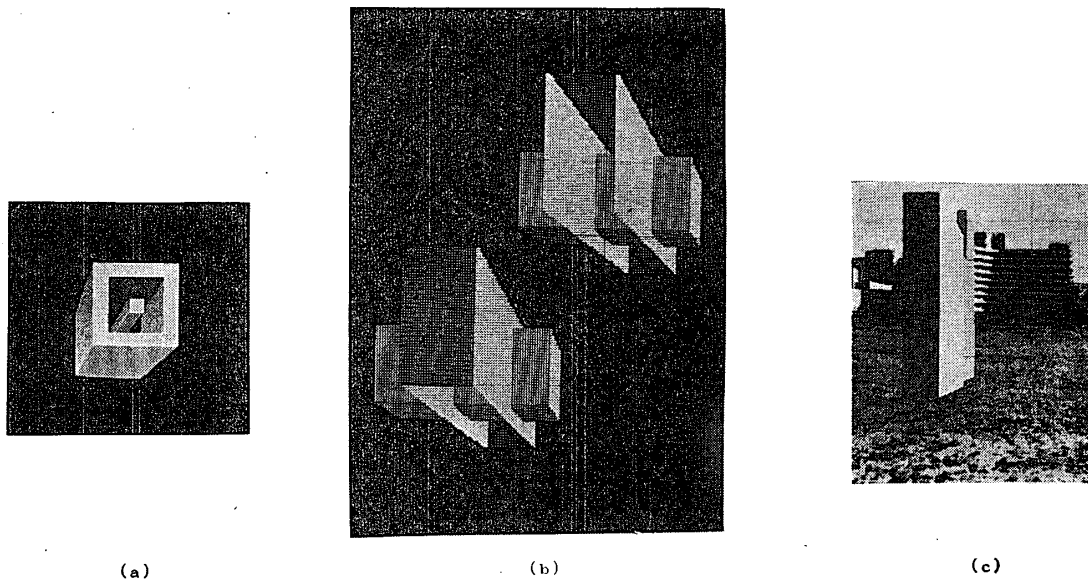
ている。各ラスタ・ユニットにおける輝度は、1か0の2値であるので中間色表示を行なうため、すなわち階調を増すには、一点の情報量をふやす必要がある。そこで透視面上の1点をさらに表示面では3×3または4×4の矩形領域に対応させこの領域内の輝点の数と分布によって中間色を出すようにした。この表示パターンはいろいろと実験してみて経験的に決められたが、本来は人間の眼の特性（光の刺激が幾何的に増大するとき、人間の眼のそれを感覚する強さは算術的に増大する）と装置の分解能とを考慮して厳密に決められるべきである。現在、各点の階調度は同一面内では一様として表示することとした。

出力は濃淡情報で行なわれるから、全透視面をスキャンし、各点ごとにその点を透視面内で含む面をすべて求め、その中で最もプライオリティの高い面の階調度を出力データとするわけであるが、一般に、1つのスキャン・ポイントの近傍内では最もプライオリティの高い面は変ることが少ない性質、すなわち、area-coherence をとり入れ、スキャン・ポイントと交わる面を求める時間を短縮した。また、最もプライオリティの高い面を求めるには、面の視線からの距離の大小によって決めなければならないが、一度、ある二面の距離が計算され優先順位が決まったら、再びそれらの面の間でのプライオリティを計算（距離計算）しなくて済むように、これをプライオリティマトリックスに書き込んでおく（Fig. 9 参照）。このように各面の組合せについて一度だけ相対プライオリティを求める



- $a_{ij}=00$ : Priority between  $i$  and  $j$  is undefined.
- $a_{ij}=10$ : Priority of Plane  $i$  is lower than that of Plane  $j$ .
- $a_{ij}=11$ : Priority of Plane  $i$  is higher than that of Plane  $j$ .

Fig. 9 Priority Matrix.



(a)

(b)

(c)

手前にある建物が計算機によって生成された図である。

Fig. 10 Examples

必要があるが透視面上で重ならない面の組ではこの操作は不要なので、最初はすべての面の相対プライオリティを作ることはせず出力しながら必要の都度行なうこととしている。また相貫する面は必ず透視面上で重なる部分を有するから、この時にのみ相貫テストを行なう。

実際の表示の例を Fig. 10 に示しておく。

### 5. あとがき

我々は、簡易画像入出力装置を用いた画像情報処理の研究の一環として多面体の表示と面の重ね合わせについてのべた。ここで用いられた面分割の手法は平面のシンタクティックな性質を利用し、再帰的な手法がとれることを意図した。これは面分割という特殊な目的にたいしてのものであるが、物体形状の言語的な表現方法と、その計算機処理の問題は人間・機械間のコミュニケーション用言語の研究にとって重要な問題であり、その手法の応用を図ったものである。

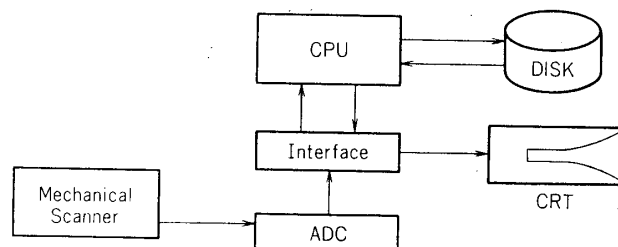
おわりに、プログラムの作成にあたって一部サブルーチンの作成およびソーステープのパンチなどに当研究室の小川由美子嬢のご協力があったことを感謝する。

1974 年 4 月 30 日 航空力学部

### 参 考 文 献

- [ 1 ] Romney, G. W.: Computer Assisted Assembly and Rendering of Solids, Dep. of Computer Science, Univ. of Utah.
- [ 2 ] Newell, M. E., Newell, R. G., Sancha, T. L.: A New Approach to the Shaded Picture Problem, Proceedings ACM National Conference, 1972.
- [ 3 ] 穂坂・木村: 立体の内部記述と立体処理の一般解法, 情報処理学会 48 年度大会前刷, p. 405-406 (1973).
- [ 4 ] 山内・大須賀: 多面体の表示と面の重ね合わせ, 情報処理学会 48 年度大会前刷, p. 425-426 (1973).
- [ 5 ] A. Rosenfeld: Picture Processing by Computer, Academic Press, 1969.

### 付 録 システム構成



•CPU HITAC-10  
コア・メモリ

16 kW/16 BIT

•CRT	ソニー・テクトロニクス	ストレージ・チューブ
	画面の大きさ	21 cm×16.3 cm
	分解能	400 pt.×300 pt.
	蓄積時間	約 15 分以上
	消去時間	500 ms 以内
	入力電圧	X軸, Y軸ともフルスケール 1 V
•ADC	A-D 変換器	ソニーテクトロニクス社製
	入力電圧範囲	0~+10 V
	入力インピーダンス	10 kΩ
	変換時間	50 μs
	分解能	純 2 進 8 BIT
•DISK	ディスク装置	IOMEC 社製
	容 量	10 Mbit
	アクセス・タイム	トラック間 30 ms
		平均ランダム移動 60 ms
		平均回転遅延 20 ms
•Mechanical Scanner		機械光学式画像入力装置
	回転ドラム	82.758 rpm (0.75 sec/rev)
	回転方向	5.87 pt/mm
	軸方向画像密度	7.4 l/mm