

## 三面図からもとの立体の自動復元†

—平面だけで構成される立体への適用—

千 田 豊 満\*\*

三面図からもとの立体を復元する手法は、3次元CADの立体の形状の入力方法として、また多くの視点の画像からもとの立体を認識する方法として利用でき、工学上著しく有益である。このためこの方法は多くの研究者によって開発されてきたが、まだまだ未完成である。本研究はCADシステムにおいてこの手法を開発することを目的とし、平面で構成されているあらゆる形状の立体に適用できる方法を開発している。つまり、2次元CADシステムで描いた正面図、平面図と側面図の投影図を入力すると、計算機の内部に対象物のソリッドモデルのデータが構築できる手法の開発である。このとき三面図が2種以上の立体を意味するときには、そのすべての立体を復元できていなければならない。本手法では、1組の投影図から3次元空間上に稜線を作成し、この稜線から平面を生成する。つぎに、これらの平面を組み合わせることで立体を生成し、これらの立体から作成される投影図をもとの投影図と比較し、一致する立体のみを選び出す。さらに、これらの立体を構成する平面に対して実体の存在する方向を決定し、隠れ線も考慮した立体の投影図を作成し、もとの投影図ともう一度比較し、一致する立体を選び出す。つまりこの段階で目的とする立体が復元できる。本手法をいろいろな投影図に適用した結果、いずれももとの立体を復元できた。また投影図が複数個の立体を意味するときも、すべての立体を復元できた。

### 1. はじめに

現在多くのCADシステムが提案・市販され、多くの事業所で設計の合理化に貢献している。しかし、これらのシステムの持つ可能性から考えると、十分利用され尽くしているとはいえない。その最も顕著な例が3次元CADシステムである。

3次元CADシステムは対象物の形状を表す内部モデルとして、ソリッドモデルを利用し、体積などのマスマトリックスを1回の操作で求められるなど、その応用範囲は非常に広い。しかし、このシステムを導入している事業所での利用実績はいま一つである。その最大の理由は、対象物の形状の入力方法が複雑なことおよび在来の製図法と大幅に異なることである。

以上の理由により、筆者は在来の製図法により対象物の投影図を描いただけで、後は計算機の内部処理で2次元のCADデータをソリッドモデルのデータ（境界表現）に自動変換する方法を開発してきた。つまり、2次元CADシステムのグラフィックディスプレイ上で対象物の三面図を描くと、後は内部処理だけで立体を復元し、体積などのマスマトリックスを1回の操作で求めることのできるシステムである。

三面図からもとの立体を復元する主な手法は、出沢

ら<sup>1)</sup>によって初期に開発された。この手法は、物体要素の真偽の判定規則を頂点と稜線のつながりによってのみ記述し、三面図から3次元空間上に稜線群を復元した時点で、この規則を適用して、探索的に偽物体要素を除去している。しかし、この方法は1組の投影図が複数個の立体を意味するときには適用不可能である。

Markowskyら<sup>2)</sup>は三面図から3次元空間上に稜線群を生成し、つぎにこれらの稜線から面群を生成し、これらの面を組み合わせることで立体群をすべて数え上げている。しかし、もとの図面に適合する立体を特定できていない。

佐々木ら<sup>3)</sup>は三面図から稜線群を、これらの稜線群から面群を生成し、これらの物体要素に真偽の判定基準を適用して、立体を復元している。しかし、この方法では対象物によっては復元できない場合もあり、また隠れ線を考慮していないため、図面が意図している立体以外にも多くの立体を復元してしまう。

本報では、三面図から稜線群（稜線候補）を生成した後、これらの稜線群から平面群（面候補）を生成する。つぎに平面群を組み合わせることで立体群（立体候補）を生成し、この立体群の投影図ともとの投影図を比較して、一致する立体（一致立体候補）を選び出している。さらに、以上の立体のうち面同士が交差しない立体（実在立体候補）の投影図を隠れ線も含めて作成し、もとの投影図と比較して一致する立体（対象立体）を選び出し、目的とする対象物を復元している。以上の

† Reconstruction of Solid from a Set of the Orthographical Three Views—Applications to Many Polyhedral Solids—by TOYOMITSU SENDA (Department of Mechanical Engineering, Faculty of Engineering, Ehime University).

\*\* 愛媛大学工学部機械工学科

手順で処理を進めることにより、計算機内部では2次元の三面図のデータを3次元のソリッドモデル（境界表現）のデータに自動変換している。

この手法はあらゆる形状の多面体に適用でき、三面図が複数個以上の立体を意味するときには、そのすべてを復元でき、実用性が最も高い手法である。

本報ではこの手法を説明するとともに、多くの投影図に対する適用結果を示す。また、この復元過程はグラフィックディスプレイ上で逐一描いているので、過程の一部分はディスプレイ上の画像を直接撮影した写真でも示す。

## 2. ハードウェアおよびソフトウェアの構成

本手法を開発するため利用しているハードウェアの構成は、日本電気製ワークステーション (EWS 4800/20, 処理速度: 7 MIPS, 主メモリー: 16 MB, 磁気ディスク容量: 328 MB, 3.5 インチ・フロッピーディスク装置, グラフィックディスプレイ: 16 インチ・カラー) を中心として、小型 X-Y プロッタ1台、ページプリンタ1台からなる。

基礎となる投影図を描くためには、日本電気製2次元 CAD ソフトウェア“CAE 2D”を利用する。また、本手法の開発には Fortran 言語を使用し、結果や途中経過を画面に表示するときは、EWS 4800 に備えられたグラフィックライブラリを利用する。

## 3. 対象物および入力条件など

つぎの条件のもとに、三面図のデータをソリッドモデルのデータに自動変換する手法を開発する。

対象物としては、

- ① 対象物は平面だけで構成されている1個の部品であること。
- ② 対象物は点または線だけでつながっているようなくびれ部を有していないこと。

入力の図面データとしては、

- ① 2次元の CAD システムで作成された正面図、平面図および側面図からなる1組の三面図であること。
- ② 対象物の形状は、外形線および隠れ線で完全に表され、対称中心線、破断線、板厚などの形状を表す製図記号などは使用されていないこと。

その他として

- ① 原理的にはあらゆる形状の対象物に適用できること。

- ② 1組の三面図が、対象物を表す図面として十分でなく、異なった数個の立体を表す可能性があるときには、そのすべての立体を復元でき、その中から利用者が意図する立体を選択できること。
- ③ 計算機の処理速度の高速化を期待し、処理速度を問題としない。つまり、処理時間が長時間に及ぶと実用的には適用不能であるが、今回は処理時間からだけではこのような結論は下さない。

以上の条件のもとに、システムの開発を行う。

## 4. 自動変換の手順

本報で開発している自動変換の手順を図1に示す。つぎに、順をおってこれらの項目を説明する。

### 4.1 三面図の作成

2次元 CAD システム (CAE 2D) で対象物の三面図を描き、対象物の2次元の形状データを作成する。

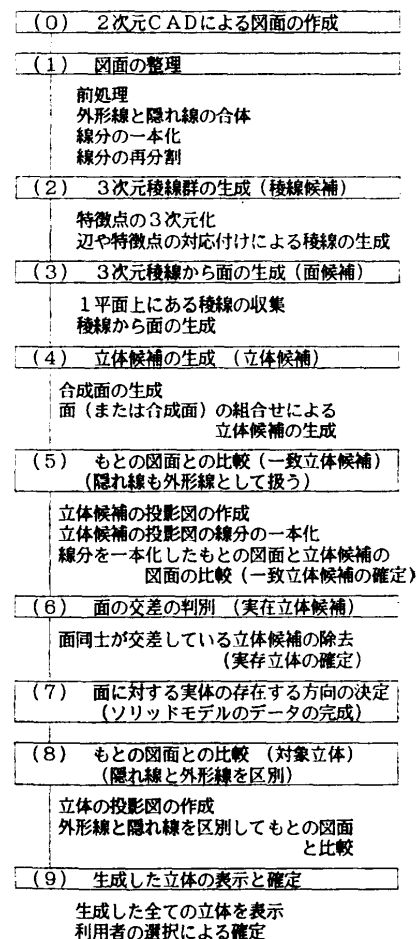


図1 変換の手順

Fig. 1 General flow of solid construction from a set of orthographic three views.

2面以下の面図だけでも対象物の形状を正確に表せる場合もあるが、今回は必ず3面からなる投影図を入力データとする。

4.2 図面の整理

2次元 CAD にもいろいろな図形データの記憶方法がある。線分データの記憶方法を例にすると、代表的なものとして、線分の始点と終点の  $x, y$  座標値を記憶する方法、最初に点に通し番号を付け、その座標値を記憶し、次に線分の両端点の点番号を記憶する方法の2種がある。いずれの方法で記憶しているシステムにも対応できるように、後者のシステムであれば、その記憶方法を前者のシステムと同じになるように変更する。

隠れ線と外形線を区別して以後の処理を進めることはこの段階では不可能であるので、最初は隠れ線と外形線と同じように扱う。もちろん、最終的には隠れ線と外形線は区別して扱うが、これは後で説明する。

CAD で描いた線分は、1本の線分であるにもかかわらず、2本に分割して描いたり、1本の線分を2重に描いていたりする。そこで、後の処理の利便を考え、このような場合は1本の線分として定義する。

つぎに、1本の線分を特徴のある点、例えば他の線

分との交点などで分割する。この処理で得られた線分を“辺”と名付ける。また、この辺の端点を“特徴点”と名付ける。

4.3 稜線候補群の生成

以上の処理は前処理とも言うべきもので、つきからが本格的な処理となる。図2(a)の実例に即して以後の処理手順を説明する。同図で右上が対象物であり、左上(平面図)、右下(側面図)、左下(正面図)が1組の投影図である。

4.2節までの処理で得られた辺および特徴点は2次元の座標値しか持たないが、以下の処理により3次元の座標値を持つ頂点(候補)と稜線(候補)に変換する。図2(b)でその手順を説明する。

正面図の辺1-2に対応する辺または特徴点を側面図から求めると、1, 6, 2の3個の特徴点と、1-6, 6-2, 1-2の3本の辺が対応する。さらにこれらの特徴点と辺に対応する辺または特徴点を平面図に求めると、側面図の特徴点1, 6, 2に1-2, 5-6と4-3の3本の辺が対応し、側面図の辺1-6, 6-2, 1-2に対応する辺は見あたらない。したがって、正面図の辺1-2、側面図の特徴点1と平面図の辺1-2から稜線①が生成される。同じ処理により、稜線②、③が生成される。また、正

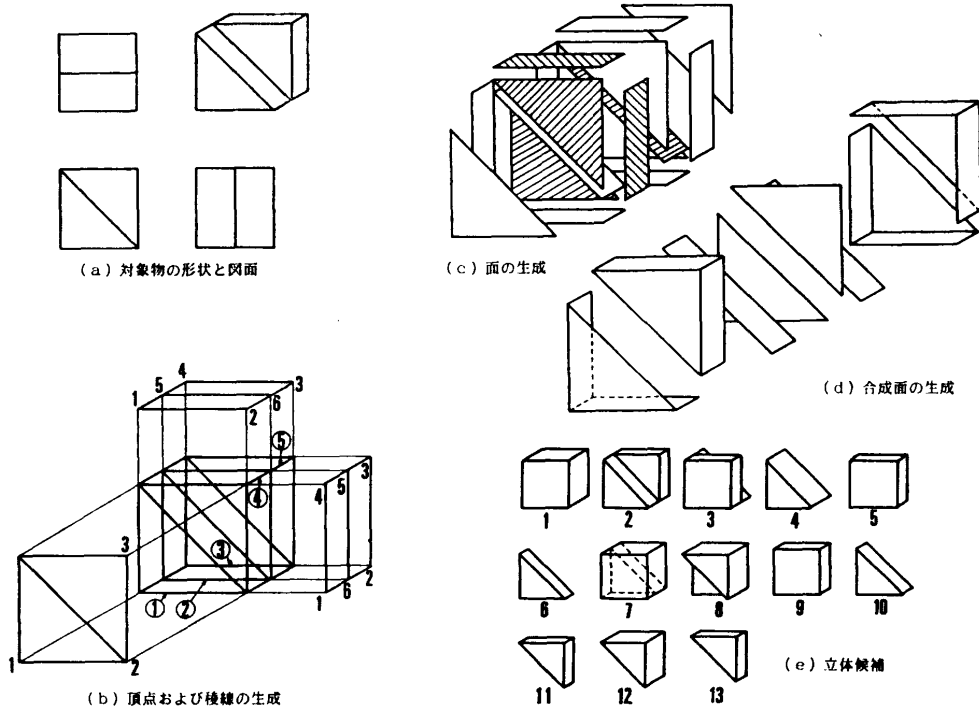


図2 変換の手順例(立体候補の生成まで)

Fig. 2 Early stages of solid construction from a set of orthographic three views (stage 0~stage 4).

面図の特徴点3, 側面図の辺 4-5, 5-3, 4-3, 平面図の辺 2-6, 6-3, 2-3 から稜線④と⑤, および④と⑤がつながった稜線が生成される。

以上と同じ操作をすべての辺と特徴点に施すと, 図2(b)の中央に太線で示す3次元の座標値を持った稜線が復元される。ここで注目しなければならないことは, この処理では実際には存在しない稜線②, ④などが生じることである。つまりこの段階では実際に存在しない稜線も生じているので, これらの稜線を“稜線候補”と呼び, このうち実際には存在しない稜線を“虚稜線”と呼ぶ。

#### 4.4 稜線候補群から面候補群の生成

前の処理で得られた稜線群をもとにし, 3次元の座標値を持つ面を生成する。最初に同一平面上にある稜線をすべて集め, その中から閉ループを構成する稜線を選び出し, それらの稜線で1つの面を生成する。このとき同一平面上で2個以上の閉ループができるときは, 互いに独立であるときには別々の面とする。また包含関係にあるときは外のループと内のループで囲まれた面を1個の面として, 内のループだけで囲まれた面を1個の面として, 別々の面とする。さらに内のループの内側に閉ループが存在したときも同じように扱う。

図2の実例では, 同図(c)の面群が生成される。この場合は穴があいた面がなく, 包含関係にある閉ループは存在しない。

この段階でも実際の対象物には存在しない面(図中斜線を施した面)が生成されるので, この段階で生成した面を“面候補”と呼び, 実際には存在しない面を“虚面”と呼ぶ。本例では, 虚面は虚稜線が関係して生成されているが, 虚稜線が全く関係せずに虚面を生じるときもある。

#### 4.5 立体候補の生成

つぎに前段階で得られた面候補群を,

- (1) 1個の稜線は必ず2個の面候補に連なる。
- (2) 立体は面候補によって過不足なく完全に包まれる。

という必要条件のもとに(この条件は面候補から立体を作るための条件として十分ではない), 面候補を組み合わせて立体を生成する。

面候補の組合せの個数は面候補の個数とともに非常に増大する。検討する面候補の組合せの回数を減らすため, 図2(d)のように, 1本の稜線に2個の面候補だけが接しているときは(つまりこの2個の面の組合せが1組に限られるときは), この2個の面候補を合

わせて“合成面”とする。もちろん3個以上の構成面が, このようにただ1つの組合せしかないときは, これらのすべての面を合わせて1個の合成面とする。

合成面と合成面を構成しない面候補を組み合わせて, 立体を生成する。そして, この立体の面候補で稜線で接して同一平面上にあるときは, これらの面の間の稜線(図2(e)(2)の立体ならば, 図2(b)に示す稜線②がこの稜線に対応する)を消去し, 両者の面候補を合体させて1個の面とする。

以上の処理により, 本例の場合は図2(e)に示す13個の立体が生成される。この段階で得られた立体には対象物でない多くの立体が生成されるので, これらの立体を“立体候補”と呼ぶ。

本例の場合同図(e)の立体候補はすべて実際に存在できるが, 場合によっては立体を構成する面が互いに交差し, 実際に存在できない立体を生成するときもある。

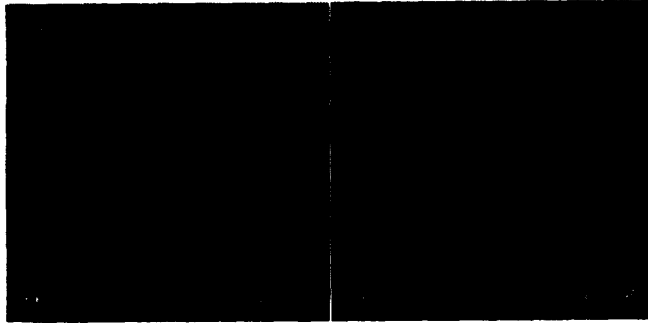
#### 4.6 もとの図面との比較

立体候補の3面からなる1組の投影図を作成する。この場合は, 隠れ線の処理を必要としないため, 立体のデータから簡単に1組の投影図が得られる。しかし, ここで注意することは, 比較するもとの投影図が, 線分を一本化し, つぎに特徴点で再分割している。この投影図でも線分を一本化し, 特徴点で再分割する。その後両者の個々の線分を比較することにより投影図が一致するかどうかを判定し, 一致する立体を選び出す。この一致する立体を“一致立体候補”と呼ぶ。

本例の場合は, 図2(e)から明らかのように, (2), (3), (7)と(8)の4個の一致立体候補が選び出される。図3(a)(b)にはこの段階でグラフィックディスプレイ上で示される写真を示す。同図では左下にもとの投影図が, 右上に1個の立体候補とその投影図が示されている。本写真は白黒であるので明らかでないが, 実際には投影図が一致するときは立体候補の投影図が青色で, 一致しないときは橙色で示されている。

#### 4.7 面の交差の判別

選び出された一致立体候補のすべてに対して, 立体を構成する面同士の交差を検討し, 交差している面を含む立体を対象から除外する。ここまでの処理により, 隠れ線を外形線と同じように扱ったときに, その投影図がもとの投影図に一致する一致立体候補のうち, 実際には存在できる立体だけが選び出される。この段階で選び出された立体候補を“実在立体候補”と



(a) 一致する例 (b) 一致しない例  
 図3 ディスプレイ上での立体候補ともとの図面との比較  
 Fig. 3 Comparisons of orthographic views of solids with those original three views on CRT screen (stage 5).

呼ぶ。

本例の場合は、図2(e)から明らかなように、面同士が交差している立体候補は存在しない。

以上までの処理により、図4(a)(b)(c)(d)に示す4個の实在立体候補が対象として残る。

4.8 面に対する実体の存在する方向の決定

3次元のソリッドモデルは対象物を構成する面を定義し、その面に対して実体の存在する方向を明らかにしたものである。そこで、今までの処理で残った实在立体候補の構成する面に対して実体の存在する方向を決定する。

つぎに図5を利用して処理方法を説明する。

- (1) 最も外側に存在する頂点(具体的には  $x, y, z$  座標値の最も大きい頂点)を捜し出す。この頂点を図5で点Aとする。
- (2) その頂点に連なる稜線のうち、最も外側に存在する稜線を検出する。実際にはいずれかの投影面に投影し、傾斜の最も大きい稜線を選び出せばよい。この稜線を図5で稜線ABとする。
- (3) この稜線ABに連なる面を2個取り出し、それぞれの面でこの頂点を通る稜線のもう1本の稜

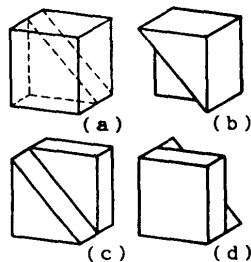


図4 实在立体候補  
 Fig. 4 All virtual solids constructed from the original three views (stage 6).

線を計2本選び出す。この稜線を図5でAC, ADとする。

- (4) 図5では稜線ACが稜線ADより手前側になるように示しているが、この段階ではいずれが手前側にあるか判定できていない。そこで、頂点A, B, Cがつくる平面ACDと点Bの関係を検討することにより、いずれが手前側にあるかを判定する。
- (5) 以上までの処理で(図5のように稜線ACが稜線ADの手前側にあるとすると)、面ACBより向こう側に実体が存在すると決定できる。

- (6) 1個の面に対して実体の存在する方向が決定できると、順次隣接する面に対する実体の方向が簡単に決定できる。

以上の処理により立体の完全なソリッドモデルのデータが完成する。

本例の場合は、一例を図6(a)に示すように、実体の存在する方向を決定できる。本システムでは、この段階で立体を構成する面ごとに図6(b)に示す図をデ

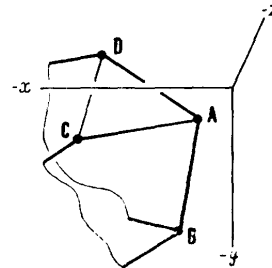
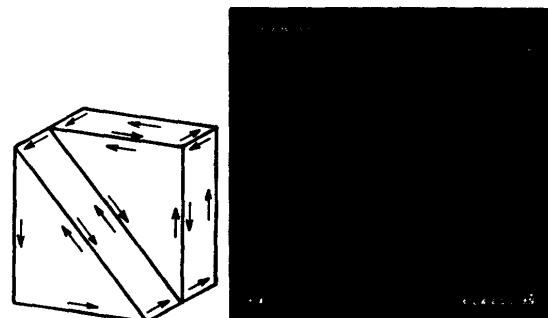


図5 ソリッドモデル化の方法  
 Fig. 5 Boundary representation technique for solid modeling of the solids (stage 7).



(a) 実体の存在する方向 (b) ディスプレイ上の1コマ  
 図6 面に対する実体の存在する方向  
 Fig. 6 Boundary representation of a solid on CRT screen (stage 7).

ディスプレイ上に表示している。

#### 4.9 もとの図面との比較

以上までの処理で残ったすべての実在立体候補の投影図を作成する。この段階では立体の完全なソリッドモデルのデータができていたので隠れ線処理が可能となり、隠れ線も考慮した投影図が得られる。この場合も線分の1本化、線分の再分割、外形線と重なった隠れ線の消去などを行い、もとの図面と比較する。

本例の場合は、図4のうちの(c)の実在立体候補の投影図だけが、もとの投影図と一致する。つまりこの立体がもとの投影図から復元される立体であり、本研究では“対象立体”と呼ぶ。この状態のグラフィックディスプレイ上の表示状態を図7(a)(b)に示す。

本例ではこの段階で、対象立体が1個であるが、2個以上生じる場合もある。

#### 4.10 生成した立体の表示と確定

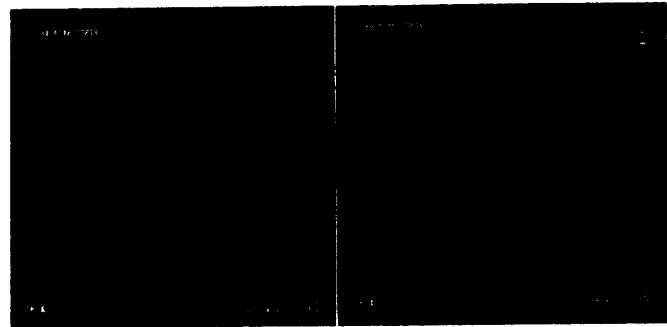
この段階まで2個以上の対象立体が候補として残ったときは、これ以上は論理的に処理を進めるのは不可能である。つまり、2個以上の立体が残る場合は、対象物の正確な形状を3面からなる1組の投影図では表せない場合であり、正確に対象物を表すためには、他に補助の投影図または断面図などが必要な場合である。本システムでは、この段階まで残った立体をすべてグラフィックディスプレイ上に表示し、利用者が意図する立体を選択するようになっている。

この操作で全処理が終了する。

### 5. 種々の図面に対する適用

本手法を組み込んだシステムは Fortran 言語で約13,000行のプログラムである。このプログラムには中間段階の結果をチェックするための出力文、配列が崩れたことを示す警告文やグラフィックディスプレイ上への中間結果の表示のための命令文などが数多く含まれ、図1の処理だけでは約10,000行である。

種々の図面に対する適用例を図8(a)~(f)に示す。また処理の過程で生じた稜線候補、面候補、立体候補、一致立体候補、実存立体候補、対象立体の個数を表1に示す。さらに、全処理時間(CPU時間)と全処理時間のうち、面候補から立体候補を生成するときに要した処理時間も表1に示す。全体の処理時間には、処理の途中経過を逐一ディスプレイ上に表示するための時間も含んでいるため、図1の処理だけの実際



(a) 一致する例(対象立体) (b) 一致しない例

図7 ディスプレイ上での実存立体候補もとの図面との比較  
Fig. 7 Comparison of constructing solids with the original three views of the object (stage 8).

表1 処理の過程と処理時間

Table 1 Process of processing and processing time.

対象物	処理の過程						全時 処理 時間 (sec.)	立を 作る 体作 時間 成間 補す (sec.)
	稜 線 補	面 候 補	立 候 補	一 体 致 候 補	実 体 存 候 補	対 立 候 補 体		
図2(a)	19	16	13	4	4	1	4.6	0.1
図8(a)	5	10	3	1	1	1	1.4	0.1
(b)	40	30	20	1	1	1	14.4	5.1
(c)	14	16	41	13	9	5	31.2	22.0
(d)	25	29	170	1	1	1	486.8	350.5
(e)	54	39	36	6	6	3	78.3	56.1
	54	36	11	6	6	3	17.9	0.3
(f)	151	71	47	1	1	1	222.0	62.1
	151	64	3	1	1	1	117.6	0.6

注 同一対象物で2段にわたり記しているのは、上段が原理に忠実な方法によるものであり、下段は穴となる面を面候補から除外して、以後の処理を進めたものである。

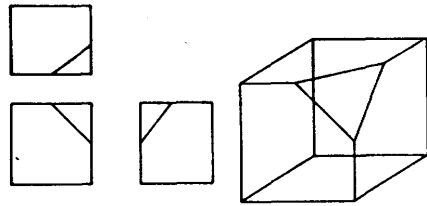
の時間はもう少し短い。

(a)は簡単な形状であるが、処理の過程で虚面と3個の立体候補が生じる例である。

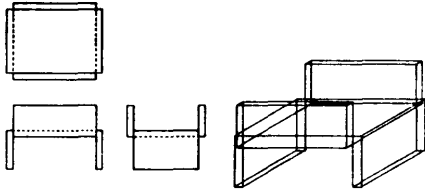
(b)は少し複雑な形状の例である。

(c)は三面図は単純であるが、多くの立体候補を生成し、またこれらの立体候補の内には面が交差するものが存在する例である。

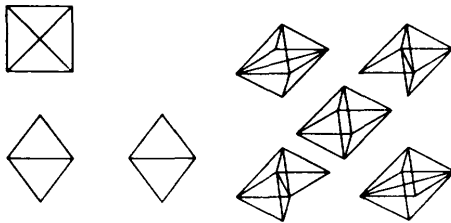
(d)は三面図は非常に単純であるが、多くの虚面と立体候補を生成する例であり、処理時間が非常に長い例である。また人間にとっては投影図から立体を復元しにくいと考えられるが(個人差はあるが)、本システムでは簡単に立体を復元できる例である(ただし、



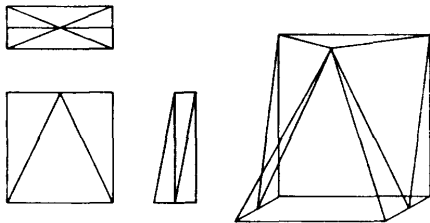
(a)



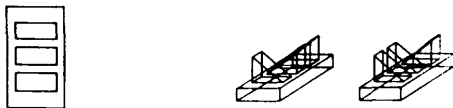
(b)



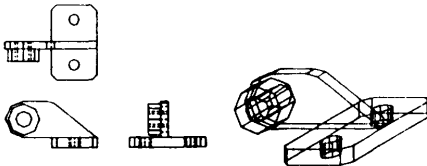
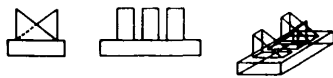
(c)



(d)



(e)



(f)

図 8 種々の図面への適用例  
Fig. 8 Many examples of solid constructions from three views.

多くの立体候補を生じているので、形状が簡単な割には、多くの処理時間を要する)。

(e)は1組の三面図から3個の対象立体が生じる例である。

(f)は形状が少し複雑で、機械部品らしい例である。

図9には図8(e)の例の最終段階でのグラフィックディスプレイ上の画面を示している。本システムでは利用者がこれらの立体から意図する立体を選べるようにしている。

図10は本適用例の中で最も長い処理時間を要する例であり、処理の中間段階(図1の第(5)段階)の一例を示している。左下が入力する投影図であり、右上が立体候補の一例と、隠れ線と外形線を区別しない場合の投影図を示している。この場合、立体候補を生成する処理(図1の(4))を全体の16分の2だけを実行させたところ、CPU時間が17時間必要であった。実際にすべての立体候補を生成し、最後まで処理させると、約140時間程度が必要であろう。立体候補の生成

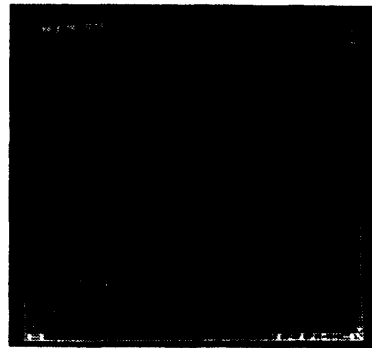


図 9 図 8 (e) の最終画面  
Fig. 9 All possible solids constructed from the three views of Fig. 8(e) on CRT screen (stage 9).

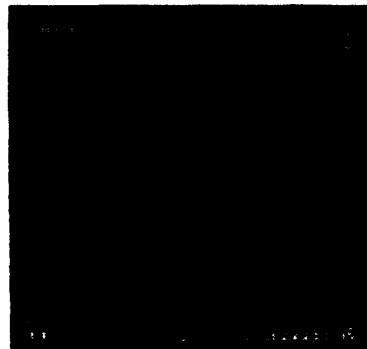


図 10 適用例の中間段階の例(立体候補ともとの図面との比較)  
Fig. 10 An example of solid construction (stage 5).

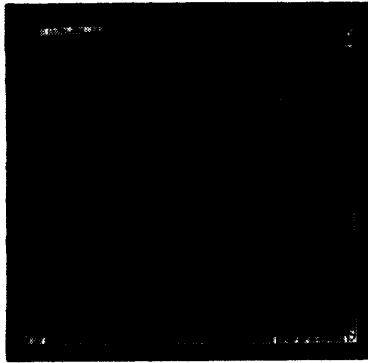


図 11 適用例の最終段階 (意図する立体の選択)  
Fig. 11 CRT screen at the final stage (stage 9).

をこの段階で切り上げ、最後まで(図1の(6)から(9)まで)処理させた結果を図11に示す。図10と図11の結果から判断して、この例も時間さえかければ復元可能であると結論できる。

## 6. 本手法の発展性

以上の多くの適用例に示すように、本手法は処理時間さえ問題にしなれば、原理的にはあらゆる形状の多面体の対象物に対して適用可能である。しかし、3次元CADシステムへの組み込みを考えると、処理時間や円柱などの曲面を含んだ対象物への適用という点で問題を残している。

つきに、処理時間や必要となる記憶容量について若干考察してみる。図1の手法は自動変換の原理の最も単純な手順を示したものであり、処理時間や必要な記憶容量を減少させる方法があり、それらを組み入れることにより処理時間や記憶容量を減少させることができると考えられる。

例えば、図1の(3)の面候補の生成で、つまり稜線候補から面候補を生成する段階で、複数個の閉ループが生じそれらが包含関係にあるときには、外側と内側のループで囲まれる面だけを面候補とし、内側のループを面候補から除外する(つまり穴の面を除外する)と、処理時間は著しく減少する。この方法を組み込んで、図8(e)と図8(f)に適用した結果を表1のそれぞれの対象物の欄の下の段に示す。表1を比較すると、立体候補の個数が減少し、立体候補を生成するための時間が著しく減少し、これにともない全処理時間も減少していることが明らかである。また、復元される立体候補の個数が減少しているので、必要な記憶容量も減少している。

図1の処理では立体候補をすべて生成した後から、もとの図面と比較し、順次処理を進めている。しか

し、もとの図面との比較以後の処理を立体候補を1個生成するごとに進めると、すべての立体候補を一時的に記憶する必要がなくなり、必要となる記憶容量は大幅に減少する。

曲面の代表例として円柱面または円弧面を含んだ対象物への発展性を考察する。三面図の段階でこれらの曲面を平面で近似するならば、対象物が平面だけで構成されることになり、後の処理はほとんど図1の手順を適用することにより、可能であると考えられる<sup>9)</sup>。

2面以下による1組の投影図からの復元は本報では扱っていないが、三面図から稜線候補群を生成する段階および立体候補もとの図面との比較の段階で、本手法を少し手直しするだけで対応できると考えられる。また、中心線などの製図記号もこれらの段階で少し手直しすれば対応できると考えられる。

なお、図1に示す処理手順のうち、(5)のもとの図面との比較と(6)の面の交差の判別は入れ換えてもよい。

## 7. おわりに

本研究では、2次元CADシステムで平面だけで完全に覆われている対象物の3面からなる1組の投影図を描き、この投影図からもとの対象物のソリッドモデル(境界表現)のデータを生成する自動変換の手法を開発した。

開発した自動変換の手法をいろいろな対象物の投影図に適用したが、いずれももとの立体が復元でき、ソリッドモデルのデータを生成できることが確認できた。

結局、原理的にはあらゆる形状の立体に適用ができ、1組の投影図が複数の立体を意味するときは、そのすべての立体を復元できた。

処理の過程で虚面を生じないものは処理時間が短かった。しかし、虚面が生じ多くの立体候補が生じる場合には処理時間が著しく増大し、完全に処理できない場合もあったが、原理的にはこの手法で可能である。また、この処理時間を減少させるためにはいろいろな方法を組み込む余地が存在すると考えられる。

本報で説明した本手法は原理の最も単純な手順を示したものにすぎず、改良発展することにより実用の3次元CADシステムに組み込むことができると確信できる。



## 参 考 文 献

- 1) 出沢正徳：三面図から立体形成のためのシステム，日本機械学会論文集，Vol. 38, No. 310, pp. 1267-1276 (1972).
- 2) Markowsky, M. A. et al.: Fleshing Out Projections, *IBM J. Res. Dev.*, Vol. 25, No. 6, pp. 934-954 (1981).
- 3) 佐々木康仁ほか：三面図からの物体自動合成のための線形擬似ブール代数解法，情報処理学会論文誌，Vol. 28, No. 12, pp. 1288-1297 (1987).
- 4) 例えば，佐藤義雄：実習グラフィックス，p. 71，アスキー出版局，東京 (1986).
- 5) 千田豊満ほか：2次元 CAD データの3次元データへの自動変換（円柱面を含む対象物への拡張），日本設計工学会講演論文集，No. 89-秋季，pp. 53-56 (1989).

(平成2年1月24日受付)

(平成2年6月4日採録)



千田 豊満 (正会員)

昭和17年生。昭和41年静岡大学工学部機械工学科卒業，昭和43年大阪府立大学大学院工学研究科修士課程修了。同年愛媛大学助手を経て現在同大学教授。その間昭和57年“衝撃荷重による材料の変形挙動に関する研究”で大阪大学工学博士。現在，“衝撃荷重下での材料の変形と強度”および“CADにおける図形処理”などの研究に従事。機械学会，材料学会，精密工学会各会員。著書：「CAD/CAM システム—基礎から構築まで」(単著)，「機械製図」(共著)，「情報科学」(共著)。