

物体理解における物体の見え方の推定

Extraction of visible patterns of 3D objects

村野 朋光

衰 東善

山本 栄一郎

Tomomitsu Murano

Tozen Hai

Eiichiro Yamamoto

(株式会社 富士通研究所)

(FUJITSU LABORATORIES LTD.)

Abstract — In this paper, we describe a method for automatically estimating in what shape a 3D object is seen. Specifically, we deal with convex polyhedral objects, examine how to estimate the relation between a 2D image of a certain object taken by a TV camera and faces composing the object, and present a method for automatically extracting all Visible Patterns, each of which is defined by the shapes of visible faces of a 3D object and the relations of their mutual connections. Finally we present some experimental results to show the effectiveness of our method.

Key words — image understanding, block world, Visible Pattern.

1. まえがき

画像を中心とした視覚情報処理の研究は、単に画像を処理して人間にとって好ましい画像を作る画像処理の世界から、対象世界の記述を作り、画像の表す内容までも捉えようとする画像理解の世界へと進んでいる。

画像理解の基本は、画像という情報を処理する過程において、対象世界に関する知識に基づいた推論を行い、画像が表す元の世界に関する記述を得ることにある。したがって、この実現にあたっては知識の蓄積、知識の活用、知識の管理が画像理解の能力を左右することになる。⁽¹⁾

こうした背景を踏まえ、筆者らは物体を対象とした知識の蓄積に関する検討を進め、物体の見え方を自動的に推定する方式を開発した。

この開発の主目的は、とくに物体をTVカメラ等で撮像したときに得られる画像が、物体自身のもつ三次元構造に対して、どの方向の視点から見た形状であるかを自動的に捉えることにある。

TVカメラで物体を撮像し二次元像に変換するシステムでは、物体の奥行き情報が失われる。これに対処するためのひとつの方法としては、ステレオ法、スリット法⁽²⁾⁻⁽³⁾のように、撮像系を工夫することにより奥行情報などの距離情報を入力時に直接得る方法がある。もうひとつの方法としては、物体に関するモデルを構造化することにより二次元像から三次元構造を推定する方法がある。後者の方法⁽⁴⁾⁻⁽⁵⁾は人間が写真や絵を解釈する場合に近いと考えられるので筆者らは後者の方法を採用することにした。

この方法では物体の三次元的な回転を許す場合には、その見え方が変わるために二次元像とモデルとの対応づけが困難である。筆者らの開発した見え方の自動推定方式は、物体の見え方をモデルとして記述することにより二次元像とモデルとの対応づけを容易にすることをめざしたものである。本稿では特に凸多面体について考察し、以下第2章では自動推定方式の基本的な考え方、第3章では見え方の推定、第4章では実験結果について述べる。

2. 基本的な考え方

物体をTVカメラ等の撮像機で観測した場合、そこから得られる画像は撮像機の支点や投影面位置等のパラメータによって表現されており、厳密ではあるが、その構造は複雑かつ多様な形態になる。ここでは物体をこうしたパラメータに左右されずに、可能な限り、人間の抱く概念に近いレベルで表現することを基本とした。具体的には人間が物体を見た時に見える形を、多角形の組み合わせとして自動的に推定できる方式の構築をめざした。

図1は筆者らが開発した物体の見え方を推定するための処理フローである。この処理にあたってはあらかじめ物体の構造を表すモデルを準備する(三次元形状データ)。その後、このモデルより導出するパラメータを活用して物体の見え方を推定するが、それは二段階の処理で行われる。まず、物体を構成する要素である各面に対して面の方程式を求め、それによってどの面が見えるかを抽出する(可視

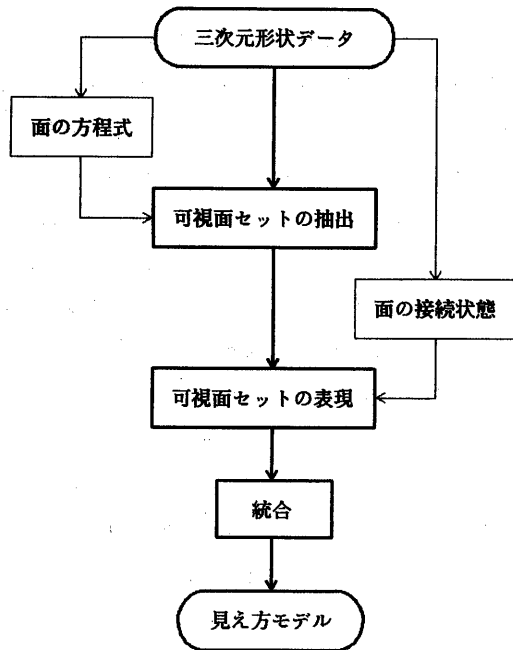


図1 見え方の推定処理フロー

面セットの抽出)。つぎに、その結果とモデルより導いた面間の接続情報とを結合することで、見える形を多角形で表現する(可視面セットの表現)。見え方モデルはこれら二段階の処理を通じて得られた多数の多角形に対して視覚的に同種であるとみなせる形どうしを統合したものである。

3. 見え方の推定

3.1 三次元形状データ

三次元形状データは物体の見え方を推定するための基本モデルであり、その構成要素は三次元空間の座標系で表現される。図2は四角錐を三次元空間に示した例であり、線($s_1, s_2, \dots, s_6$)は両端点の座標値、面($f_1, f_2, \dots, f_5$)は線によって囲まれた閉ループ単位で表される。

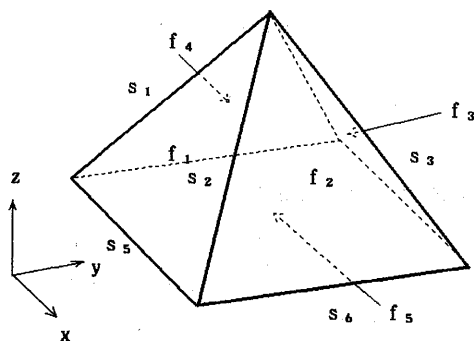


図2 四角錐のモデル

さて、この基本モデルより物体の見え方の推定に必要な各種パラメータを導出するが、面の方程式は各面ごとに次式で表される。

$$\begin{aligned}
 f_1 & ; \quad \vec{n}_1 \cdot \vec{X}_1 = d_1 \\
 f_2 & ; \quad \vec{n}_2 \cdot \vec{X}_2 = d_2 \\
 f_3 & ; \quad \vec{n}_3 \cdot \vec{X}_3 = d_3 \\
 f_4 & ; \quad \vec{n}_4 \cdot \vec{X}_4 = d_4 \\
 f_5 & ; \quad \vec{n}_5 \cdot \vec{X}_5 = d_5
 \end{aligned}$$

..... (1)

ただし、 $\vec{n}_i$ は f_i の法線、 X_i は f_i 上の点であり、 d_i は原点から f_i までの距離である。

また、面の接続状態は注目する面の構成要素である線を時計回りに順次追跡した時に、隣接する面を抽出することで求まる。表1は図2の四角錐における面の接続状態を表した例であり、注目面ごとに、その形を意味する形状属性及び接続する面とその抽出された順序列が表現される。

表1 面の接続状態

注目面	形状属性	接続関係
f_1	triangle	$f_4 \rightarrow f_2 \rightarrow f_5 \rightarrow$
f_2	triangle	$f_1 \rightarrow f_3 \rightarrow f_5 \rightarrow$
f_3	triangle	$f_2 \rightarrow f_4 \rightarrow f_5 \rightarrow$
f_4	triangle	$f_3 \rightarrow f_1 \rightarrow f_5 \rightarrow$
f_5	square	$f_1 \rightarrow f_2 \rightarrow f_3 \rightarrow f_4 \rightarrow$

3. 2 可視面セットの抽出

3. 2. 1 可視領域と不可視領域

物体の見え方は視点の位置に応じて変化するが、物体の構成要素である面が見えるか否かは、一定の範囲にある視点に対して一意的に決まる。

図3は二次元図形を仮定した時に各辺 (a, b, c) が見える視点の範囲を示しており、この図形においてはa、b、c、及びaとb、bとc、cとaの6種類が見える範囲を持ち、aとbとcとが同時に見える範囲はない。

ところで、視点を辺aだけが見える範囲に置いた時、他の辺あるいは複数個の辺の組み合わせは見えない。この時辺aに対して、それが見える範囲を可視領域、それ以外を不可視領域と呼ぶ。

3. 2. 2 領域の判定

第3. 2. 1項の論理に基づき、物体に対する可視領域の判定は、物体を構成する各面に対しての面の方程式により定式化できる。即ち、各面の可視領域、不可視領域はそれらが単独に存在する場合、次式で表わされる。

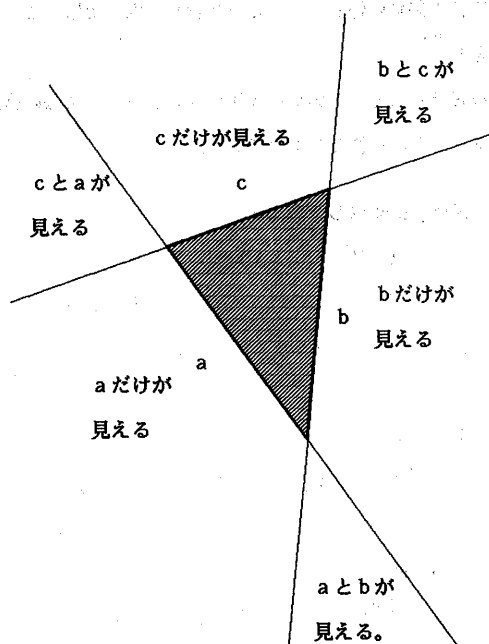


図3 可視領域

$$\begin{aligned}
 \cdot \text{可視領域} & \{ \vec{X} ; \vec{n} \cdot \vec{X} \geq d \} \\
 \cdot \text{不可視領域} & \{ \vec{X} ; \vec{n} \cdot \vec{X} < d \} \\
 & \dots \dots (2)
 \end{aligned}$$

したがって、m個の面からなる凸多面体において、1番目からk番目までの面が見え、それ以外の面が見えない状態を意味する式は、

$$\begin{aligned}
 \vec{n}_i \cdot \vec{X} & \geq d_i \quad i=1 \sim k \\
 \vec{n}_j \cdot \vec{X} & < d_j \quad j=k+1 \sim m \\
 & \dots \dots (3)
 \end{aligned}$$

となる。

3. 2. 3 解法

式(3)を解くことで、領域に対する可視あるいは不可視が判定できる。それによって見える面の組み合わせ(可視面セット)が抽出される。

式(3)は法線ベクトル $\vec{n}_i$ を係数とした3次元空間上の点を示す変数 $\vec{X}$ からなる不等式であり、数学の連立一次

不等式の問題に帰着する。これは線形計画法で解くことができる。⁽⁹⁾

次式は式(4)を線形計画法のシンプレックス法に適用できるように変換したものである。

・標準型に変形した理論式

$$\begin{aligned}\vec{n}_i \cdot (\vec{X}^+ - \vec{X}^-) - S_i &= d_i \\ i &= 1 \sim k \\ \vec{n}_j \cdot (\vec{X}^+ - \vec{X}^-) + S_j &= d_j \\ j &= k+1 \sim m\end{aligned}$$

・非負の条件

$$X_i^+, X_i^-, X_i, X_1, X_2, X_3, S_h \geq 0 \\ h = 1 \sim m$$

・目的関数

$$Z(\vec{X}) = \vec{n} \cdot (\vec{X}^+ - \vec{X}^-) \\ \dots\dots\dots (4)$$

ただし、 $\vec{X}^+$ 、 $\vec{X}^-$ は $\vec{X}$ を非負の実数の差で表すために導入した変数、 S_i は不等式を等式に変換するために導入した変数、 $\vec{n}$ はすべての可視面の法線の総和を可視面の総数で割ったベクトルである。

式(4)より3種類の形態をもった解が求まるが、表2はその解の種類(ケース1、ケース2、ケース3)である。表2において、上界は物体から遠く離れた領域、下界は物体に近接する領域、有界は領域が無限の範囲までおよばないことを示す。

表2 解の種類

	上界	下界
ケース1	有界でない	有界
ケース2	有界	有界
ケース3	解なし	解なし

したがって、ケース1は可視領域を持ち、しかもその領

域が無限の距離まで広がっていること、ケース2は可視領域をもつが、その領域はある範囲に限定されていること、また、ケース3は可視領域がないことを意味する。

3.3 可視面セットの表現

可視面は三次元形状のデータより導出される面の接続状態をもとにして表現する。図4は図2の四角錐において、面 f_1 と面 f_4 が同時に見えたと判定された可視面セットに対する表現例である。

可視面は面の接続状態から f_1 と f_4 に対応した内容を取り出し、それらの面に関する新たな接続関係ならびに形状属性を生成することで表現される。ここで生成する形状属性は、三次元形状データで表現した形状属性だけではなく、面間の相違も表される。この理由は、物体の見え方は物体を構成する面間の相対的な位置関係によって決まるため、見え方においては、同一の形状でも区別する必要があるからである。

3.4 統合

3.4.1 処理手順

一つの物体に対する可視領域は数多く存在し、各領域ごとに物体の可視面セットが抽出される。しかし、この中には人間の概念として極めて類似した形を呈するものが混在する。例えば、四角錐においては可視面セットが22個見出されるが、その中で二つの面からなる可視面セットを考えてみると、“triangle と triangle”、“triangle と square” がそれぞれ4通りずつも存在する。

統合処理ではこのように類似した見え方をもつ可視面セットどうしをまとめあげ、それぞれを“triangle と triangle”、“triangle と square”があるという表現に変換する。

この統合は可視面の表現内容を用いて実現される。具体的には可視領域ごとに、形状属性が同じである可視面をすべて選びだし、それらの各々に対して接続関係の順序列を調べる。この時、順序列が循環(ローテーション)の状態にある可視面どうしは同じ見え方であると判断し、それら

をすべて任意に選んだ可視面の表現内容に変換すればよい。

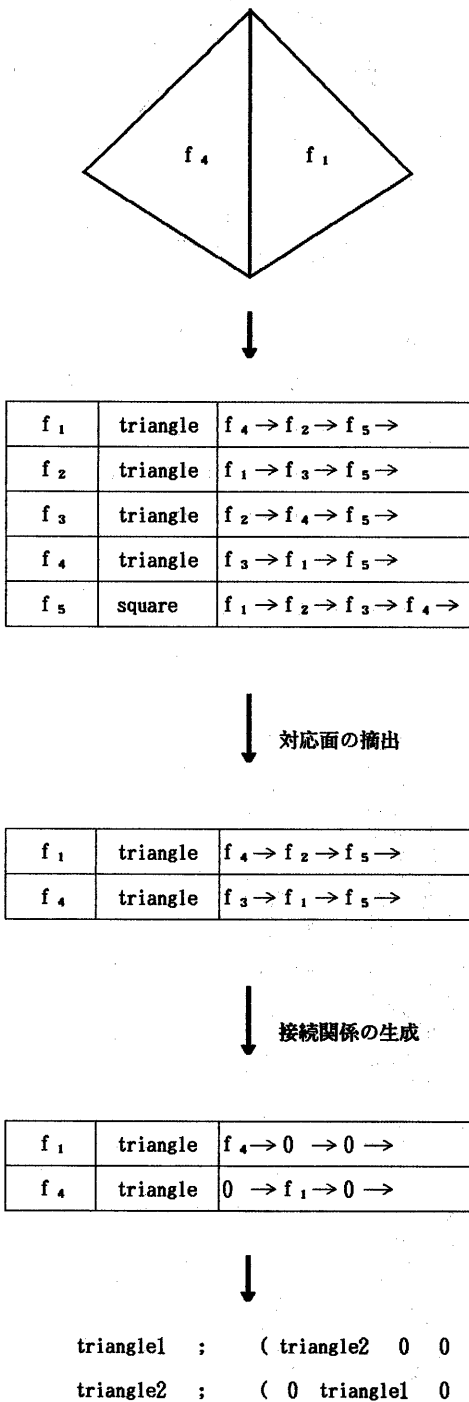


図4 可視面セットの表現例（四角錐）

3. 4. 2 ローテーションの検証

形状属性の選択は可視面の表現内容を検索することによって容易に行うことができるが、ローテーションの検証は次式で行う。

$$P(e_1 e_2 e_3 \cdots e_s) = (e_2 e_3 \cdots e_s e_1) \cdots \cdots (5)$$

ここで、 $e_1 \sim e_s$ はそれぞれ可視面の形状属性であり、 P は順序列の構成要素に対する作用素で、ローテーションを変えずに要素の順番を変える作用をもつ。したがって、何回かの P の作用で順序列が等しくなる接続関係どうしは同等とみなせる。

なお、可視面の形状属性は第3. 3節で述べたように、可視面の構成要素である面間の相違を表すために生成されている。そのため、実際のローテーションの検証においては、可視面どうしの形状属性を対応する形状（triangleはtriangle、squareはsquare、・・・）ごとに組み換えながらすべての場合に対して、式（5）を適用しなければならない。

4. 実験結果

4. 1 実験システム

FACOM M-380をホストとし、その配下にデータ入力用のタブレットと結果を表示するディスプレイからなるシステムを構築した。タブレットは三次元形状データを入力するためのもので、実際には物体を表した三面図をベクトルで入力した。M-380は入力されたベクトルをもとに面の方程式や面の接続状態などの各種データの加工⁽¹⁰⁾⁻⁽¹²⁾を行うとともに、可視面セットの抽出、可視面セットの表現、統合および表示のための図形生成処理を実行する。

なお、M-380で実行させる各種処理用プログラムはFORTRANとUTILISPで作成したが、その内容は表3のとおりである。

表3 言語と処理内容

FORTRAN	UTILISP
各種データ加工	見え方の表現
可視面の抽出	統合
	図形発生

4. 2 実験例

図5はタブレットより入力した四角錐に対し、その見え方を推定した実験結果である。

図5 (a) は入力した三面図であり、図5 (b) は三面図より生成された立体と、そこから求められた構成面の法線ベクトルである。図5 (c) は第3. 2節で述べた手法を用いて抽出した可視面セットの一覧表である。この一覧表において、UPPER 及び LOWER BOUNDはそれぞれ“上界”、“下界”、VISIBLE LISTの数値は“可視面セット”を意味し、UN- BOUNDED は“有界でない”、LOWER-BOUND の数値は“有界の位置”を表す。図5 (d) は抽出された複数の可視面セットを第3章で述べた統合処理によってまとめあげ、その結果を図示したものである。これにより、四角錐は22個の可視面セットが存在するが、それは7個の見え方にまとめられることが分かる。

図6は同様の処理を四角錐台に適用した結果であり、この場合には5個の見え方が存在する。

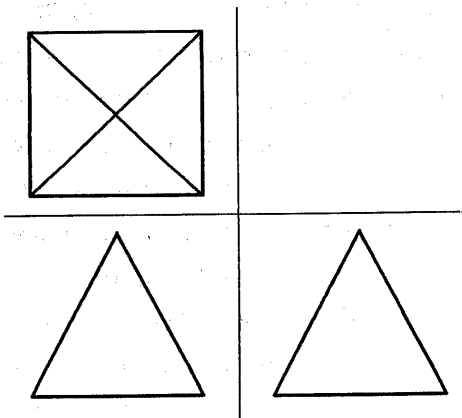
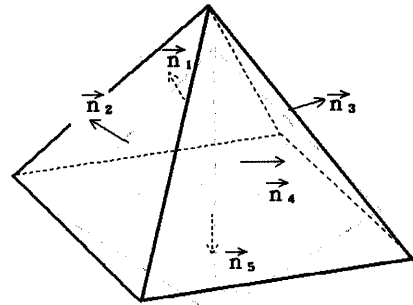


図5 (a) 入力した三面図



$$\begin{aligned}\vec{n}_1 &= (-0.848 \quad 0.000 \quad 0.530) \\ \vec{n}_2 &= (0.000 \quad -0.848 \quad 0.530) \\ \vec{n}_3 &= (0.000 \quad 0.848 \quad 0.530) \\ \vec{n}_4 &= (0.848 \quad 0.000 \quad 0.530) \\ \vec{n}_5 &= (0.000 \quad 0.000 \quad -1.000)\end{aligned}$$

図5 (b) 生成された立体と法線ベクトル

UPPER BOUND	LOWER BOUND	VISIBLE LIST
UN-BOUNDED	0.0	1
UN-BOUNDED	0.0	2
UN-BOUNDED	42.4	3
UN-BOUNDED	42.4	4
UN-BOUNDED	0.0	5
UN-BOUNDED	0.0	1 2
UN-BOUNDED	26.5	1 3
SAME-POINT	SAME-POINT	1 4
UN-BOUNDED	0.0	1 5
SAME-POINT	SAME-POINT	2 3
UN-BOUNDED	26.5	2 4
UN-BOUNDED	0.0	2 5
UN-BOUNDED	53.0	3 4
UN-BOUNDED	42.4	3 5
UN-BOUNDED	42.4	4 5
UN-BOUNDED	23.5	1 2 3
UN-BOUNDED	23.5	1 2 4
UN-BOUNDED	0.0	1 2 5
UN-BOUNDED	47.1	1 3 4
UN-BOUNDED	35.4	1 3 5
NO-ANSWER	NO-ANSWER	1 4 5
UN-BOUNDED	47.1	2 3 4
NO-ANSWER	NO-ANSWER	2 3 5
UN-BOUNDED	35.4	2 4 5
UN-BOUNDED	70.7	3 4 5
UN-BOUNDED	40.0	1 2 3 4
NO-ANSWER	NO-ANSWER	1 2 3 5
NO-ANSWER	NO-ANSWER	1 2 4 5
NO-ANSWER	NO-ANSWER	1 3 4 5
NO-ANSWER	NO-ANSWER	2 3 4 5

図5 (c) 四角錐に対しての可視面

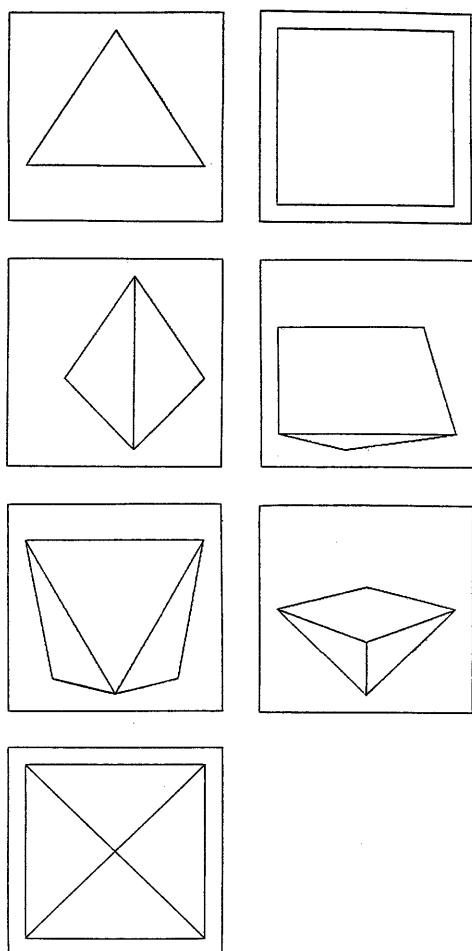


図5 (d) 四角錐の見え方

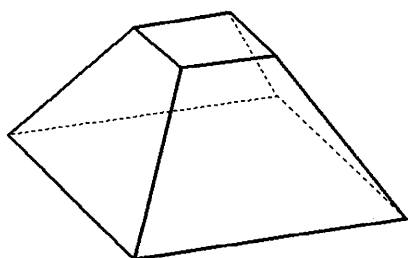


図6 (a) 四角錐台

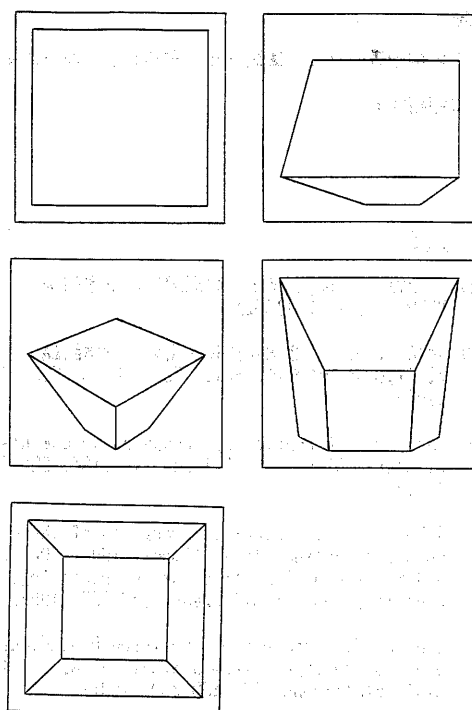


図6 (b) 四角錐台の見え方

5. むすび

本稿では物体に関する知識蓄積の一例として、凸多面体からなる物体をとりあげ、その見え方を自動的に推定する方法について述べた。

特に、ここではTVカメラ等で撮像された二次元構造の画像が、物体を構成する面に対してどのような対応関係になるかを予測するための検討を行った。その結果、物体の三次元形状データとして、線や面等の情報を入力するだけで、物体に対する様々な見え方が自動的に推定できる方式を考案し、FACOM M-380をホストとした実験システムによりその有効性を確認した。

謝辞

日頃御指導いただく峯島部長、吉田室長、並びに関連各位に深謝する。

参考文献

- (1) 大須賀, “知識工学とその周辺”, 信学技法 PRL82-51, 1-12 (1982).
- (2) 大島, 白井, “3次元情報を用いた物体認識”, 電子通信学会論文誌, Vol.J 65-D, No.5, 629-636 (1982).
- (3) Marr, D. and Poggio, T., Cooperative Computation of Stereo Disparity, Science, Vol.194, 283-287 (1976).
- (4) Roberts, L. G., Machine Perception of Three dimensional solids. In J. Tippett (Ed.), Optical and Electro-Optical Information Processing. Cambrige, Mass. : MIT Press, 157-197 (1965).
- (5) Kanade, T., Recovery of the Three-Dimensional Shape of an Object from a Single View. Artificial Intelligence 17: 409-460 (1981).
- (6) Duda, R. O., and Hart, P. E. Pattern Classification and Scene Analysis. New York: Wiley. (1973).
- (7) Kender, J. R., Shape from Texture: An aggregation transform that maps a class of texture into surface orientation. IJCAI 6, 475-480 (1979).
- (8) Barnard, S. T., Interpreting Perspective Images. Artificial Intelligence 21: 435-462 (1983).
- (9) 関根, 「数理計画法」, 岩波書店, (1976).
- (10) M. Idesawa., A System to Generate a Solid Figure from a Three View. Bull. JSME 16. 216-255 (February 1973).
- (11) George Markowsky and Michael A. Wesley, Fleshing Out Wire Frames, IBM J. Res. Develop. 24, 582-597 (September 1980).
- (12) George Markowsky and Michael A. Wesley, Fleshing Out Projections, IBM J. Res. Develop. 25, 934-954 (November 1981).