

境界による見える部分の記述を持つ投影体の生成

Generation of the Description of Visible Regions in terms of their Boundaries

越川 和忠

白井 良明

Kazutada KOSHIKAWA

Yoshiaki SHIRAI

電子技術総合研究所

(Electrotechnical Laboratory)

An efficient method for describing visible regions by their boundaries is presented.

For given solid models and a viewer's direction, this method obtains the description of visible regions in the following steps:

- (1) Partition the projection plane into visible regions.
- (2) Find the face which each region belongs to.
- (3) Determine the relationships among the regions in each face.

The obtained description of visible regions is useful for model-matching in order to recognize 3D objects.

1. はじめに

われわれは、物体認識の研究に用いるための3Dモデラを試作している。この報告は、その際具体化した認識向き機能に関するものである。

空間にある対象物に対して機械が何らかの働きかけをする場合には、それらの幾何学的形状や位置、向き、相互の関係などの情報が必要となる。『物体認識』は、主に、そのような情報をシーンのTV画像にある手掛かりから求める手続きを指す。ここでは、対象物が既知で、その形状が3Dモデラで記述されているとき、その同定やシーンでの位置、姿勢を決定する場合を考えている。

3Dモデラは、多くの場合、表示の際、立体をカメラ座標系で表し投影変換した何らかの構造体を作る。その構造体(『投影体』と呼ぶ)は、主に人の視覚的理解を助ける表示用の作業データに過ぎない。しかし、物体認識では、画像に三次元の手掛かりを求めるので、投影体には、モデルとの照合を行う際の参照体としての積極的な役割が潜在している。

例えば、多面体のある方向から見るとき、見えている一つの面を考えると、この面に接して見える他の面は、本来隣合う面か、この面を隠す面である。他の面に隠される部分では、隠している面の稜でその面に接するように見える。立体は目の位置によって見え方が異なるが、ある範囲内では、投影面上での幾何学的寸法の違いはあっても、見える稜や面の関係は変わらない。したがって、見えている部分がどのような面に隣接して見えるかは、立体の位置や姿勢を知る上で有力な手掛かりとなる。

そこで、物体認識向き3Dモデラに備えるべき機能の一つとして、そのような隣接の記述を投影体に載せることを試みた。その一つの手法は既に報告した¹⁾が今回、より効率よく求めることのできる別の手法を得たのでここに紹介する。

なお、試作中の3Dモデラは、GEOMAPのプロトタイプ²⁾を基にしており立体は多面体に限定している。

2. 隠線処理の強化

立体とその投影体の対応する境界要素を、便宜上、次のように呼ぶ(図1)。

面	←→	域
稜	←→	辺
頂点	←→	隅

カメラ側を向かない面の域は作らないことにすると、域が両側にある辺と片側だけにある辺とが生じる。後者を特に区別するときには、『縁』の語を用いる。

見える部分を左に見る辺で囲む多角形を『柵』と名付け、この

向きを含めた柵の辺をその柵の『貫』、柵で囲まれた見える部分を『領域』、領域の所属する域を『柵域』と呼ぶ。貫は、柵域自身の辺か、他の域の辺なら領域側の域が空の縁が受持つ。見える辺は、二つの柵で貫の役を果たす。

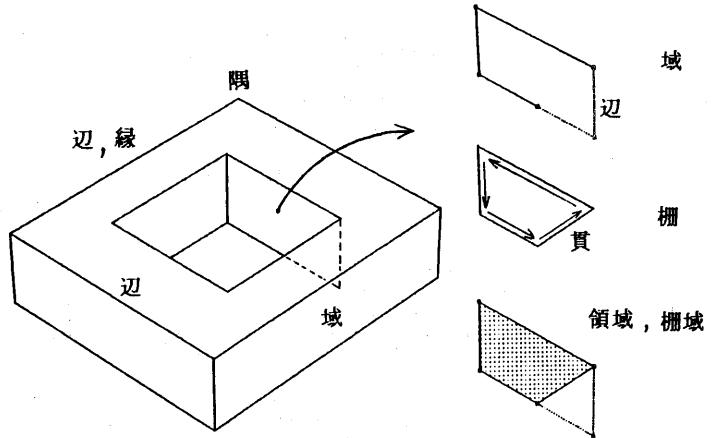


図1 投影体境界要素の呼称

隠線処理を施す際、投影体を単に表示用のデータとして生成する場合には、辺は、見えるか見えないかの単位で生成すれば済む。しかし、柵を得るには、貫を考慮した単位で生成し、貫から貫への接続(『渡り』)も見出せるようにしておかねばならない。そのためには次のような処理を必要とする。

(1) 見える辺の分割

図2 (a)で、面 F_1 の稜 E_1 は、面 F_2 の稜 E_2 を境に一部が隠れる。表示用であれば、(b)のように、 E_1 の辺を、 E_2 の辺 A_2 との交点 P で、見える辺 A_{11} と見えない辺 A_{12} に分けるだけですむ。しかし、面 F_1 が実際にどのように見えるかを知る必要がある場合には、(c)のように、辺 A_2 も P で A_{21} と A_{22} に分けないと、領域 R_1 が $\dots \rightarrow A_{11} \rightarrow P \rightarrow A_{21} \rightarrow \dots$ のように囲まれていることを導けない。(d)の場合も、完全に隠れる稜 E_3 の辺 A_3 を「見えない」とするだけ(e)ではなく、辺 A_2 も隅 P で A_{21} と A_{22} に分ける(f)。

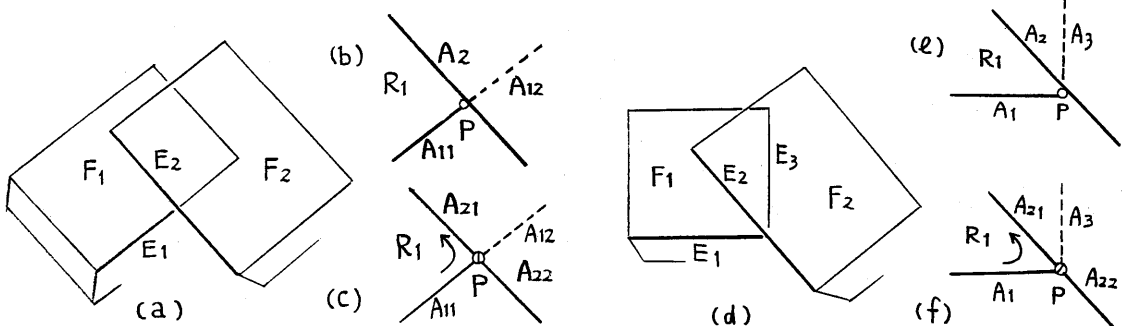


図2 見える辺の分割

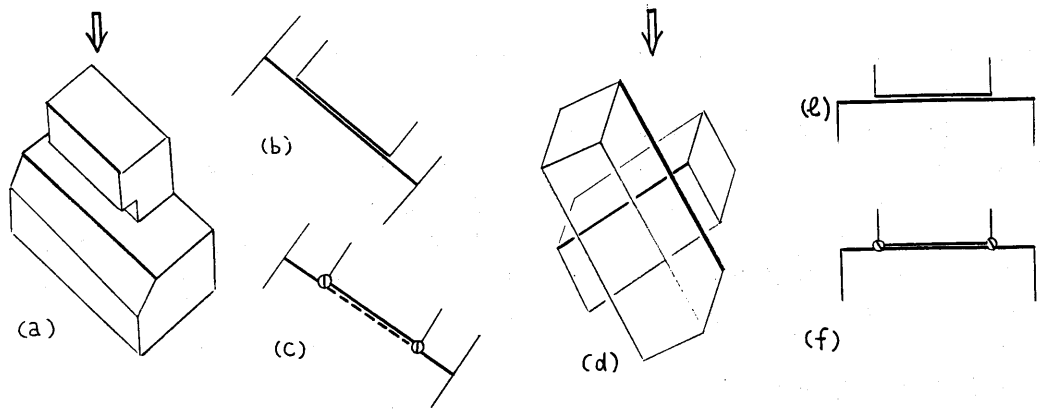


図 3 重なる辺の分割

(2) 重なる辺の分割

図 3 で、(a) の立体を矢印の方向から見ると、(b) のように辺が重なる。表示には共に「見える」としても差支えないが、柵を作る場合、貫同志の競合を生じるので、(c) のように、手前だけを「見える」とする。ただし、縁同志が向かい合う場合には、(d) のように前後の関係が交差することもあるので、共に「見える」とする(f)。

(3) 重なる隅

図 4 で、域 R_1 の柵を考える
と、域 R_2 の辺 A_2 は、域 R_1 の
辺 A_{12} に渡る。この渡りは、隅 P_1
と隅 P_2 が偶々投影面上で幾何
学的に一致するために生じるもの
で、位相的な関係からは得られ
ない。隅 P_1 、隅 P_2 とも、
対応する頂点 V_1 、 V_2 が立体
にあり、独立しているからである。
このように、貫として接続
する辺を探するときには、位相的
な隅だけではなく、幾何学的に
重なっている隅から出ている辺
も調べなければならない。この
幾何学的な重なりは、 R_2 による A_{11} や A_{12} の隠れを調べる際に分るので、その
とき印をつけておく。

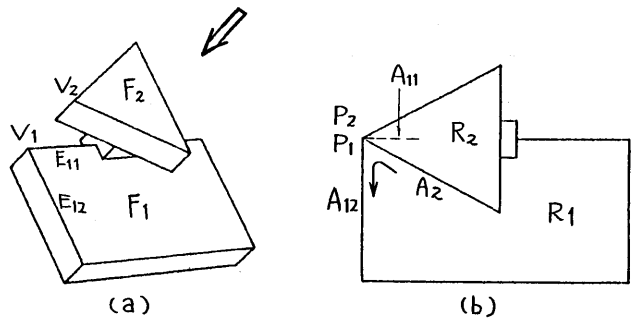


図 4 重なる隅

(1) ~ (3) を含む隠線処理を行うと、正しい貫を求めることができる。

3. 従来の方法による面の見える部分の抽出

既に報告した方法¹⁾では、柵を、特定の面のうちに見える部分を知る観点から

域ごとに求めた。今回の方法との比較のため、その要点を次に記す。

(1) 域 R の柵の貫になる辺に印をつける。

これは、見える辺で、R の辺か、R の辺でなければ、次の条件をすべて満たす辺である：① 縁、② R の外に出ない、③ R との間に他の域がない。

(2) 印のついた辺だけを対象にして、一つを採り、それに接続する貫を探す。

同じ隅で接続しうる貫がいくつがある場合には、最も左に折れるものを探る。

同様にして接続する貫を次々求め、始めの辺に戻ったら、一つの柵とする。

採用済の辺は印を消し、残りの辺に対して同様に繰返す。

(3) 柵同志の包含関係を決定する。

この方法は、考え方は自然であるが、処理に著しく時間がかかる。その原因は隠線処理による分割が完了してから、改めて域ごとに貫を拾うので、(1) の条件②、③を調べるために、辺の内外判定を頻繁に行い、さらに(2)で、接続しうる貫をその都度探す点にある。

この問題は、隠線処理の際に『渡り』も記載してしまうことにより避けられそうに思えるが、必ずしもそうはいえない。

例えば、図 5 の場合、(a) であれば、確かに、辺 A1、A2 の分割のとき、矢印のように渡るとすればよいから問題ない。しかし、(b) の状態であると、域 R に対する A1 を(a) のように処理したあと、R に対する辺 A3 の隠れを調べたとき、隠れは生じないが、渡りは矢印のように訂正しなければならない。また図 6 では、域 R2 の辺 A1、A2、A3、A4 は、交差の全くない他の域 R1 を覗く柵を形成する。これを得るには、やはり、(1) ③に相当する判定を必要とする。

このように、その度に調べ直しながら隠線処理を行うことになり、かえって複雑になる。

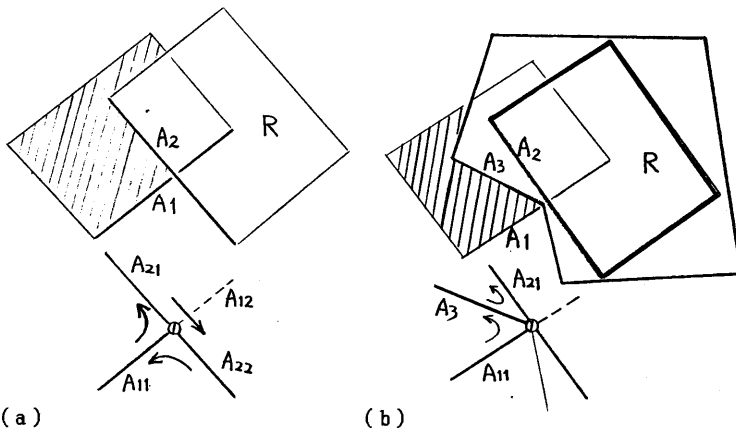


図 5 『渡り』の修正

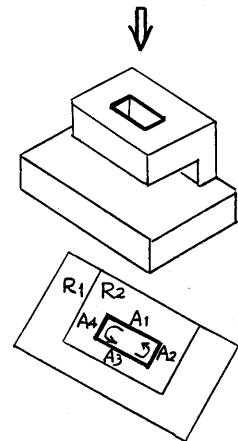


図 6 他の域を覗く柵

そこで、柵に対する見方を変え、やはり分割を完了させてから生成する方針で改良した方法を次に述べる。

4. 柵の領域による投影面の敷き詰め

物体認識では、域→柵だけではなく、柵→域の参照も必要になる。したがって柵を域ごとに作る必然性はなく、始めから投影面全体を対象にしてよい。

そこで、背景も一つの域として扱い、この域を囲む柵も考慮すると、見える辺はすべて、貫として、二つの域に関わる。柵をこのように捉えると、投影面全体は、柵による領域で敷き詰められる。このことから、辺に注目してその接続辺を探す代わりに、接続点である隅に注目して、それに集まる辺の間で渡りを決めればよい。

この敷き詰めを次の手順で求める。

- (1) 隅に集まる辺に対して貫としての接続関係を与える。
- (2) 境界要素としての柵を生成し、その柵域を決定する。
- (3) 同じ柵域を持つ柵同志の包含関係を決定する。

これらの内容を以下に説明する。

(1) 隅に集まる辺の接続関係（渡り）の決定

次の手順で行う。

- ① 重なる隅同志をリンクする。
- ② 見える辺を順に取出し、両端の各隅に、この辺をリンクする。
既に重なる隅や他の辺がリンクされていればそれにつなぐ。
- ③ 見える隅ごとにリンクされている辺を隅周りにソートする。
二つの辺が重なるのは、共に縁の場合である（図7）。このときは、域が空である側が向い合うようにソートする。
ソートの結果、隣合う辺同志がこの隅で渡る貫同志になるので、それぞれの辺の受持つ貫に渡りを記載する（図8）。
ただし、二つの辺が重なる場合には、渡らずに、その印をつけ、代りに隣接情報として、空でない方の域を互いに載せることにする。

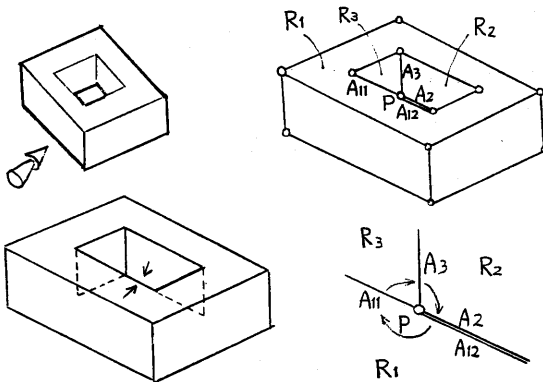


図7 向い合う縁

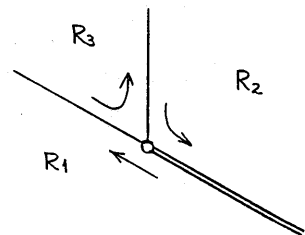


図8 隅における貫の渡り

(2) 柵の生成と柵域の決定 (図9)

(1)の結果、投影面には貫のつながりによる囲みが揃う(c)。しかし、それぞれの囲みはまだ柵としては意識されない。そこで、

- ① 見える辺を順に取り、その辺の受持つ貫の柵を(未生成なら)生成する。
新たに柵を生成したとき、その貫をたどり、各貫が関わる域を調べ、
 - ・空でない域があればその域の柵とする(d)。
 - ・すべて空(つまり貫がすべて縁)ならば、仮に背景域の柵としておく(e)
貫には柵生成済の印をつける。すべての見える辺に対してこの処理を行う。
- ② 背景の域に仮所屬している柵を一つずつとり、その柵を内側に含む域を探す。
あれば、そのうち最も手前にある域に所屬を直す(f)。
なければ、そのまま背景の柵として残す。
- ③ 各見える縁に、域が空の側の柵域を記載する。
この記載により、どの辺も両側にそれぞれ見える域が直ちに分る。

(3) 包含関係の決定

一つの域の見える部分は、その域を柵域にもつ柵から成る。そこで、同じ柵域を持つ柵同志の包含関係を決定し、柵に記載しておく(図9(g))。これは、投影面上の任意の位置でそこに見える面を効率よく探すために必要である。

図10は、二つの立体をある方向から見たときの投影体をこの方法で生成して表示した例である。

(a)は、辺による通常の表示であるが、(b)以下は、柵を表示してある。(b)は、柵域が空の柵、すなわち輪郭であり、(c)は、いくつかの柵を、(d)はすべての柵を、それぞれ同時に示したものである。(d)は、曲面の近似稜が表示されている点を除けば、視覚的效果は、(a)と同じになる。

5. おわりに

立体と同様の境界表現をもつ投影体に、投影面で見える部分を囲む多角形の記述を載せる手法を述べた。この方法は、物体認識の研究用に今試作している3Dモデル・システム上で具体化した。任意の見える部分とそれに関わる他の面や稜の関係は、画像と立体モデルを照合する際に役立つ。

認識向きに必要な機能はほかにもあり、さらに整備してゆく予定である。

参考文献

- 1) 越川、白井：物体認識のためのGEOMAPの整備、第2回日本ロボット学会学術講演会、1984。
- 2) Kimura, Hosaka; Program Package GEOMAP Reference Manual (preliminary version), 1977.

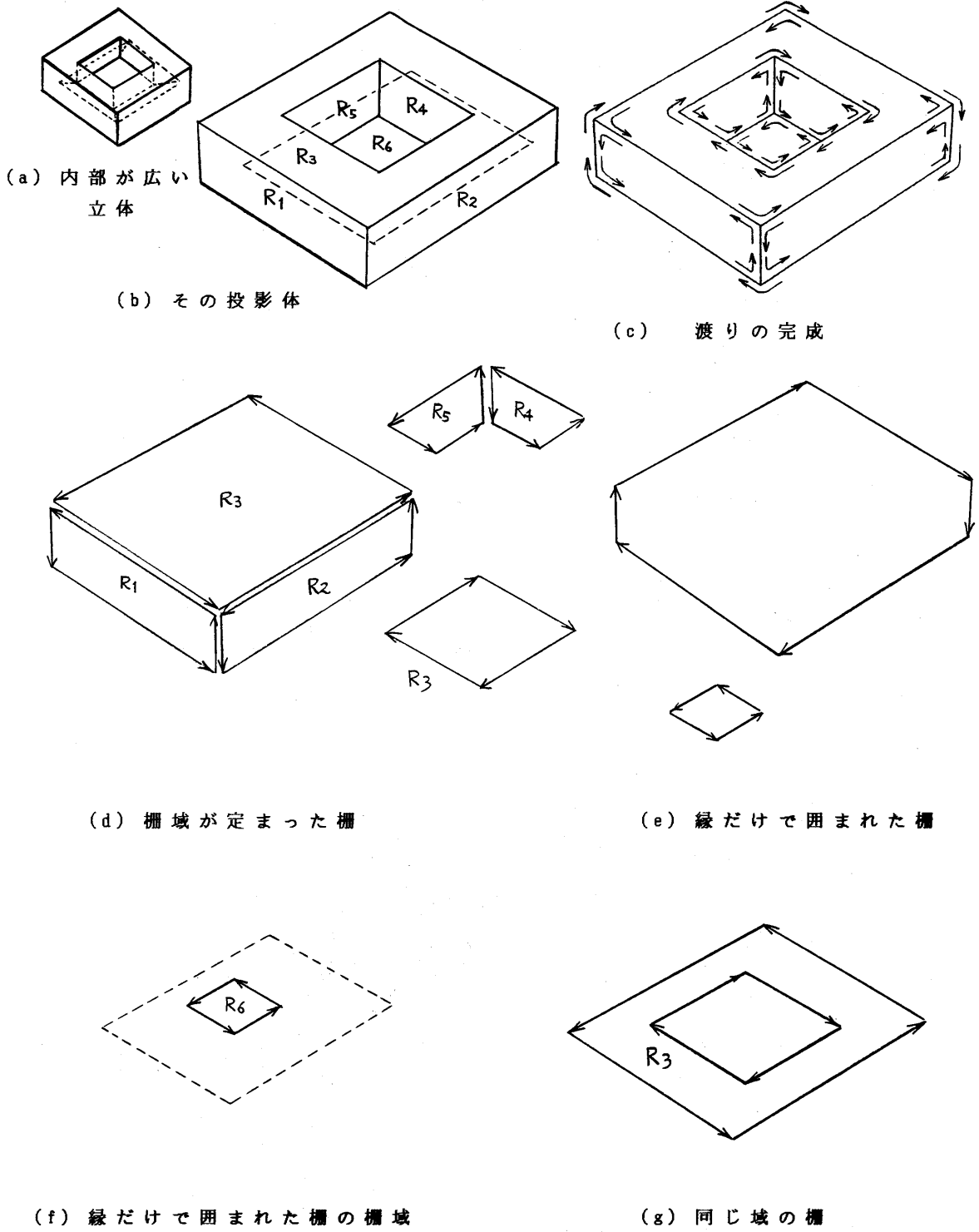
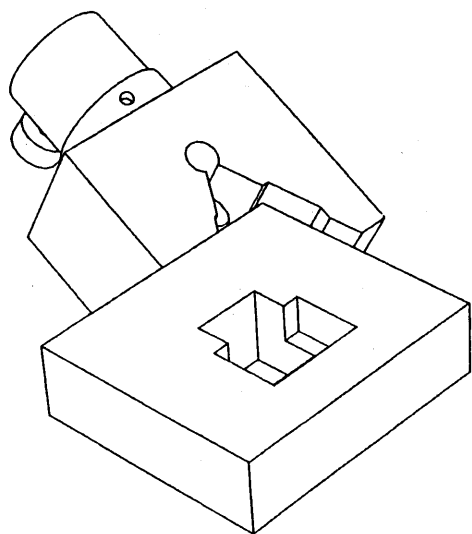
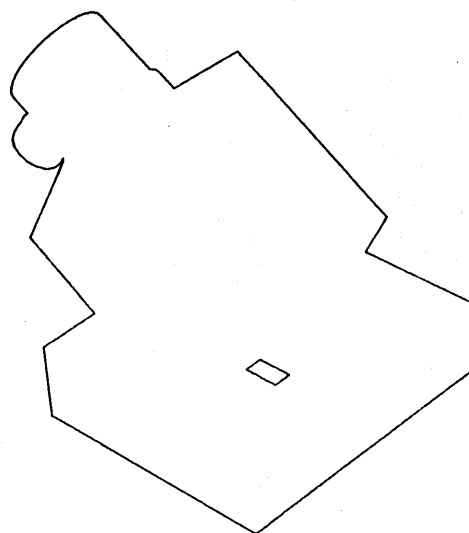


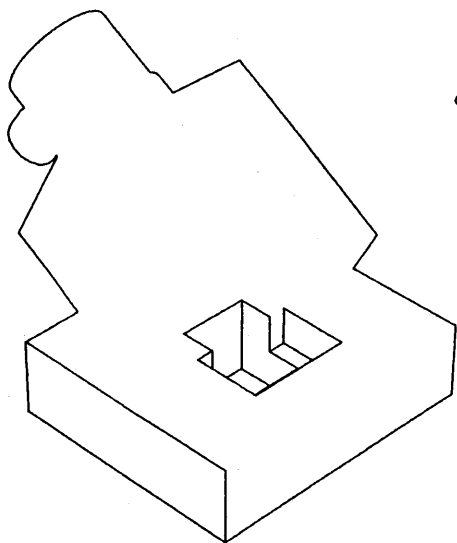
図 9 柵の生成



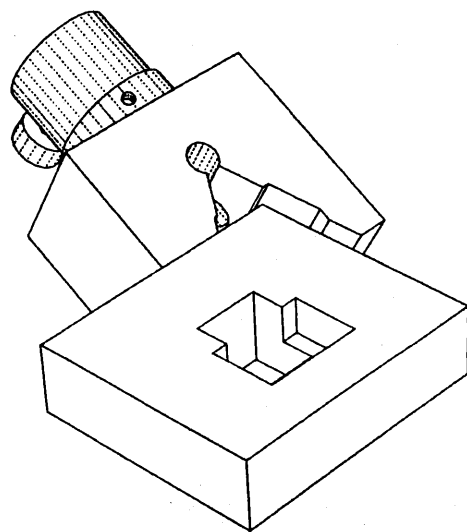
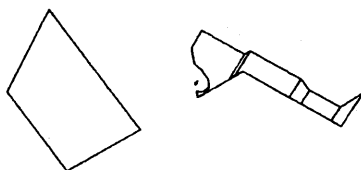
(a) 辺による表示



(b) 輪郭



(c) いくつかの柵の集合



(d) すべての柵による表示

図 1 0 柵による表示例