

# 知識表現法とステレオ写真解析への応用

小川 均 西淳一郎 栗岡 進 田中幸吉  
(大阪大学 基礎工学部)

## 1. はじめに

写真から、または、カメラから直接得られた画像を処理する際、原画像から得られるデータだけではうまく処理できない場合がある。このような場合、医者等の、その画像が表わす対象分野での専門家が持つ知識、および、その画像に関する特別な知識を利用することによって、うまくいく場合が多い。この時、問題となるのは、いかにこれらの知識を蓄積し、利用すればよいかということである。知識を画像処理によく用いられる FORTRAN 等のプログラミング言語で記述した場合、知識を変更するにはプログラム自体の変更が必要となり、これによるプログラミングミスも生じるおそれが十分出てくる。また、プログラミング言語に依存した考え方に陥ってしまう危険も生じる。しかも、推論が必要となる知識の利用は不可能に近い。したがって、知識を自由に扱える機構が必要となる。

本研究では、知識を表現するために  $\mu$ -actor を使用した。 $\mu$ -actor は人工知能用モジュールであり、互いの干渉にはメッセージ転送のみが許されている。この  $\mu$ -actor を用いて、定理証明機や、幾何図形に関する質問応答システムが作成されている。 $\mu$ -actor で構成されたシステムは、以下の特徴を有する。

- (1) 事実の断言と、事柄の関係や動作の定義が両方できる。
- (2)  $\mu$ -actor 間の実用上のモジュール性を保ちつつ、適当な相互作用が得られる。
- (3) 追加、変更、削除が簡単にできる。
- (4) 知識は、必要な時のみ使用できる。

本論では、知識利用の一例として、脳血管ステレオ X 線写真の血管追跡、および、そのモデルとしての針金のステレオ写真の探索の際の、交差点における処理を扱う。入力される画像の中には多数の血管(針金)が存在しており、ある一本の血管(針金)の追跡を行う時、他の血管(針金)との交差点で行き詰まったり、他の血管(針金)を間違えて追跡してしまう場合が多くみられる。そこで、交差点のデータと処理方法(これは画像処理研究者から得た)を蓄積し、必要な時に必要な処理を血管(針金)追跡システムに指示する知識ベースシステムを  $\mu$ -actor を用いて作成した。

## 2. 問題点とアプローチ

前章で述べたように、原画像から得られるデータだけではうまく処理できない場合がある。本研究では、画像処理の一例として、脳血管ステレオ X 線写真の血管追跡、および、針金のステレオ写真の針金追跡を扱う。本章では、血管(針金)追跡の失敗例、および、それを解決した知識利用システムの概要について述べる。

### 2.1 追跡失敗例

ステレオ写真における血管(または、針金)の追跡は、左右 2 枚の写真の追跡結果を比較しながら行われる(詳細は次章参照)。写真 1, 2 は、それぞれ、脳

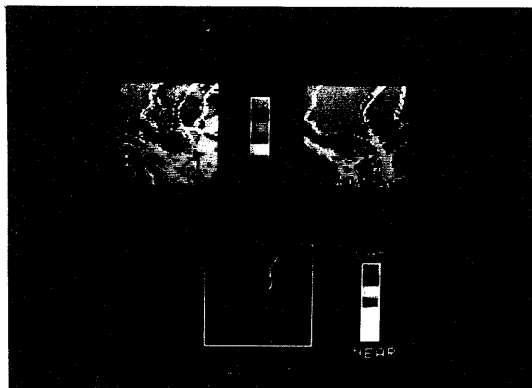


写真1  
脳血管ステレオX線写真の  
血管追跡結果（失敗例）

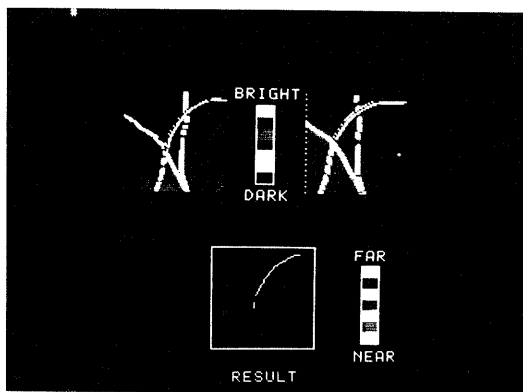


写真2  
針金ステレオ写真の  
針金追跡結果（失敗例）

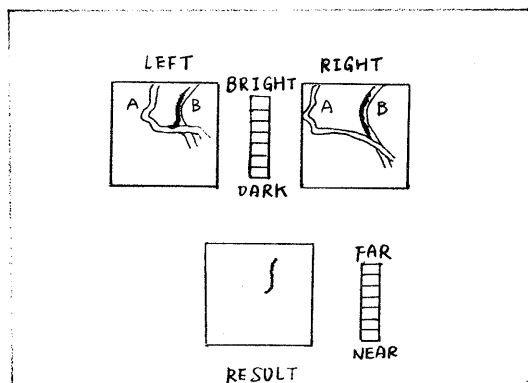


図1  
写真1の説明

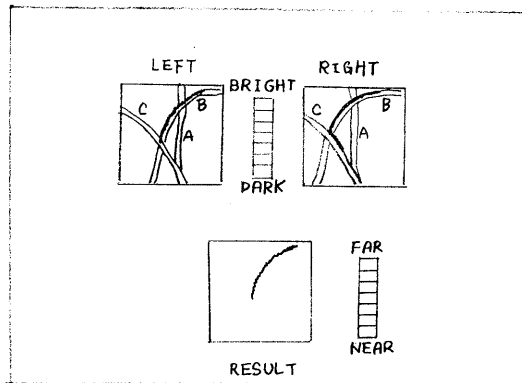


図2  
写真2の説明

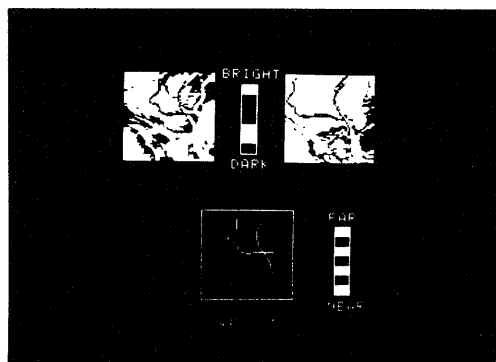


写真3  
知識ベースシステムを使用した  
血管追跡結果（成功例）

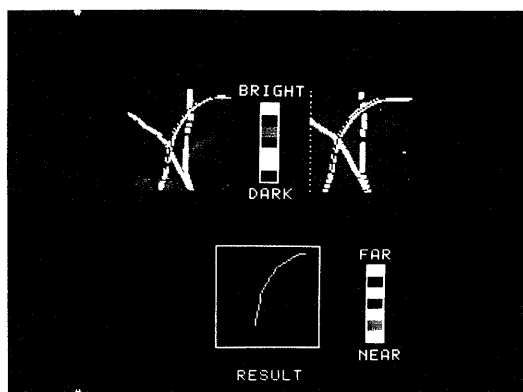


写真4  
知識ベースシステムを使用した  
針金追跡結果（成功例）

血管ステレオX線写真の血管の追跡結果、針ステレオ写真の針金追跡結果の一例である。また、写真1, 2では分りにくいため、それぞれ図1, 2に説明図を示す。各写真は3つの図から構成されている。上側左右の図はそれぞれ、左目、右目から見た図に対応している。表示の際は濃淡を擬カラー表示する。下側の図には上側左図の追跡結果を出力するが、右図追跡結果との視差を色で区別して表示する。使用者は、血管追跡の開始点および、追跡すべき方向を入力する。追跡プログラムは、北橋氏（豊橋技科大）、横矢氏（電総研）らが開発したものである。このプログラムは、ステレオ写真の左右2枚の画像の追跡点を比較しながら、両眼視差の特長を利用して、境界線を追跡するものである。これは、医用画像等のような複雑なものや、雑音が多い画像に対して有効であった。しかしながら、追跡中、他の血管（針金）との交差点で、追跡不可能になったり、間違った追跡を行う場合がある。写真1（図1）では、血管Bの追跡中、左図の交差点で血管Aに移った為、両眼視差の条件違反により、追跡を中止している。写真2（図2）では3本の針金A, B, Cのうち針金Bを追跡した例である。針金Aとの交差点では、うまくBのみを追跡しているが、Cとの交差点では誤ってCを追跡している（上側右図）。

## 2.2 アプローチとシステムの概要

血管（針金）追跡システムに交差点における処理の能力を付け加えれば、前節で示した追跡失敗は防げると思われる。我々は交差点での処理方法（これは画像処理研究者から得た）を $\mu$ -actorで表現した知識ベースシステム（KBS: Knowledge Base System）に蓄積し、必要な時に必要な処理を実行したり、血管追跡システム（BVFS: Blood Vessel Following System）に指示するようにした（図3）。

BVFSはFORTRANでプログラムされ、KBSは $\mu$ -actor上に遂行されている。 $\mu$ -actorはLISPで実現されている為、両システムを同一計算機で実行するのは困難である。そこで、2台の計算機を使用した（図4）。すなわち、KBSをFACOM 230/45S (256KB)、BVFSをPANAFACOM U-300 (32KB) でそれぞれ実行した。また、探索結果をPANAFACOM U-300に接続されているカラー・グラ

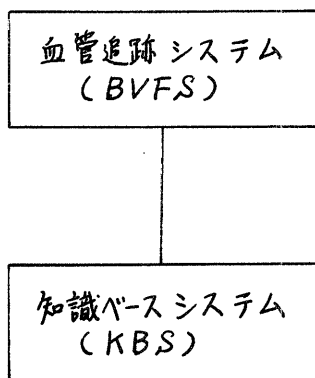


図3  
システムの概要

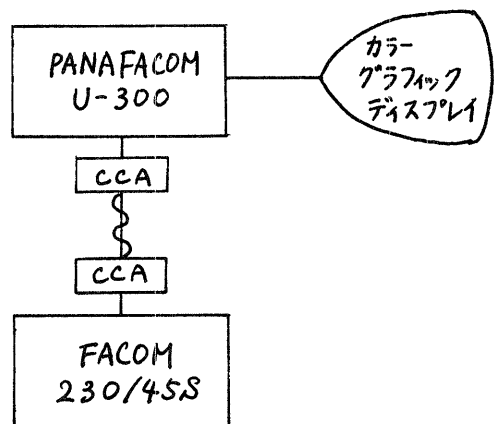


図4  
ハードウェア構成

ウィック・ディスプレイに出力した。

### 2.3 システム全体の動き

BVFSは画像データの処理に関するプログラムを保有しており、必要に応じて、すなわちKBS側の指示に従って、それらを使用する。

KBSには、画像中の血管の交差点の座標および、クロスエリアの大きさを与えておく。クロスエリアとは、交差点を中心としたある一定の領域である。現在のところ、交差点は別の画像処理プログラムの結果から得ている。またクロスエリアの大きさも使用者があらかじめ決定したものをKBSに与えている。しかしながら、これらのデータを原画像から自動的に集めることも可能である。

このシステムでは血管追跡を行っていく過程において、状態を“Normal”，“Left-Blind”，“Right-Blind”，“Both-Blind”の4つの場合に分類して考える。個々の状態について以下に説明する。

1) Normal 左右両画像において、両方の追跡点がクロスエリアに入っていない状態。

2) Left-Blind 左画像における追跡点がクロスエリアに入っており、かつ右画像での追跡点がクロスエリアに入っていない状態。

3) Right-Blind Left-Blindのちょうど逆の状態。

4) Both-Blind 左右両画像の追跡点がどちらもクロスエリアに入っている状態。

以下に、BVFSとKBSの4つの状態における動作について述べる。

1) Normalでの動作 BVFSは、3つの前提条件と画像データによる濃淡微分値を用いることによって、現在の追跡点に続く新しい点を見つける。(前提条件および濃度微分値については次章で説明する。)

BVFSは、追跡点をKBSに報告する。このときKBSがその点を受け取るだけであれば、BVFSは状態をNormalとみなす。

一方、KBSでは、その時受け取る1つ前の点についての調査をすでに終えている。すなわち、4つの状態のうち1つを決定しているわけである。もし、状態がNormalならば、新しい点を受け取りそれについて調べる。この時、2つのシステムは全く別の行動を並列して行っている。Normalでなければ、次に述べる、2)、3)、4)の動きをする。

2) Left-Blindでの動作 KBSは、左側画像における追跡点は信頼性が低いとみなす。そして、右側画像のみの追跡をBVFSに指示する。この指示を受けたBVFSは、左側画像のデータを使用せずに、右側画像のデータをより詳しく解析することにより新しい追跡点を見つけ、KBSに知らせる。KBSは、この右側画像の点から左側画像の追跡点を決定する。

3) Right-Blindでの動作 Left-Blind状態の動作を左右反対にして行う。

4) Both-Blindでの動作 KBSは、画像データから得られる情報の信頼性は低いとみなす。そのため、BVFSが持っているプログラムは使わずに、過去の追跡点のパターンから左右それぞれの点を決定する。

2)、3)、4)の状態が終了した後は、Normal状態に戻る。そして新しい追跡点を見つけたあとでその状態にあった処理を行う。以下、この動作を繰り返して次々と追跡点を決定していく。

上記のシステムにより、前節で示した例を追跡した結果を写真3, 4に示す。写真3は、A, B両血管の追跡結果であるが、両血管とも正しく追跡されている。また、写真4においても針金Bが正しく追跡された。

### 3. 血管追跡システム

血管追跡システム (BVFS) の役目は、与えられた画像データから、前提条件および濃度微分値を用いることによって血管の3次元境界線の追跡を行うことである。この場合、BVFSは左右画像データのうす、どこがクロスエリアかなどという全体的な視野はもたない。BVFSは常に求めた点を知識ベースシステム (KBS) に送る。そして、そのKBSの判断によって決定された4つの状態に応じて、次の点を求める方法を変えるだけである。

ステレオ写真撮影モデルは、横矢氏の方法を採用している。ここでは、奥行量を  $l$ , 両眼視差を  $d$ , レンズ間距離を  $b$ , 焦点距離を  $f$  とすると、

$$l = \frac{b \cdot f}{d}$$

なる関係を用いている。詳細は文献(1)(2)を参照されたい。

#### 3.1 前提条件

左右画像は、それぞれ量子化された濃度レベルをもつ点の集合  $G_L, G_R$  からなるものとする。以下、添字の  $L, R$  は左側, 右側を示している。左右画面の境界線追跡開始点, 終了点はすでに与えられており、それぞれを  $P_L^0, P_R^0, P_L^E, P_R^E$  で表わす。ここで、始点は  $B$  で、終点は  $E$  で表わす。追跡開始後、次の番目に得られる点を、左右それぞれ  $P_L^1, P_R^1$  とし、その座標を次のようにする。

$$P_L^1: (x_L^1, y_L^1)$$

$$P_R^1: (x_R^1, y_R^1)$$

ただし、座標は各画面左上隅に原点をもち、水平右方向に  $x$  軸, 垂直下方向に  $y$  軸をとる。

境界線追跡の際に、以下のような3つの前提条件を置く。これは、血管の性質および奥行量計算に関するものである。

1) 前提条件 I 奥行量の計算は両眼立体視原理に基づき、外界の一点が左右画面上に写す像のずれ(視差)による。このとき、ずれは2つのX線管球の焦点を結ぶ直線に平行な方向、今の例では  $x$  軸方向にのみ生じ、それに垂直な方向  $y$  軸方向には生じない。

視差  $d$  は、左右画面上の対応点が  $P_L = (x_L, y_L), P_R = (x_R, y_R)$  であるとするならば、

$$d = x_L - x_R$$

から求めることができる。ただし、上に述べたように、

$$y_L = y_R$$

なる関係がある。

2) 前提条件 II 血管を示す曲線は、3次元的に滑らかである。血管は急激に曲ることはない。

3) 前提条件 III 同一領域内での濃度レベルの差は異なる領域に存在する画素間の濃度差より小さい。

具体的に例を示すと、図5においてA点とB点との濃度レベルの差は、A点と

C点とのそれよりも小さいということである。この条件は、追跡している同一血管内での濃度レベルは大きく変化しないと考えられるからである。

### 3.3 BVFSの動作

KBSの判断によって、状態が決定されるが、それら4つの状態におけるBVFSの動作を説明する。

#### 3.3.1 Normal

BVFSは前節で述べた前提条件を用いることにより、次の追跡点の候補をしばらく、1段先読みにより、追跡点を決定する。そして、BVFSは左右対になった点をKBSに送る。もし、このときKBSが何の指示も出さないならば、BVFSは状態をNormalとみなし、上に述べた処理を再び行う。

#### 3.3.2 Left-Blind

KBSから“今はLeft-Blind状態だ”という指示が与えられたとする。すると、BVFSは右側画像のみで追跡を行い、追跡点を決定する。

片側画像だけの追跡の場合、前節で挙げた前提条件Iがはずされる。制限事項がなくなれば、それだけ正確な追跡をすることが難しくなるので、この場合、2段先読みをして追跡点を決定するようにした。

BVFSが、右側の追跡点を送ることによって、KBSでは、左側の座標を計算する。そして、BVFSは計算された左側の点を受け取る。というのは、BVFSが次の追跡点を求める際に、追跡方向を考慮するからである。

以上のことから、Left-Blindの場合、先にNormal状態で決定した追跡点は、左右両方とも無駄になる。

#### 3.3.3 Right-Blind

KBSから、“今はRight-Blind状態だ”という指示がきたならば、Left-Blindのときとは左右逆の動きをする。

#### 3.3.4 Both-Blind

2の場合も、Normal状態で捜し出した点は信頼性が低いということでも無駄になる。BVFSは、この状態では何もせず、ただKBSが計算した左右の対になった点を受け取るだけである。

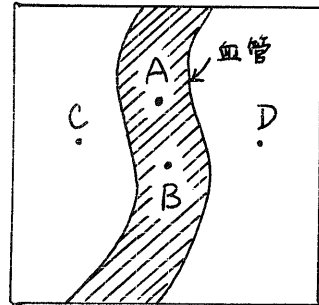


図5  
写真のモデル

## 4. 知識ベースシステム

知識ベースシステム(KBS)はBVFSに適切なアドバイスを行うシステムであり、 $\mu$ -actorによって実現されている。初めに、 $\mu$ -actorについて簡単な説明を行い、KBSがどのような知識を蓄えているか述べる。

### 4.2 $\mu$ -actor

人間の社会では、1つの大きな問題を解く場合、各小さな問題に関するエキスパートがそれぞれ適当な情報交換を行いながら、各自の責任を果たすことによって目的を達成している。 $\mu$ -actorシステムにおいては、各 $\mu$ -actorが各エキスパートに対応している。 $\mu$ -actorの行動は、行動情報部とデータ部の2つの部分によって定義される。行動情報部は、その $\mu$ -actorが受理できるメッセージのパターンと、そのパターンに照合するメッセージが送られた時に行うべき行動の組を

幾つか持っている。行動は、前提条件、付随条件、発生事象の3つの部分からなるルールによって構成される。データ部は、その $\mu$ -actor固有の値や、記憶すべきデータを蓄積する場所である。現在のところ、Minsky (MIT)の提案したFrame表記を用いているが、各 $\mu$ -actor毎に形式が変わってもよい。 $\mu$ -actorはメッセージが送られることにより、行動を開始し、必要な場合は他の $\mu$ -actorにメッセージを送り、データを得たり、仕事の一部を依頼する。

この $\mu$ -actorを用いて定理証明機や、幾何図形に関する質問応答システムが開発されている。 $\mu$ -actorはLISPで実現され、FACOM 230/45Sで遂行されている。 $\mu$ -actorの構造、動作等についての詳細は文献(3)~(5)を参照されたい。

#### 4.2 KBSが蓄えるべき知識とその実現

KBSが持っている知識は、血管(針金)追跡の際の交差点での処理に関するものである。その概略は以下のようになる。

“血管追跡を行っている途中で、左右どちらかが交差点に近づいた時には、交差点に近づいていない方の画面の情報を用いる。また、左右両方共交差点に近づいている場合には、それまでに追跡した点から新しい点を推測する。”

4つの状態を用いて、KBSの処理をより具体的に述べると次のようになる。

“BVFSから送られた左右画面の追跡点の座標をそれぞれ $P_L$ ,  $P_R$ とする。

$P_L$ ,  $P_R$ の両方とも交差点に近いならばBoth-Blindの処理を行う。

$P_L$ が交差点に近いならばLeft-Blindの処理を行う。

$P_R$ が交差点に近いならばRight-Blindの処理を行う。

上記以外ならばNormalの処理を行う。”

上で述べたことを知識として実現するために1つの状態に1つの $\mu$ -actorを割り当てる。そのため、各状態を独立したものとみなしてプログラムすることが可能である。 $\mu$ -actorを用いてシステムを構成する場合、それぞれの状態の処理を複数個の $\mu$ -actorで行うこともできる。各 $\mu$ -actor間の干渉はメッセージ転送のみであり、各 $\mu$ -actorは個々の役割を果たすということも考慮すると、KBSの処理の概略は以下のように書く。 $\mu$ -actor名には下線を引く。

“BVFSから送られた左右画面の追跡点の座標をそれぞれ $P_L$ ,  $P_R$ とする。

LEFT-CR に $P_L$ , RIGHT-CR に $P_R$ を送る。両方の返答がOKであれば、BOTH-BLIND に処理を依頼する。

LEFT-CR からの返答のみがOKであるならばLEFT-BLIND に処理を依頼する。

RIGHT-CR からの返答のみがOKであるならばRIGHT-BLIND に処理を依頼する。

```
( (MES: ?%N MAIN ?A ?B TO: ?CONT)
  ( (P-C: (=> LEFT-CR (MES: !%N BLIND? !A TO: ME))
    (=> RIGHT-CR (MES: !%N BLIND? !B TO: ME)) )
    (C-E: (=> BOTH-BLIND (MES: !%N BOTH-BLIND TO: !CONT))) )
  ( (P-C: (=> LEFT-CR (MES: !%N BLIND? !A TO: ME)) )
    (C-E: (=> LEFT-BLIND (MES: !%N LEFT-BLIND !B TO: !CONT))) )
  ( (P-C: (=> RIGHT-CR (MES: !%N BLIND? !B TO: ME)) )
    (C-E: (=> RIGHT-BLIND (MES: !%N RIGHT-BLIND !A TO: !CONT))) )
  ( (P-C: )
    (C-E: (=> STACK-POINT (MES: !%N STACK-POINT !A !B TO: !CONT))) )
)
```

#### 図6

$\mu$ -actor MAIN-ACTORの行動情報部  
(KBSの処理の概略)

その他の場合は、STACK-POINTに $P_L$ と $P_R$ を送り、点を蓄積する。”  
ここで、LEFT-CRとRIGHT-CRはそれぞれ左右画面の追跡点が交差点に近いかどうかを調べる $\mu$ -actorである。BOTH-BLIND、LEFT-BLIND、RIGHT-BLINDはそれぞれの状態における処理を行う $\mu$ -actorである。STACK-POINTは追跡点を記憶する $\mu$ -actorである。

上記の処理は簡単に、図6のように表現される。これは、 $\mu$ -actor MAIN-ACTORの行動情報部となる。

## 5. おまじ

本論では、画像処理への知識表現の応用を取り上げ、知識表現に $\mu$ -actorを用いた。 $\mu$ -actorによる知識表現は基本的には、手続き的表現であるが、宣言的な使い方も可能である。このことから、手続き的表現と宣言的表現の長所、すなわち、相互作用性とモジュール性をうまく組み合わせたシステム作りが可能である。

本研究では、脳血管ステレオX線写真の血管追跡の際の血管の交差点（および針金ステレオ写真の針金の交差点）における処理についての知識（ここでは、画像処理者の知識）を $\mu$ -actorで扱った。BVFSは大久保氏（富士通）が阪大大学院在学中に開発した血管追跡システムを修正したものであり、システム全体の動きは、同氏の考えを使用した。

血管の追跡プログラムを記述するにはFORTRANが適している。一方、 $\mu$ -actorはLISPで実現されている。このため、システム全体を1つの計算機で実現するのは困難である。また、大容量の主記憶が必要なことから、PANAFACOM U-300でBVFSをFORTRANで実現し、FACOM 230/45SでKBSを $\mu$ -actorで実現した。これにより、Normal状態では、血管の追跡と得られた点のチェックが並列して行えた。

本論で扱った知識は、画像処理方法に関するものであり、同様のことはアセンブラ言語やFORTRANだけで実現が可能である。しかしながら、本研究の目的は人間の考えたことを、その構造に近い形で計算機上に実現することである。このような研究の発展として、より高度な専門知識（血管追跡の場合は医学知識）を推論を用いることにより理解し、さらにその知識を画像処理等の他の研究分野へ応用することが考えられる。また、自然言語解析機を使用すれば知識の入力も簡単になり、計算機に不慣れな他分野の専門家も自由に使えることが期待される。

最後に、日ごろ御討論、御協力下さる豊橋技術科学大学北橋忠宏助教授、電総研横矢直和氏、なまびに阪大基礎工学部田中研究室諸氏に感謝する。

### 〔参考文献〕

- (1) 横矢, 北橋, 田中: 簡単なデータ表現を用いた両眼視に基づく距離情報の抽出, 情報処理, Vol. 18, No. 3, pp. 245~252 (1977).
- (2) 北橋, 大久保, 横矢, 田中, 西村, 木村, 藤野: 対話形3次元情報抽出・表示システム, 信学会技報, MBE 78-4, pp. 29~35 (1978).
- (3) 小川, 木島, 田中:  $\mu$ -actorの実現と知識表現の構造, 情報処理, Vol. 19, No. 9, pp. 846~851 (1978).
- (4) 小川, 田中: 変数を共有する問題の擬似並列処理, 情報処理, Vol. 19, No. 9, pp. 839~845 (1978).
- (5) Ogawa, Nanba, Tanaka: An Active Frame For The Knowledge Representation, Proc. of IJCAI-79 (1979).