

モデルベースロボットビジョンVLSIプロセッサの構成

佐々木 慶文, 亀山 充隆
東北大学工学部

〒980 宮城県仙台市青葉区荒巻字青葉
東北大学工学部情報工学科

あらまし

自律動作を行なう知能ロボットにおいて、対象物体の認識およびその3次元距離情報の獲得は、極めて重要な処理である。さらに、これらの処理はスループットが高いだけでなく、演算遅れ時間が小さいという意味での高速処理が要求される。本稿では、対象物体のモデルが既知である場合に、モデルマッチングにより3次元計測および3次元物体認識が同時可能であるモデルベースロボットビジョンVLSIプロセッサを提案する。提案するVLSIプロセッサは、演算遅れ時間を小さくするために、アルゴリズムの持つ並列性に着目した高並列アーキテクチャとしている。これにより、28.5MIPSのワークステーションと比較して10000倍の高速処理が可能である。

和文キーワード 専用VLSI, ビジュアルフィードバック, 3次元距離情報, モデルマッチング, 透視投影

Design of a Model-Based Robot Vision VLSI Processor

Yoshifumi Sasaki and Michitaka Kameyama
Faculty of Engineering, Tohoku University

Department of Information Engineering
Faculty of Engineering, Tohoku University
Aoba, Aramaki, Aoba-ku Sendai-shi, 980 Japan

Abstract

For intelligent robots, a robot vision system is usually required to perform three-dimensional(3-D) position estimation together with object recognition at high speed. In this paper, we propose an algorithm for 3-D object recognition and position estimation suitable for the implementation of the VLSI processor. The principle of the algorithm is based on the model matching between an input image and the models stored in a memory. Because of enormous computation time, the development of a high-performance VLSI processor is essential. Highly parallel architecture is introduced in the VLSI processor to reduce the latency using parallelism of the algorithm. As a result of highly parallel computing, the computation time becomes 10000 times faster than that of a 28.5 MIPS workstation.

英文 key words Special-purpose VLSI, Visual feedback, 3-D distance information, Model matching, Perspective projection

1 まえがき

変化する環境に迅速に対応しながら自律動作を行なう知能ロボットを実現するためには、ビジュアルフィードバックなどのセンサフィードバックを密に行なう必要がある。さらに、ロボットビジョンではビジュアルフィードバックの中に含まれる種々の処理はスループットのみではなく、演算遅れ時間が小さいという意味で高速処理する必要がある。しかしながら、これらの処理は計算量が極めて膨大であり、従来の汎用計算機などでは実現が困難であるといわれている。これに対して筆者らは、専用VLSIプロセッサの開発による超高速処理の実現を提案し、このような研究分野をロボットエレクトロニクスと呼んでいる[1][2]。本稿では以上のような観点から、モデルベースロボットビジョンVLSIプロセッサを提案している[3][4]。

ロボットビジョンにおいては、対象物体を認識し、その3次元距離情報を得ることが、極めて重要な処理となっている。このロボットビジョンのアルゴリズムとして、様々なものが提案されているが、複雑かつ計算量が膨大となりがちである。これに対し、あらかじめ記憶してあるモデルと対象物体とのマッチングにより3次元計測および3次元物体認識を同時に行なうことができるモデルベースロボットビジョン (Model-Based Robot Vision: MBRV) アルゴリズムを提案している[5][6]。

提案するアルゴリズムでは、対象物体がカメラから2次元シルエット画像 (入力画像) で取り込まれるため、モデルに種類、姿勢および位置の各候補を与え、透視投影により等価な2次元シルエット画像 (投影画像) に変換する。これらの入力画像と投影画像とを照合し、その類似度を計算する。モデルに与えた候補と対象物体の種類、姿勢および位置が一致する場合、類似度は最大となる。従って、すべての候補に対して類似度を総当り的に計算し、最大類似度の候補を対象物体の種類、姿勢および位置として出力すればよい。

このアルゴリズムにおいて候補数は極めて膨大であり、これに比例して計算量が膨大となる。しかしながら、アルゴリズムは比較的簡単な演算の繰り返しによる処理であり、ハードウェア化に適合する。従って、提案するVLSIプロセッサによる超高速処理が可能である。また、提案するアルゴリズムでは、各候補に対する類似度の計算は並列に実行することが可能である。従って、本プロセッサはこのアルゴリズムの並列性を充分活用できるような高並列アーキテクチャとしている[7][8]。以上のVLSIプロセッサ化により、28.5MIPSの汎用ワークステーションと比較して10000倍以上の高速処理が可能となる。

2 MBRVアルゴリズム

提案するロボットビジョンシステムを図1に示す。

以下の図およびアルゴリズムに関する議論において、サブスクリプト c はカメラの焦点を原点とする計測座標系の座標、 m はモデルを定義するためのモデル座標系の座標、 s は投影面の座標であることを表す。また、サブスクリプト t は上端点に関する座標、 b は下端点に関する座標、 l は左

端点に関する座標であることを表す。

システム内には、対象物体の表面離散点座標をモデルとして記憶してある。これらの座標は、モデルを定義するモデル座標系の座標である。対象物体の2次元シルエット画像 (入力画像) がカメラより取り込まれる。この入力画像とモデルとのマッチングにより3次元計測および3次元物体認識を行なう。マッチングのため、モデルに種類、姿勢および位置の候補を与える。

種類の候補は、あらかじめ記憶されている物体ごとのモデルを1種類選択することにより与える。姿勢の候補は、モデルをモデル座標系 Y_m 軸回りに θ 回転させることにより与える。また位置の候補は、モデル座標系原点 O_m の計測座標系座標 (x_{co}, y_{co}, z_{co}) を指定する。

候補を与えたモデルを透視投影により、入力画像と等価な2次元シルエット画像 (投影画像) に変換する。入力画像と投影画像とを画素ごとに照合し、式(1)に示すように一致した画素数を類似度と定義する。

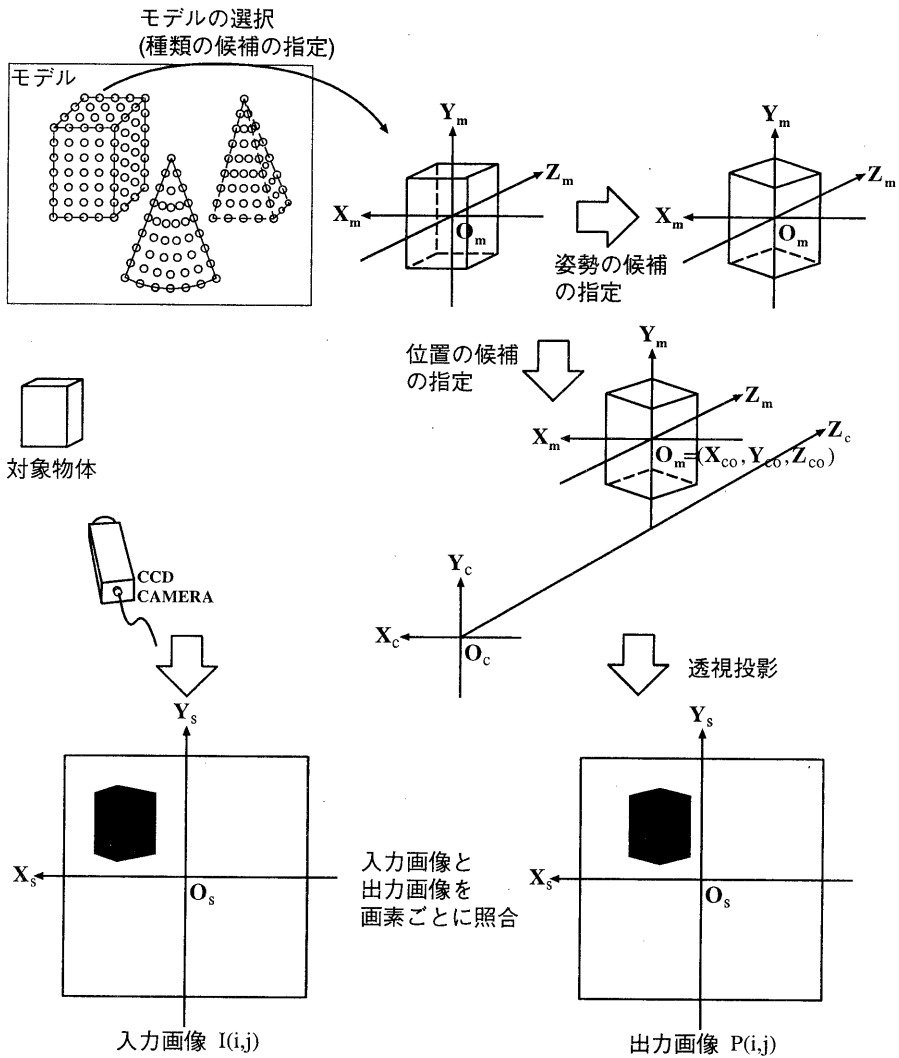
$$SI = \sum_{i=1}^M \sum_{j=1}^N \{I(i, j) \text{ xnor } P(i, j)\} \quad (1)$$

ここで、 $I(i, j)$ 、 $P(i, j)$ はそれぞれ、入力画像および投影画像の各画素であり、 M 、 N は投影面を構成する x, y 方向のそれぞれの画素数である。また、 $I \text{ xnor } P$ は I と P の一致論理関数である。モデルに与えた候補と対象物体の実際の種類、姿勢および位置が一致する場合には、類似度が最大となることは明らかである。従って、すべての候補に対し、類似度が最大となる候補を総当り的に探索すればよいことになる。

このアルゴリズムでは、モデルマッチングに基づき3次元計測を行なうため、3次元物体認識が同時に行なえるという特長を持つ。また、入力画像は1方向からみた像でよいので、簡単な構成でロボットビジョンシステムを実現できる。しかしながら、すべての種類、姿勢および位置の候補に対して、総当り的に類似度を計算するため、計算量が極めて膨大となる。従って、高速に3次元計測および物体認識を行なうためには計算量を減少させる工夫が必要である。

2.1 高速アルゴリズム

計算量を減少させるため、本稿では入力画像の端点の座標を用いて、モデルに与える位置の候補について不要な探索を解消したアルゴリズムを提案する。入力画像の上端点、下端点および左端点はそれぞれ図2に示すように最大の y_s 座標 (y_{st})、最小の y_s 座標 (y_{sb}) および最大の x_s 座標 (x_{st}) をもつ部分として得られる。モデルに与えた候補と対象物体の種類、姿勢および位置が一致する場合、入力画像と投影画像の各端点は一致する。このとき、図3に示すように、モデルは計測座標系原点 O_c とそれぞれの端点を通る3つの平面に同時に接するような位置に存在する。従って、各平面と接するモデルの表面離散点座標 $(*, y_{mt}, z_{mt})$ 、 $(*, y_{mb}, z_{mb})$ および $(x_{ml}, *, z_{ml})$ を透視投影の逆問題に基づき探索すれば、式(2)からモデルがこれら3つの平面に同時に接する場合の $(*, y_{mt}, z_{mt})$ の計測座標系座標 $(*, y_{ct}, z_{ct})$ を計算することができる。ここで、*は任意の座標であることを表



$$\text{類似度 } SI = \sum_i \sum_j (I(i,j) \text{ xnor } P(i,j))$$

図1: MBRVロボットビジョンアルゴリズム

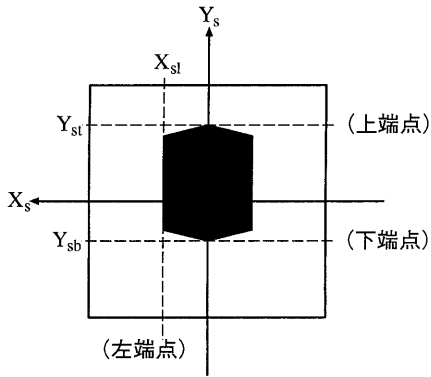


図 2: 入力画像の端点

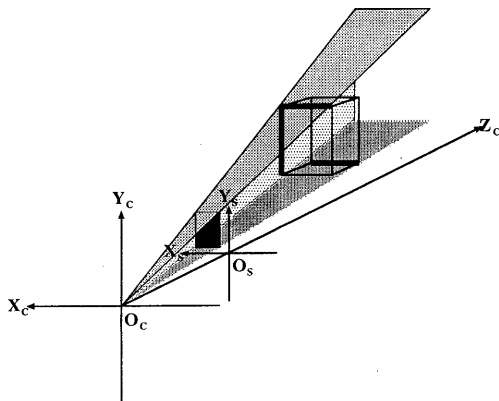


図 3: モデルの位置の候補

す記号である。

$$z_{ct} = \frac{(y_{mt} - y_{mb})}{(y_{st} - y_{sb})} h - \frac{(z_{mt} - z_{mb})}{(y_{st} - y_{sb})} y_{sb} \quad (2)$$

$$y_{ct} = \frac{y_{st}}{h} z_{ct}$$

さらに式 (3) に基づき、モデル座標系原点 O_m に与える計測座標系座標 (x_{co}, y_{co}, z_{co}) を計算することができる。

$$x_{co} = \frac{x_{sl}}{h} (z_{ct} + z_{ml} - z_{mt})$$

$$y_{co} = y_{ct} - y_{mt} \quad (3)$$

$$z_{co} = z_{ct} - z_{mt}$$

このように高速アルゴリズムでは入力画像の端点に投影されるモデルの表面離散点座標を探索し、透視投影の逆問題に基づき与えるべき位置を計算することが可能である。

2.2 アルゴリズムの評価

まず、計測誤差について述べる。提案するロボットビジョンでは、入力画像が離散画像であるために誤差が生じる。実験から、奥行き 800mm 前後で x, y 方向それぞれ 10mm 程度の誤差で計測が行なえることを確認している。

次に、計算時間について評価を行なう。総当りの探索および高速アルゴリズムを用いた場合の計算時間はそれぞれ (総当りの探索)

$$[\text{種類}] \times [\text{位置}] \times [\text{姿勢}] \times [\text{計算時間/候補}]$$

(高速アルゴリズム)

$[\text{種類}] \times [\text{姿勢}] \times [\text{計算時間/候補}]$ となる。1つの候補に対する計算時間は高速アルゴリズムの方が大きい、位置の候補数の倍数分だけ飛躍的に探索回数が減少しているため、かなりの高速化が実現できる。実験により測定したそれぞれの計算時間を図 4 に示す。図

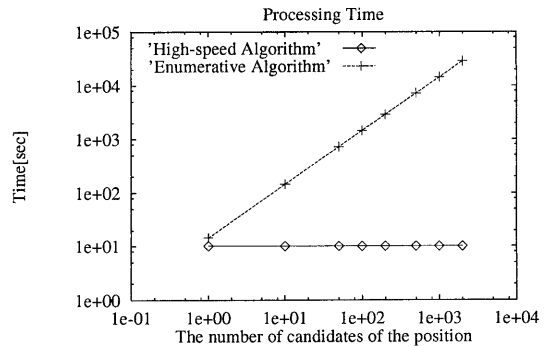


図 4: 計測時間

において横軸は位置の候補数、縦軸はその候補数における計測時間を示す。ここで、計測における入力画像および投

影画像の画素数は 256×256 とし、計算は 28.5MIPS の汎用ワークステーションにより行なっている。また、高速アルゴリズムの効果を確認するため、種類および姿勢の候補数をそれぞれ 1 としている。総当りの探索の場合、位置の候補数に比例して計算時間が増大するのに対し、高速アルゴリズムでは計算時間は一定である。位置の候補数は、計測精度および計測範囲により決定されるパラメータであり、高精度かつ広範囲に計測を行う場合、位置の候補数はそれに比例して大きくなる。例えば計測範囲を

$$\begin{aligned} -5\text{mm} &\leq x_c \leq 5\text{mm} \\ -5\text{mm} &\leq y_c \leq 5\text{mm} \\ 0\text{mm} &\leq z_c \leq 10\text{mm} \end{aligned}$$

とし、これを 1mm 間隔の精度で計測する場合、位置の候補数は 1000 となる。このとき総当りの探索による計算時間は、28.5MIPS のワークステーションで 2 時間程度となる。これに対し、高速アルゴリズムではわずか 5 秒程度で計算できることがわかる。

しかしながら、実際には種類および姿勢の候補数を考慮に入れる必要があり、例えば、候補数が 1800 の場合には上記のワークステーションで 5 時間程度を要する。知能ロボット実現のためには、ロボットビジョン以外にも軌道計画や制御などの処理が必要であるため、3次元計測および3次元物体認識は数 msec ~ 数 sec で行なう必要がある。従って、専用 VLSI プロセッサによる超高速処理が不可欠である。

3 VLSI プロセッサの構成

知能ロボットは実世界との相互作用があるため、提案する VLSI プロセッサはスループットだけではなく演算遅れ時間が小さいという意味で高速に動作するものが要求される。提案するロボットビジョンアルゴリズムでは、膨大な数のデータを繰り返し処理する。また、データは常時メモリから呼び出すことが可能である。このような場合には、パイプライン処理を行なうことで演算遅れ時間が短縮できる。また、類似度の計算が互いに独立に行なえることから、複数のプロセッサによる空間的並列処理を併用することでさらに大幅な遅れ時間の短縮が可能となる。

3.1 プロセッサの構成

提案するアルゴリズムのフローチャートを図 5 に示す。このフローチャートに基づき、 10^3 以上のすべての候補数に対し、繰り返し類似度を計算する。ここで、処理すべきデータはモデルの表面離散点座標であり、常時メモリバッファから読み出すことができる。従って、図 6 に示すように、各処理部に PE およびメモリバッファを設け、パイプライン処理を行なうことで演算遅れ時間を短縮することができる。また、各処理部は 1 つの候補当り、 10^3 点もの表面離散点座標を繰り返し処理するため、それぞれ専用の PE を設けて超高速処理を行なう必要がある。

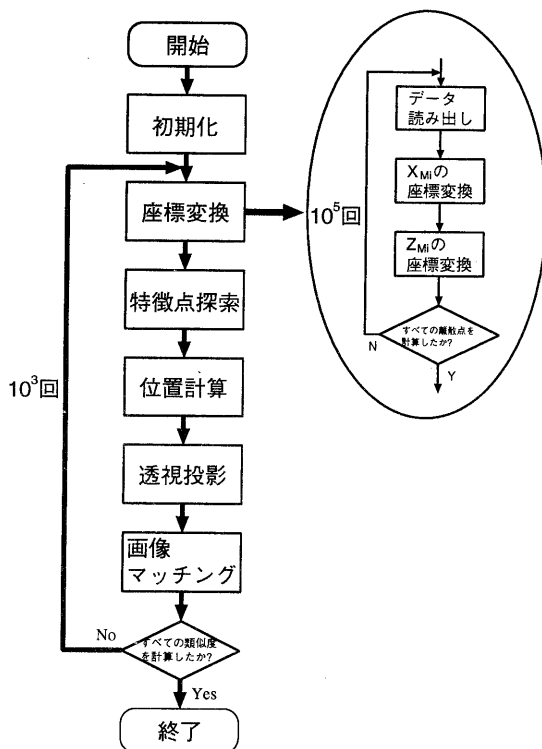


図 5: 高速アルゴリズムにおけるデータの流れ

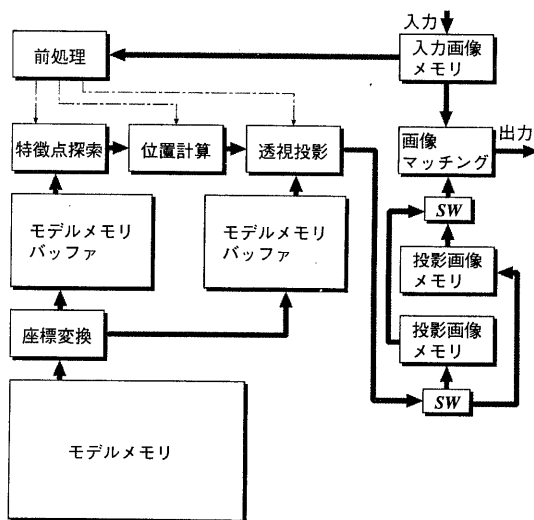


図 6: VLSI プロセッサの構成

3.2 PEの構成

一例として特徴点探索用PEについて説明する。特徴点探索は式(4)により行なわれる。

$$\begin{aligned}
 y_{dt} &= y_{mti} - \frac{z_{mi} - z_{m0}}{h} y_{st} - y_{mt0} \\
 y_{db} &= \frac{z_{mi} - z_{m0}}{h} y_{sb} + y_{mb0} - y_{mbi} \\
 x_{dt} &= x_{mi} - \frac{z_{mi} - z_{m0}}{h} x_{st} - x_{m0}
 \end{aligned} \quad (4)$$

(1 ≤ i ≤ N_m)

ここで N_m はモデルの表面離散点数である。上式に基づき図7に示すデータフローグラフを作成し、演算遅れ時間が小さい処理手順を決定する。基本的にはこのデータフロー

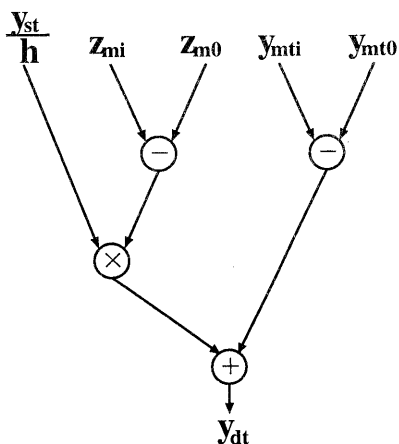


図7: データフローグラフ (特徴点探索用PE)

グラフに基づきハードウェアを構成すればよいことになる。特徴点探索用PEの具体的な構成例を図8に示す。各PEにおける処理は膨大なデータに対する繰り返し処理であり、また、入力にメモリバッファを設けているため、データは常時読み書き可能である。従って、各PEは内部を動作クロック周期に合わせて適当なセグメントに分割し、セグメント間にレジスタを配置するなどしてパイプライン処理が可能な構成としている。

3.3 並列処理による遅れ時間の短縮

提案するアルゴリズムでは類似度の計算は、各候補ごとに独立に行なえる。従って、図9に示すように、複数のVLSIプロセッサによるマルチプロセッサ構成とし、複数の候補に対して並列に類似度を計算することで、さらに遅れ時間を小さくすることが可能である。

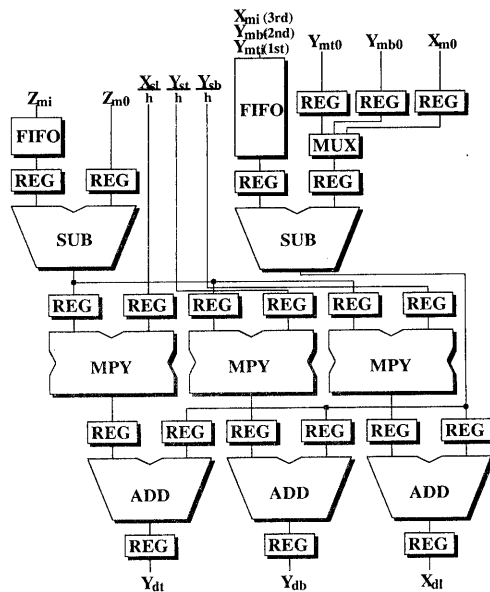


図8: 特徴点探索用PEの構成

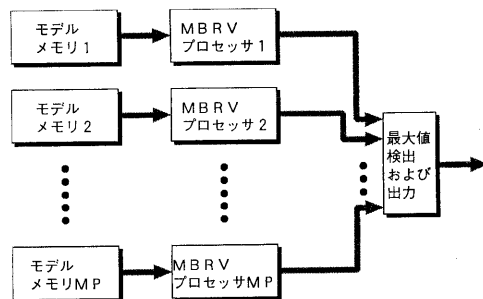


図9: マルチプロセッサによる並列処理

3.4 ハードウェアの評価

本節では、提案するVLSIプロセッサの評価を行なう。ここで用いる演算器はシミュレーションの結果より32bit固定小数点演算器とする。

3.4.1 チップ面積

特徴点探索用PEは5つの加減算器(CLA)、3つの乗算器等で構成される。この場合のチップ面積は、図10に示すレイアウト設計により、CMOS $1\mu\text{m}$ 設計ルールでおよそ $6\text{mm} \times 8\text{mm}$ となる。他のPEについてもほぼ同程度

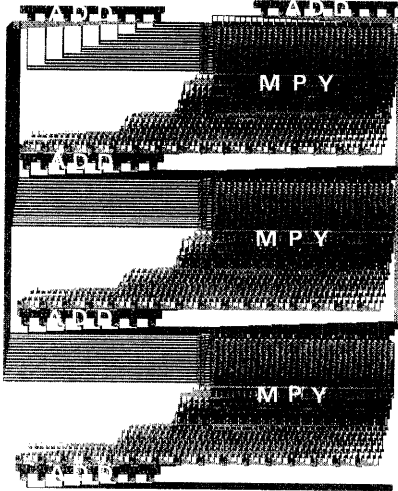


図10: レイアウト (特徴点探索用PE)

の規模となる。従って、1つのPEが1チップ化可能な程度の大きさとなる。

3.4.2 処理速度

まず、PEの処理速度について評価を行なう。例として、特徴点探索用PEの性能を評価する。PEでは、加算器の性能に基づき決定された動作クロック周期でパイプライン処理を行なう。ここでは、動作クロック周期が10nsecである場合のPEの処理速度を考える。動作クロック周期10nsecでPEを動作させるために、乗算器を4つのセグメントに分割する。この場合のパイプライン処理は、図11のタイムチャートに示すようなタイミングで行なわれる。従って、特徴点探索PEの処理時間 T_{Pr} は、1つの候補当りの平均データ数が 10^5 点であるから、

$$\begin{aligned} T_{Pr} &= (7 + 10^5) \times 10^{-8} \\ &\approx 1.0[\text{msec}] \end{aligned}$$

となる。また、動作クロック周期が T_c で動作する場合、PEの処理時間 T_{Pr} は式(5)のようになる。

$$T_{Pr} = (N_{seg} + 10^5)T_c$$

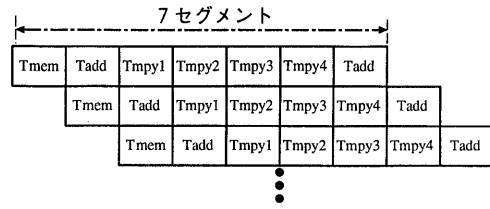


図11: タイムチャート (特徴点探索用PE: クロック周期10nsec)

ただし、 N_{seg} はPEを構成するセグメントの数である。他のPEについても同様である。表1に各動作クロック周期における各PEの処理時間を示す。

表1: PEの処理時間

PE	動作クロック周期			
	20nsec	10nsec	6.7nsec	5nsec
座標変換	2msec	1msec	0.7msec	0.5msec
特徴点探索	2msec	1msec	0.7msec	0.5msec
位置計算	$1\mu\text{sec}$	$0.5\mu\text{sec}$	$0.3\mu\text{sec}$	$0.25\mu\text{sec}$
透視投影	2msec	1msec	0.7msec	0.5msec
画像マッチング	2msec	1msec	0.7msec	0.5msec

次に、提案するVLSIプロセッサの処理速度について評価を行なう。本VLSIプロセッサでは、各処理部を1つのセグメントとしたパイプライン処理により、遅れ時間の短縮をはかっている。パイプライン処理は、図12に示すタイミングで行なわれる。この場合の動作クロック周期は、

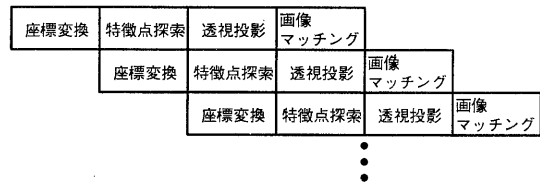


図12: タイムチャート (VLSIプロセッサ)

最も処理時間のかかるPEの性能に基づき決定される。各PEが動作クロック周期10nsecで動作する場合、表1からVLSIプロセッサの動作クロック周期はほぼ1msecと決定される。従って、例えば候補数が1800である場合のVLSIプロセッサの処理時間 T_{Pc} は、

$$\begin{aligned} T_{Pc} &= (4 + 1800) \times 10^{-3} \\ &\approx 1.8[\text{sec}] \end{aligned}$$

となる。これは、28.5MIPSの汎用ワークステーションと比較しておよそ10000倍の高速処理である。

次に、複数のVLSIプロセッサを用いた場合の処理速度について評価を行なう。複数のVLSIプロセッサで並列処理を行なう場合、図9で示したように、各VLSIプロセッサにより計算された類似度の最大値をとる構成となる。ここで、最大値の検出にかかる時間は、各VLSIプロセッサの処理時間と比較して、無視できる程度のものである。従って、MP個のプロセッサによる並列処理を行なう場合、VLSIプロセッサ単体による処理と比較してほぼMP倍の高速化が実現できる。例えば、100個のVLSIプロセッサを用いた場合、1800の候補に対する処理時間は $1.8[\text{sec}] \times \frac{1}{100} = 18[\text{msec}]$ となる。

4 むすび

提案するモデルベースロボットビジョンVLSIプロセッサは、3次元計測と3次元物体認識が同時に行なえるという従来には重要な特長を持っている。さらにロボットビジョンシステムをVLSIプロセッサにより構成することで、ソフトウェアによる処理と比較して、およそ10000倍の高速処理を可能としている。しかしながら知能ロボット実現のためには、この他にも軌道計画や制御などの処理が必要であるため、3次元計測および3次元物体認識は数msec～数sec程度の処理速度が要求される。また、今後提案するプロセッサを実環境に適用する場合、照明問題などを解決するために照明条件などもモデルとして持たせる必要があると考えられる。この場合、モデルから2次元画像への変換などにおいて処理量が増大するため、さらに高速化が要求される。また、提案するVLSIプロセッサにおいて各PEは命令レベルの並列性を最大限に活用できるように必要な最大数の演算器により構成している。そのため、各PEがそれぞれ1チップ化可能な程度の大きさとなり、VLSIプロセッサはマルチチップ構成となる。このような場合、必要最小限の処理が可能であるようなユニバーサルチップの開発などが重要な課題となる。

謝辞

常日頃から有益な御助言をいただき、東北大学工学部樋口龍雄教授に深謝する。

参考文献

- [1] 亀山：“知能ロボットとスーパーチップ”，日本ロボット学会誌, 8, 1, pp. 105-107 (1990).
- [2] 亀山, 樋口：“ロボットとVLSIコンピュータ”，日本ロボット学会誌, 6, 4, pp. 21-26 (1988).
- [3] 佐々木, 亀山, 樋口：“モデルベースロボットビジョンVLSIプロセッサの高並列アーキテクチャ”，1992 信学春季全大, C-576 (1992).

- [4] 佐々木, 亀山：“モデルベースロボットビジョンVLSIプロセッサの性能評価”，1992 信学秋季全大, D-247 (1992).
- [5] 尺長：“単眼視画像からの3次元情報復元—中心投影の逆問題とその解法—”，情報処理, 32, 2, pp. 180-188 (1991).
- [6] M. Dhome, M. Richetin, J. T. Laprestè and G. Rives: “Determination of the Attitude of 3-D Objects from a Single Perspective View”, IEEE Trans. Pattern Anal. Machine Intell., 11, 12, pp. 1265-1278 (1989).
- [7] J. L. Hennessy and N. P. Jouppi: “Computer Technology and Architecture: An Evolving Interaction”, Computer, 24, 9, pp. 18-29 (1991).
- [8] K. Hwang: “Computer Arithmetic: PRINCIPLES, ARCHITECTURE AND DESIGN”, John Wiley & Sons, Inc. (1979).