

三次元物体の隠線消去問題および陰影表示問題

西田 友 是
(東洋工業株式会社自動車企画部)

1. はじめに

最近、計算機とその周辺機器の発達に伴い、計算機による図形処理が重要となってきた。三次元物体を二次元平面上に表示するには透視図 (Perspective drawings) が直観的で判り易い。このとき投影された図形は実際には見えない線までが表示され、いわゆる wire frame となり、複雑で見づらいものとなる。そこで見えない線分を消去する隠線消去問題 (Hidden line problem) が生じる。さらに real に表現するには濃淡をも表示する Half tone representation が必要で、これにより陰影が判り、より立体感を得ることが出来る。また人間が三次元物体に近づいた際、物体の一部が視野からはみ出る。このはみ出た図形を消去して表示する Windowing problem も必要となる。

建築物、機器の設計結果、三次元関数の計算結果などに対してこれらの処理をして表示することにより、それらの変更、選択が容易に行なえ、いわゆる CAD (Computer Aided Design) を行なうことができる。これらの問題について研究したのでその理論の概略と応用例を示す。

2. 凸多面体集合の Hidden Line Problem

隠線消去問題については多くの解法が発表されているが、筆者は多面体および平面をそれぞれ凸多面体および凸多角形の集合として取り扱うことにより隠線消去のアルゴリズムの簡易化と高速性を計った。さらに多面体上に描かれている図柄および空間中の線分の処理も行なえるようにした。

使用頻度の多い形状（平行六面体、円柱等）の凸多面体は標準化し、必要なパラメータのみを入力することで、従来の方法に比べて、データの作成が早く、かつデータ・エラーも減少し、検討の対象となる凸多面体の大きさ、位置の変更も容易である。また同形状ものが複数個並ぶ場合は、その変化分と個数を入力することで自動作成することができる。

2.1. データ構造

凸多面体、面および頂点は階級構造とし、各々のデータ構造は配列形式とする。頂点は三次元座標で、面は物体外部から見て右回りの頂点番号で定義する。なお面上に図柄のある場合はその図柄の属する面番号と、図柄の三次元座標の並びで与える。空間中の線分は始点と終点の三次元座標で与える。

2.2 アルゴリズム

多面体の面の法線ベクトルが視点を向いている面を表の面 (front face)、反対の向きを裏の面 (back face) とし、^{表と}表の面の_{内線}の_{内線} (inner edge)、表と裏の面の_{内線}の_{内線}を輪郭線分 (con-

tour edge) . そのつながらりを輪郭線 (contour line) と定義する。凸多面体を考えくいるので輪郭線は透視図面上で必ず閉じた凸多角形であり、右回りに定義することができる。ここで内線と輪郭線分が見える可能性のある線分である。三個の凸多面体の輪郭線の間には図1.に示す三つの場合が存在し、それぞれの場合に応じた処理を必要とする。

i) 輪郭線が交わらない場合

: 不可視部分は生じない。

ii) 輪郭線が交わる場合: 一

方の輪郭線、内線に不可視部分を生じる。この場合は輪郭線の交点の三次元上の関係で凸多面体の前後関係を求め、手前の凸多面体の輪郭線と隠される凸多面体の内線との交差判定を行ない、不可視線分を求めぬ。

iii) 一方の輪郭線が他方の輪郭線を囲む場合: 凸多面体 V_B の任意の頂点とそれを含む V_A の面との関係から V_A , V_B の前後関係を求める。 V_A が手前なら V_B は不可視であり、 V_B が手前なら V_A の輪郭線と V_A の内線との交差判定を行ない、不可視線分を求めぬ。

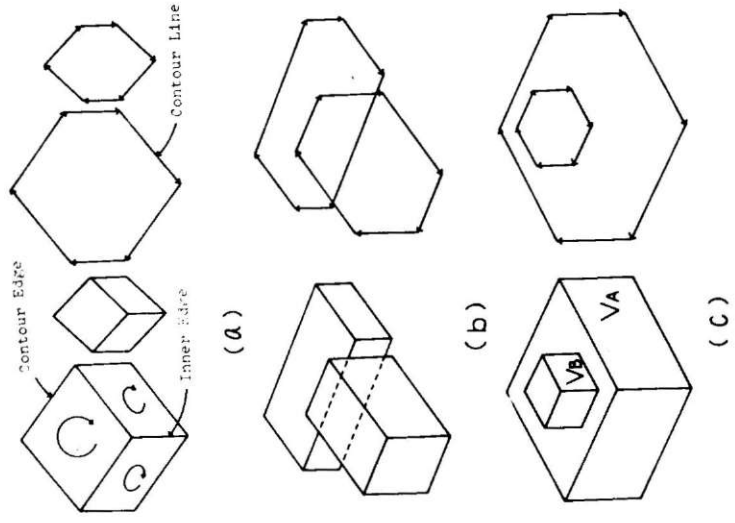


図1. 凸多面体相互関係

な和図柄は内線と同様に取り扱うことができる。次に計算手順

に従って述べる。

まず各頂点と透視面上へ点投影により座標変換をよる。次に面
への向きを求め、これは面の構成三頂点が透視面上で右回りが否
かで表の面か裏の面かを判定するし式(1)で判定し。この面の向き
により内線、輪郭線とよる輪郭線分を輪郭線とよる。

輪郭線の相互関係の求め方は次のようである。頂点 P_i 、 $i=1 \sim n$
よりなる凸多面体 V_i の輪郭線と、 P_j ($j=1 \sim m$)よりなる凸多面体
 V_j の輪郭線を考える。

$$f_{ij} = \begin{array}{c|c|c} s_j & j & 1 \\ \hline -x_i & y_i & 1 \\ \hline x_{i+1} & y_{i+1} & 1 \end{array} = s_j \cdot x_j + v_j \cdot y_j + c_j \dots \dots (1)$$

とよると線分 $[P_i, P_{i+1}]$ の右半平面上に P_j が存在すると f_{ij} は
正である。いま V_i の線分に対する V_j の頂点の位置関係とよる f_{ij}
と要素とよる行列を考えると次の性質がある。

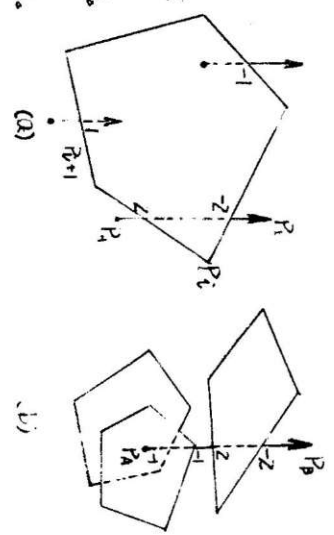
- i) $f_{ij} \leq 0$ ($j=1 \sim m$) ; V_j は線分 $[P_i, P_{i+1}]$ の左半平面上
に存在するから独立である(図1(a))。
- ii) $f_{ij} \geq 0$ ($j=1 \sim m$) ; 線分 $[P_i, P_{i+1}]$ は V_j と交差しない。
- iii) $f_{ij} \geq 0$ ($i=1 \sim n, j=1 \sim m$) ; V_i が V_j を囲む(図1(c))。

f_{ij} にも同様な性質がある。i), iii)以外の線分は交差する可能性
があるからii)以外の線分に対して次の式により交差する線分とよる
交点を求める。

$$d_{ij} = \{ \text{syn}(f_{ij+1}) - \text{Syn}(f_{i,j}) \} \cdot (\text{Syn}(f_{j,i+1}) - \text{Syn}(f_{j,i}))$$

$$= \begin{cases} +1 & (P_i, P_{i+1}) \text{ と } (P_j, P_{j+1}) \text{ が交差,} \\ -1 & \text{線分と頂点が重なる,} \\ 0 & \text{頂点と重なる,} \\ & \text{独立.} \end{cases} \quad \dots\dots (2)$$

以上の交差する線分はわかるが、実際にはどの部分の線分が可視であるかは求めていない。それゆえのようにして判定する。



凸多面体の輪郭線と線分との 図2. 不可視部分の判定

交点はたかたか2個で、図2.に示す三通りしかない。また輪郭線は右回りであるから、交差二線分のベクトル積で交点のどちら側が不可視であるが判る。一本の線分は他の輪郭線とも交点を持た得るから、交点の個数に従って出入り指標を図2.に示すように、±1, ±2で与えておき、表示する際は-1の個数をその線分の始点 (Pn) の不可視度数 (order of invisibility) とし、交点を通過することに入りの指標を加えると、度数が零の所から可視部分となる。

3. Windowing Problem.

透視面上での図形は視点を中心とした点投影によって得られるが、この際視点より後方の頂点は透写像されるので視点より

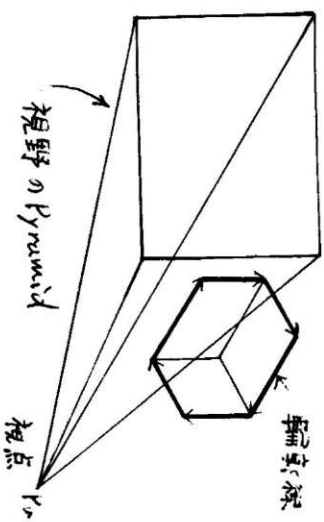


図3. 視野のピラミッド

後方の頂点を除去が必要がある。実際には人間の視野は限られてい

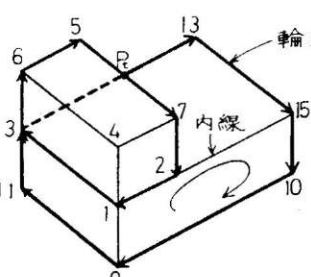
るから、視野に含まれるもののみ処理することが好ましい。いま視野を中心とするピラミッドと考える。(図3.参照) まずこの視野のピラミッドと凸多面体の相互関係を求める。すなわちピラミッド内に凸多面体が含まれるか、交差するか、外部かを求める。これはピラミッドの構成面と凸多面体の三次元上での輪郭線との関係で求める。外部の凸多面体は取り除くが、交差する凸多面体はピラミッド内の凸多面体に切断する。実際には輪郭線と内線を再構成すればよい。以上の処理後2.の方法と同様に透視面上での凸多面体の相互関係を求め Hidden line 処理を行ない表示する。

4. 凸多面体集合の Half Toned representation

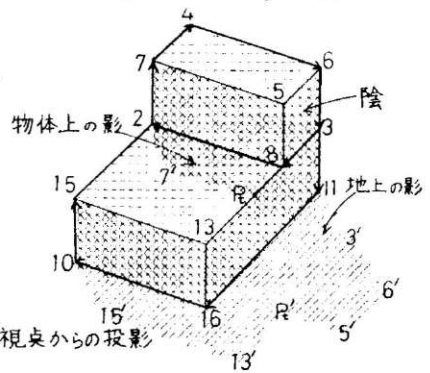
平行光線によって写し出された三次元物体の濃淡図形を表示する。2.の方法と同様に凸多面体の集合として処理することにより入力が簡単で、陰影処理および図形の走査の高速化が期待できる。

4.1. アルゴリズム

まず光源を第1の視点と仮定し、影 (shadow) の成り立ちについて考える。



(a) 光源からの投影



(b) 視点からの投影

透視面上へ座標変換し、光源に対する面の向きを求めると、裏の面が陰く

物体	頂点	面	輪郭線	内線
1	8	6	6	3
2	16	12	12	6

物体テーブル

面番号	1	2	3	4	5	6	7	...
総和	4	8	12	16	20	24	28	...

(a)

面の構成頂点	1	4	7	2	2	7	5	8	8	5	6	3	3	...
--------	---	---	---	---	---	---	---	---	---	---	---	---	---	-----

(b)

面テーブル

表 1. テータ構造

shade)の面である。面の向きを判用して各凸多面体の内線と輪郭線を求め、ここでこの輪郭線が地上の影の輪郭を構成するからその構成頂点の座標変換を行ない地上の影を求め、次に透視面上で凸多面体の輪郭線が交差するか否かを判定する。交差した部分、光源に近い凸多面体の遠い凸多面体上への影であるから、後者の各面上で影を求め記憶しておく。

また通常の視点を第2の視点と考え、透視面上の濃淡を求め、すなわち光源の場合と同様な座標変換を行ない、視点から見た透視面での面の向きを求め、内線および輪郭線を求める。次に透視面上での凸多面体の交差関係を求め、交差する場合および囲みられる場合は凸多面体の前後関係を求め記憶しておく。最後に透視面上でテレビの走査と同様に図形を走査 (scan) して各点の濃淡を求め表示する。(図4. (b) 参照) ここでscanをする水平線を、scan line とすると、このscan line 上にある凸多面体、地上の影面上の影を求めめる必要がある。しかしこれらはいずれも透視面上では凸多角形であるので容易に求めることができる。従ってこのscan line ごとに濃淡図形を出力する。

なおデフォルト構造はほぼ2. と同様であるが表1に図4の例を示す。

5. 一価関数の Hidden Line Problem

前節までは多面体について述べたが、ここでは $z = f(x, y)$ の形式で表現できる一価関数の隠線消去問題について述べる。デ-

夕構造は $(n \times m)$ の格子状とし、各格子点に z 座標を記憶する。また格子点間は直線で結ばれているものとする。

5.1 アルゴリズム

データ構造が格子状であるので、視点に近い edge (これを *leading edge* とする。) は必ず可視である。まずこの *leading edge* を表示し、次にこの *leading edge* に続く線分を次々に可視部分を求め表示する。こ

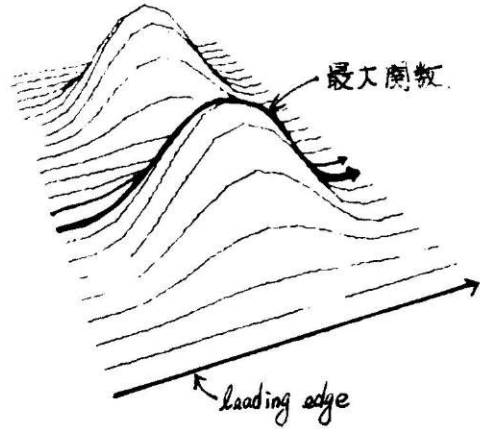


図5. 一面関数隠線消去

の際即ちアロッドした部分を示す最大関数 G を考え、その最大関数より上側にある線分のみを表示し、そのつど最大関数を更新すると、可視線分を描いたことになる。曲面の裏側が見える場合も同様に最小関数を考え最小関数より下側の線分をアロッドすればよい。この最大関数は高列で表わされているものとする。さらにこの手法を拡張し、隣り合う格子点の可視関係を利用して他の方向の要素の線分を描くことで、より曲面の形状を表現できる。

6. 曲面の Hidden Line Problem

5.と同様にデータ構造は格子状であるが、格子点に z 座標のみでなく、 y 座標を記憶することで一面関数以外の曲面及び曲面体を表現できる。すなわち格子の位相幾何学的な接



図6. 曲面の輪郭線

続により、円筒、球、トーラス、および曲面を表現することができる。

6.1 アルゴリズム

格子点に囲まれる格子面を考え、その格子面が表の面 (front face) か裏の面かを透視面上での面積ベクトルにより求め、その面の向きから内線と輪郭線を抽出する。不可視部分は必ず輪郭線と交差することによって生じることを利用して次のような処理をする。すなわち格子に従って次々と線分を進め、判定中の線分と輪郭線が交差した場合には不可視度数を(向きの面によって隠されるか)求め、不可視度数が0ならその線分は可視であるから表示する。

曲面を対象にしているから、格子面が透視面上でねじれる場合の処理などは本手法には問題がある。

7. 応用例

図7に応用例を示す。(a)~(d)は2.の方法で、(a)は同形の直方体60個を並べた例で計算時間は6.6秒、(c)は建築中の山村開港センターで(面数243, 頂点数334, 図柄数52)、(c)は windowing の処理をしたものである。(e)は本稿の中では述べてないが、凸凹面体に分割しない方法で、輪郭線、稜線の凹凸、不可視度数の性質を利用して Hidden line 処理を行なった例であり、(a)と併に名古屋大学の G.D. で表示した。(f)~(g)は3.の方法の例で重ね印字により、ラインアリナーで表示した(濃淡レベル20印階)。(g)は

平面上の図柄を示すために2.の方法も利用して、X-Yプロットで表示した病院の鳥観図で、計算時間は Half tone 処理が 49 秒、Hidden line 処理が 1.7 秒く出力表示の時間(除く)である。

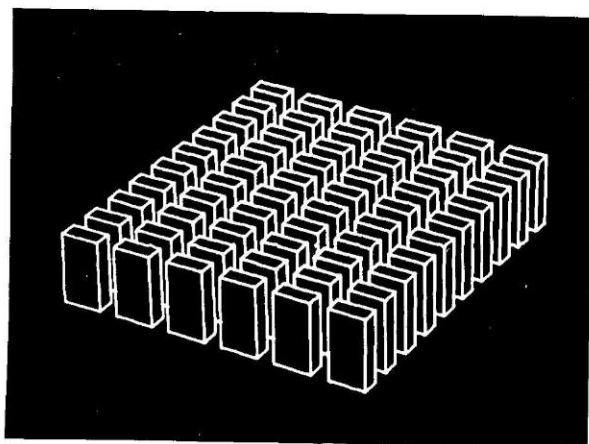
(e)~(h)は一価関数の曲面の例で、(j)は一価の曲線のみに表示した例で、(g)は碧梯山である。(l)は6.の方法の例で、トラスである。なお使用計算機は FACOM 230-60 である。

8. おわりに

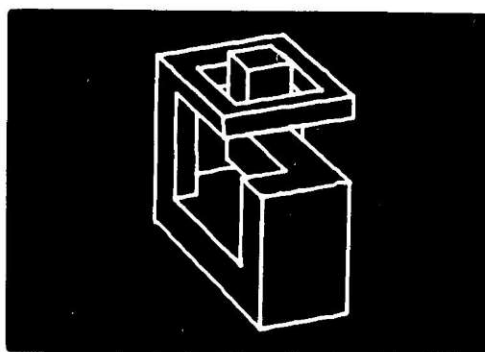
応用例でも示した如く、従来の方法に比べて凸多面体の集合として処理することは、入力データの作成時間が短く、変更が容易で計算時間も短いことから、充分実用に供し得るプログラムと思われる。特に計算式も(1)式のように簡単でしかもその値を重複利用できることと、凸多面体の輪郭線の性質を利用することで処理時間が短縮された。Half tone 問題では、出力装置が汎用のプリンターやプロットターでも満足な画質が得られることが判ったが、もっと良い出力装置を利用できるなら、速達によるコントラストや曲面のスムージングや反射なども考慮する必要があると思う。

三次元関数を透視図で表現することは、その関数の性質を直観的に判断できるし、図7、(h)の例のように地形を表示して整理後の形状を検討するのも便利である。

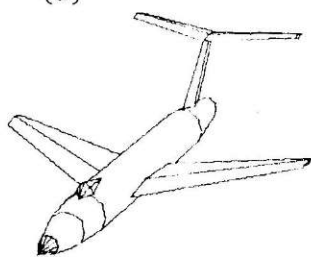
なおプログラムは Fortran を使用し、2.4.の方法では約 1500 ステップで、5.は約 200 ステップである。



(a)



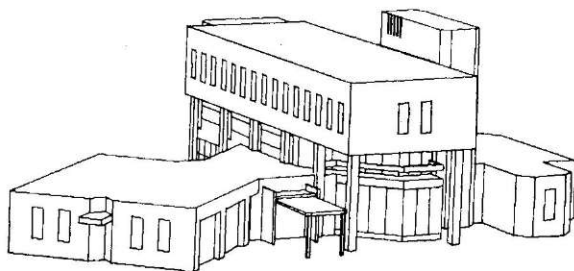
(e)



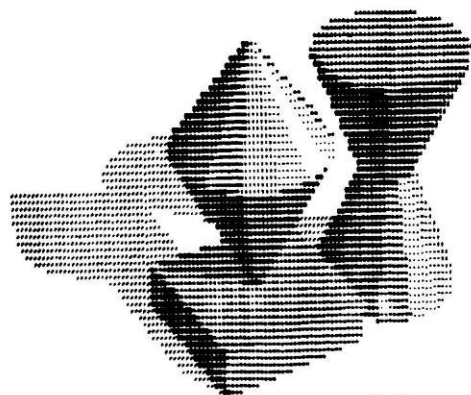
(b)



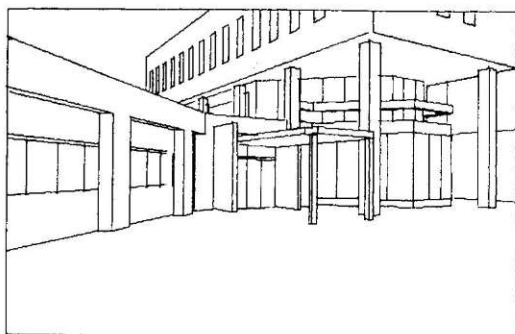
(f)



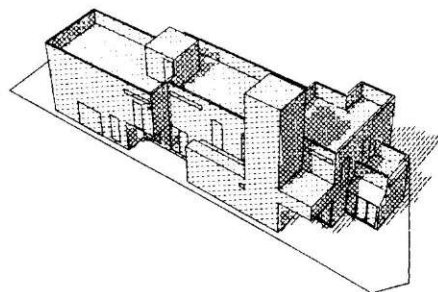
(c)



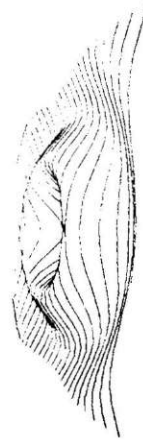
(g)



(d)



(h)



(i)



(j)

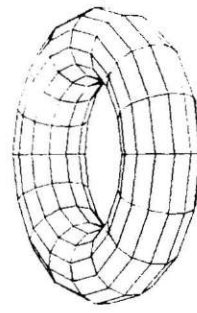
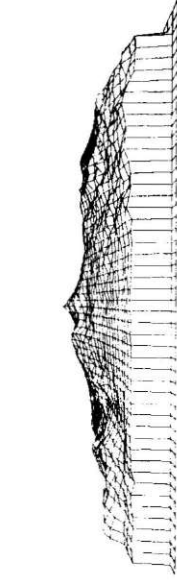


図7. 応用例

(1)

謝辞 本研究にあたり、くさまと御助言いただいた。広島大学の
中前、山下両先生、計算機を使用させていただいた。広島大学計
算センターおよび名古屋大学大型計算機センターの各位に感謝し
ます。

文献

1. P.P.Loutrel: "A solution to the hidden line problem for computerdrawn polyhedra,"
IEEE, Trans. Computers, Vol. c-19, P205~P213, March, 1970.
- 2 中前, 西田: "三次元物体のHidden Lineの処理の手法" 電気学会全国大会 P.1919 (1971)
- 3 中前, 西田: "多面体の隠線消去の手法" 情報処理, Vol.13 No.4 P.239 (1972).
- 4 中前, 西田, 西田: "凸多面体の性質を利用した隠線消去と, その応用" 豊田学会中国
支部大会, 33 13 (1972)
- 5 西田, 中前: "三次元物体の陰影表示シミュレーション" 信学会, 画像工学研究会資料 11-72k(1972)
- 6 西田, 中前: "輪郭線を利用した多面体の隠線消去の手法" 豊田学会中国支部大会, 33, 19 (1972)
- 7 西田, 中前: "自由曲面の隠線消去の手法" 電気学会全国大会 P. 1450 (1972)