[計　測　と　制　御
Vol.40, No.12, 1/8（2001）]

知能ロボットとシステム LSI†

平　井　慎　一*

System LSI's for Intelligent Robots

Shinichi Hirai*

Applications of system LSI technology to intelligent robots will be reviewed. First, I will focus on inverse dynamics computation chips and will show why they have not been applied to the control of mechanical systems. Second, I will survey some developments in vision chips for intelligent robots. Then, I will introduce a project to develop Radon-Fourier transform chips for object handling. Finally, I will examine the future direction of system LSI technology in robotics.

Key Words: システム LSI(System LSI), ロボティクス (Robotics), 知能 (Intelligence), 感覚 (Perception), 運動 (Action)

1. はじめに

ロボティクスとは，感覚と運動の知的な結合 (Intelligent connection from perception to action) を研究する工学であるという定義がある．ハンドアイシステム，ヒューマノイド，移動ロボットは，いずれも実世界の情報をリアルタイムで取得し，多自由度の運動をリアルタイムで制御するところに共通点がある．また，バーチャルリアリティやヒューマンマシンインタフェースにおいても，実世界の情報をリアルタイムで取得し，仮想世界や実世界情報をリアルタイムで人間に提示する点で，感覚とコンピュータ内の運動との知的な結合とみなすことができる．このような感覚と運動の知的な結合を実現するときには，多くの情報をリアルタイムで処理し，多自由度の運動を制御する必要がある．ロボティクスでは，視覚や触覚等，大量の信号を処理することにより得られる情報が重要であり，さらに多自由度の運動を実現する必要がある．したがって，信号処理と運動制御の機能をシステム LSI 化することにより，リアルタイム性を実現することが期待できる．近年のシステム LSI 技術の急速な進展は，ポータブル，ウェアラブル，インプランタブルなシステムの実現を可能にしつつある [1)]．このようなシステム LSI の技術は，知能ロボットの概念を大きく変える可能性を秘めている．

本解説では，知能ロボットにおけるシステム LSI について解説するとともに，著者らが進めている物体ハンドリングのための視覚チップの概略を述べる．第 2節では，1980 年代から盛んに研究された逆動力学計算チップが，なぜ実用化されなかったかを簡単に述べる．ロボティクスにおいてビジョンは重要なセンサである．第 3節では，ビジョンチップについていくつかの研究例を挙げる．第 4節では，著者が進めているラドン・フーリエ変換法チップの開発について述べる．第 5節では，ロボティクスにおけるシステム LSI 開発について，著者の私見を述べる．

2. 運動制御チップ

ロボットは通常，複数のリンクがジョイントにより結合された，多リンク機構から成る．ロボット制御とは，各ジョイントの運動を制御することに帰着する．多リンク機構に代表されるメカニカルシステムにおいては，サンプリング周波数 1000Hz すなわちサンプリング時間 1msec 程度が安定な制御のために望まれる．

1980 年代，多リンク機構の動的制御に関する研究が進められた．多リンク機構の運動が高速になるにしたがって，動特性の影響が顕著に表れる．結果として，リンクに作用する慣性力や遠心力，コリオリ力のため，運動に誤差が生じる．動的制御とは，動特性の影響を補償して，多リンク機構の運動を制御する手法を指す．特に，多リンク機構の動特性を計算し，動特性に起因する慣性力や遠心力，コリオリ力を補償する手法として，計算トルク法が提案された．計算トルク法においてはあらかじめ，リンク機構の各ジョイントの角度，角速度，角加速度と，動特性に起因する力と重力のために各ジョイントに作用するトルクの関係式を求めておく．各ジョイントの角度，角速度，角加速度から，動特性に起因する力

†

* 立命館大学ロボティクス学科，VLSI センター　滋賀県草津市野路東 1-1-1
* Dept.Robotics and VLSI center, Ritsumeikan Univ., Shiga 525-8577, Japans
（Received September 1, 2001）
（Revised October 1, 2001）

 © 2001 SICE

と重力によって各ジョイントに生じるトルクを計算する．この計算を，逆動力学 (inverse dynamics) とよぶ．逆動力学により各ジョイントのトルクを計算し，リンクの慣性力や遠心力，コリオリ力ならびに重力をキャンセルするというのが，計算トルク法である．計算トルク法では，多リンク機構の逆動力学の計算を，サンプリング間隔以内で計算する必要がある．ニュートン・オイラー法に基づく逆動力学の定式化が行われ，シリアル機構の補償トルクを，リンク数オーダーの時間で計算する漸化式が提案された[2]．また，ラグランジュ法に基づく同様の定式化も行われている[3]．しかしながら，当時の計算機パワーでは，たとえば 6 自由度のシリアル機構の動特性を 1msec 以内に計算することは困難であった．そこで，並列計算により逆動力学を高速に計算する手法が開発され，リンク機構の逆動力学をリアルタイムで計算するチップが試作されている．たとえば，ビットスライスアーキテクチャを用いる手法[4] や行列演算を実行する専用 VLSI を用いる手法[5] が提案された．また，リンク機構のヤコビ行列を計算する VLSI も試作されている[6),7]．

残念ながら，逆動力学を計算する VLSI は，実用化されていない．一つの理由は，1990 年代に CPU の計算速度が急速に上昇して，逆動力学の計算が 1msec 以内で可能になったことである．また，ジョイントがギア等の減速器を有する場合，減速器の摩擦が運動において支配的になる．しかし，摩擦は運動中に変動するため，補償することが難しいことも理由の一つである．さらに本質的な理由は，逆動力学の計算を行うことなく，リンク系を動的に制御できることが判明したことにある．たとえば，PD 制御と重力補償により，各ジョイントの指令トルクを計算する手法で，リンク系の運動を制御することができる．このとき，慣性力や遠心力，コリオリ力の計算は必要ない．以上のように，逆動力学の計算は，多リンク機構の運動制御には必須でないことが理論的に明らかになっている[8),9]．力学系の動特性を利用することにより，リンク系を制御する手法は，ダイナミクスベースト制御とよばれている．ダイナミクスベースト制御では，逆動力学の計算が不可欠でないので，逆動力学計算チップは必要性がない．このように，運動制御に関しては，システム LSI 技術の必要性が薄れているのが現状である．

近年，ヒューマノイドやペットロボットなど，従来より運動自由度の多いロボットが注目を浴びている．ロボットメカニズムにおいては，運動を伴う部分にアクチュエータを取り付けることが一般的であるので，アクチュエータとドライバ，コントローラは可能な限り小型かつ軽量であることが必要である．これまでにも，VLSI 技術を用いて小型のドライバ，コントローラが開発されている．さらに，アクチュエータと一体化できるほど小型のドライバ，コントローラが望まれており，そこに VLSI 技術が適用されると考える．

3. ビジョンチップ

視覚信号は，二次元の時系列信号であり，その情報量は極めて大きい．したがって，画像処理には，多くの演算を要求される．たとえば，512×512 ピクセルの画像に，サイズ 3×3 の線形フィルタを適用する場合，2×10^6 回以上の乗算が必要である．画像がビデオフレームレート，すなわち 30 frame/s (fps) で得られる場合，演算速度は 70 MFLOPS を越える．また，フーリエ変換やハフ変換など，画像処理に有効であることが示されているが，演算量の大きいアルゴリズムが多い．一方，画像処理は一般的に並列性が高く，処理の高速化のためにはシステム LSI 化が有効な手段である．特に，ロボティクスにおけるビジョンにおいては，リアルタイムで画像処理を実行する必要があり，ビジョンチップの開発が進められてきた．本節では，ビジョンチップ開発の例をいくつか紹介する．

3.1 相関演算チップ

対象物を発見し追跡することは，ロボットビジョンにおける最も基本的な機能であり，ターゲットトラッキングとよばれる．ターゲットトラッキングを実現する一つの方法は，あらかじめ与えられる参照画像と類似度の高い領域を，入力画像内から探索する手法である．入力画像内で，参照ウインドウと同じサイズを有する領域を，候補ウインドウとよぶ．類似度を評価する一つの規範は，二つの画像領域間の相関関数 (Correlation) である．相関関数は 0 以上 1 以下の値を持ち，二つのウインドウ内の画像が一致しているとき相関関数の値は 1 である．ウインドウのサイズを $M \times N$ で表し，参照ウインドウ RW 内の画素値を $R_{0,0}, R_{0,1}, \cdots, R_{M-1,N-1}$，候補ウインドウ CW 内の画素値を $C_{0,0}, C_{0,1}, \cdots, C_{M-1,N-1}$ とする．このとき相関関数 Cor は，

$$Cor = \frac{E_{RC}}{\sqrt{E_{RR}E_{CC}}} \tag{1}$$

で与えられる．ここで

$$E_{RC} = \sum_{i=1}^{M-1}\sum_{j=1}^{N-1} R_{i,j}C_{i,j},$$

$$E_{RR} = \sum_{i=0}^{M-1}\sum_{j=1}^{N-1} R_{i,j}^2, \quad E_{CC} = \sum_{i=0}^{M-1}\sum_{j=1}^{N-1} C_{i,j}^2.$$

すべての候補ウインドウに対して相関関数の値を計算し，相関関数の値が最大となる候補ウインドウの位置を物体の位置とする．以上の手法を，相関法とよぶ．

上式に示す相関関数は積の個数が多く，演算時間を要する除算と平方根を含む．そこで，相関関数に近い性質を持ち，計算が単純な規範が使用されることがある．このような規範の一つは，二つの画像領域間の絶対差分和 (Sum of Absolute Differences, SAD) である．絶対差分和は，絶対誤差平均 (Mean Abosule Difference, MAD) とも言われる．SAD は 0 または正の値を取り，二つのウインドウ内の画像が一致しているとき SAD の値は 0 である．すなわち，SAD の値が小さいほど，相関度が高い．絶対差分和 SAD は，

$$SAD = \sum_{i=0}^{M-1}\sum_{j=0}^{N-1} |R_{i,j} - C_{i,j}| \tag{2}$$

で与えられる．すべての候補ウインドウに対して SAD の値を計算し，SAD の値が最小となる候補ウインドウの位置を物体の位置とする．

相関法の計算は単純であり並列性が高いので，VLSI 化に向いている．SAD の計算を行うチップを用いて実時間ビジョンシステムが構築され[10]，マイクロプロセッサのコプロセッサに適したチップが開発された[11]．また，SAD と正規化相関関数の計算アルゴリズムが VLSI 化され，それを用いたビジョンシステムがトラッキングビジョンの名称で市販されている[12]．

3.2　ステレオビジョンチップ

ステレオビジョンの原理は三角測量であり，ステレオビジョンにおいて三次元情報を得るためには，複数の画像間の対応点を求める必要がある．図 1に示すように，対応点の算出では一般に，一方の画像に参照ウインドウを設定し，それに対応する領域を他方の画像内から探索する．三次元空間内の点と二つのカメラの投影中心により定まる面をエピポーラ面，エピポーラ面と画像平面との交線をエピポーラ線とよぶ．画像間の対応を求めるときには，エピポーラ線上で対応点を探索すれば十分であることが知られている[13]．したがって，エピポーラ線上に複数の候補ウインドウを設定し，参照ウインドウと最も相関度の高い候補ウインドウを求める．このような探索を，複数の参照ウインドウに対して実行することにより，画像間の対応点を求める．以上のように，ステレオビジョンにおいて対応点を求める手続きは，計算量が多く，リアルタイムで実行することが困難である．そこで，対応点の算出を高速かつ正確に実行するために，ステレオビジョン VLSI チップが試作されている．

対応点を評価する一つの規範は，(2) 式に示す絶対差分和である．エピポーラ線上のすべての領域で SAD を計算し，SAD の最小値を求めることにより，対応点を決定する．対応点探索においては，図 2-(a) に示すように，(2) 式の MN 個の絶対差分 (AD) を並列に計算できる．また，図 2-(b) に示すように，エピポーラ線上の複数の候補ウインドウに対して，並列に SAD を評価することができる．このように，対応点探索のアルゴリズムは並列性が高いため，アルゴリズムを VLSI 化することにより，高速に対応点を求めることができる．

対応点を求めるときの問題点は，ウィンドウのサイズである．ウインドウが小さいと，相関度の高い領域が複数存在する可能性が高くなる．すなわち，対応の一意性が失われる．一方，ウインドウサイズが大きいと，相関度が全般的に低くなる．すなわち，ウインドウの中央部より遠い部分の画素が考慮されるため，対応点の信頼性が低下する．したがって，対応点を一意に決定することができ，可能な限りサイズが小さいウインドウを設定することが望ましい．そこで，ウインドウサイズを適応的に変化させるアルゴリズムが提案されている[14),15]．張山らは，ウインドウサイズを適応的に変化させるアルゴリズムを，VLSI 上に実現している[16),17]．この

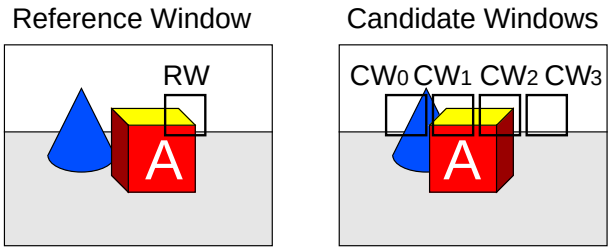
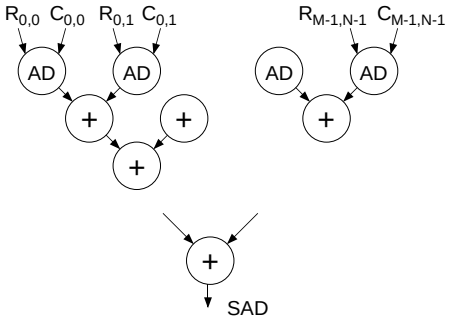
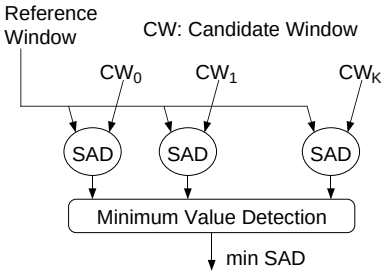


Fig. 1　Correspondance in stereo vision



(a) Pixel-parallelism



(b) Window-parallelism

Fig. 2　Parallelism in correspondance matching

ステレオビジョン VLSI においては，エピポーラ線上のすべての領域で SAD を計算する．SAD の最小値を f_1，二番目に小さい極小値を f_2 とする．ウィンドウサイズが $W \times W$ の場合，SAD の最小値のユニーク性を，$S_w = (f_1 - f_2)/W^2$ で評価する．この評価値を用いることにより，適切なウインドウサイズを自動的に決定している．このとき，図 2-(a) に示す AD の並列演算は演算部の稼働率が低くなるため採用せず，図 2-(b) に示す候補ウインドウに対する SAD の並列計算を採用している．対応点探索部には Module Based Array とよばれるゲートアレイを，入力画像に対する LOG フィルタと最小値検出部には FPGA(Field Programmable Gate Array) を用いている．

3.3　高速ビジョンチップ

ビデオ入出力で使用されている NTSC 信号のレートは 30fps である．これは，人間に動画を提示するときに十分なレートとして定められた規格である．一方，第 2節で述べたように，メカニカルシステムの安定な制御のためには，サンプリング周波数 1000Hz，すなわちサンプリング時間 1msec が必要である．したがって，ビジョン信号をメカニカルシステムのフィードバック制御に用いるためには，画像を 1000fps

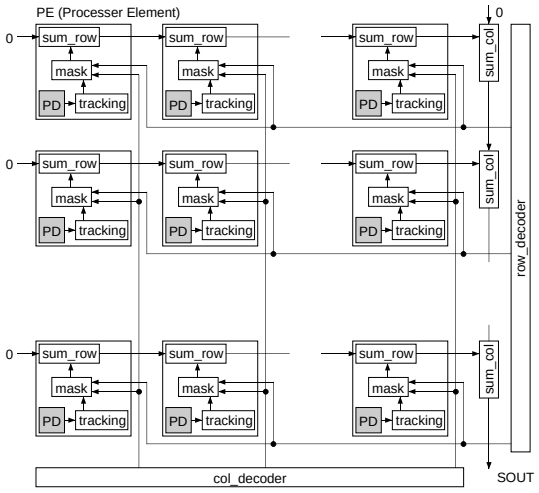


Fig. 3 Structure of target tracking chip

Fig. 4 Target tracking chip

で取得し，フィードバック信号を求めることが望まれる．結局，NTSC 信号は，人間への画像の提示が目的であり，メカニカルシステムのフィードバック制御においてはレートが不足していることがわかる．そこで，メカニカルシステムの制御を目的として，高速で画像を取得し，信号処理を実行できるビジョンシステムが研究されている．

石川らは，ビジョンチップのアーキテクチャとして S^3PE (Simple and Smart Sensory Processing Elements) アーキテクチャを提唱し，フィードバックレート 1000Hz を有するビジョンシステムを開発している [18),19)]．このアーキテクチャにしたがって，小室らは，ターゲットトラッキング用の高速ビジョンチップを開発している [20),21)]．図 3に示すように，光検出器 (PD) と信号処理部を一体化した処理要素 (PE) を並列に配置する．PE は，トラッキング回路と行総和演算回路から構成される．各行の右端には，列総和演算回路が取り付けられている．行方向，列方向には，それぞれ制御ラインが構成されている．行方向，列方向の制御ラインへの信号は，命令をもとに行方向デコーダと列方向デコーダが生成する．チップの総和計算機能を用いることにより，画像の 0 次モーメントと 1 次モーメントを計算することができる．画像サイズ 256×256 の場合，0 次モーメントを 8μs で，1 次モーメントを 66μs で計算することができる．したがって，ターゲットの重心を，約 270μs で求めることができる．0.6μm CMOS プロセスを使用して，64 × 64 の要素を集積したチップを試作したところ，図 4に示すチップが得られ，フレームレート 2.4msec でターゲットを追跡できると報告されている．

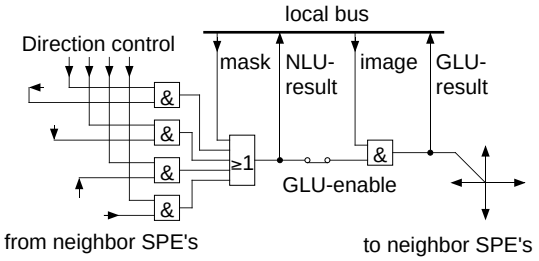


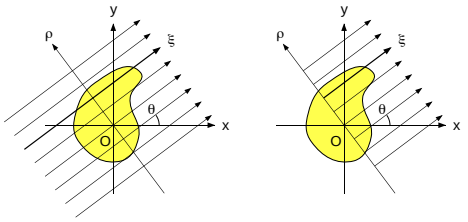
Fig. 5 GLU logic design

Eklund らは，Near-Sensor Image Processing (NSIP) の概念に基づき，ビジョンチップを開発している [22)]．このビジョンチップでは，フォトダイオードと信号処理部を一体化した要素 SPE (Sensory Processing Element) を，二次元的に配置する．すべての SPE に，図 5 に示すグローバル論理ユニット (GLU) が接続されている．各 GLU は，隣接する SPE に対応する GLU と接続されている．GLU は画像全体で並列に作用し，演算結果を隣接 4 方向に伝搬させることができる．この機能を用いることにより，領域を囲む最小長方形の計算や連結領域の抽出等，広域的な演算を実現することができる．

4. ラドン・フーリエ変換法チップの開発

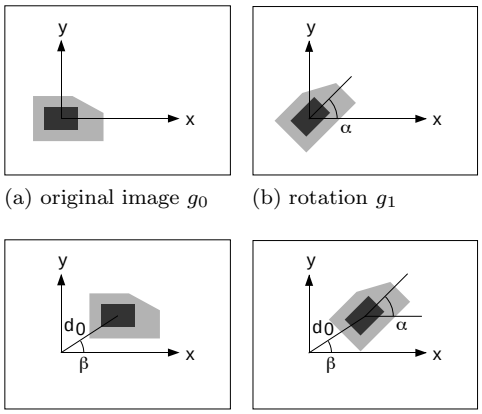
著者らは，NEDO 地域コンソーシアムのプロジェクトとして,「柔軟物ハンドリング用ビジョンチップの開発」を進めている．本節では，プロジェクトの目標と概要を述べる．

物体ハンドリングにおいては，対象物の運動を計測し，ハンド等の運動を制御する必要がある．一般的に対象物は平面運動を行うので，平面運動における位置と姿勢を検出できれば十分である．平面運動を検出する方法は，1) 正規化相関法，2) 輪郭法，3) 一般化ハフ変換法 [23)]，4) ラドン・フーリエ変換法 [24)] など，多くの手法が提案されている．正規化相関法は，物体の並進運動を検出できる反面，物体の回転運動を検出できない．輪郭法は，物体の並進運動と回転運動を検出することができる．一方，凹物体や穴を有する物体など，輪郭関数を定義できない場合や，物体の輪郭が検出できない場合には適用できない．一般化ハフ変換法とラドン・フーリエ変換法は，物体の並進運動と回転運動を検出することが可能であり，どのような物体にも適用することができる．すなわち，重心や輪郭が不明確な場合でも，物体の位置と姿勢を計測することができる．これら二つの手法の計算量は極めて大きいが，計算の並列性が高く，システム LSI 化することにより高速に実行することが期待できる．本プロジェクトでは，特にラドン・フーリエ変換法に焦点を当て，ラドン・フーリエ変換チップを試作するとともに，チップを用いた柔軟物ハ



(a) Radon　(b) onesided-Radon

Fig. 6　Integral paths in Radon transform



(a) original image g_0　(b) rotation g_1

(c) translation g_2　(d) planar motion g_3

Fig. 7　Planar motion of rigid body

ンドリングを実現することを目指す．

4.1　ラドン・フーリエ変換法

ラドン・フーリエ変換法では，入力画像と参照画像のラドン変換，フーリエ変換を比較することにより，平面運動物体の位置と姿勢を計算する [25),26)]．図 6-(a) に示すように，x 軸からの角度が θ で原点からの符号付き距離が ρ で与えられる直線に沿って，画素値 $g(x,y)$ を積分する．積分値は，角度 θ と距離 ρ に依存するので，$R[g](\rho,\theta)$ と記す．すなわち，

$$R[g](\rho,\theta)=\int_{-\infty}^{\infty} g(\xi\cos\theta-\rho\sin\theta,\xi\sin\theta+\rho\cos\theta)d\xi. \quad (3)$$

積分 $R[g](\rho,\theta)$ をラドン変換 (Radon transform) と呼ぶ．

図 7-(a) に示す原画像 g_0 を，角度 α 回転させて得られる画像を g_1 とする．また，原画像 g_0 を，角度 β 方向に距離 d_0 並進移動させて得られる画像を g_2 とする．画像 g_0, g_1, g_2 のラドン変換を，それぞれ $R_0(\rho,\theta), R_1(\rho,\theta), R_2(\rho,\theta)$ で表すと，次式が満たされる．

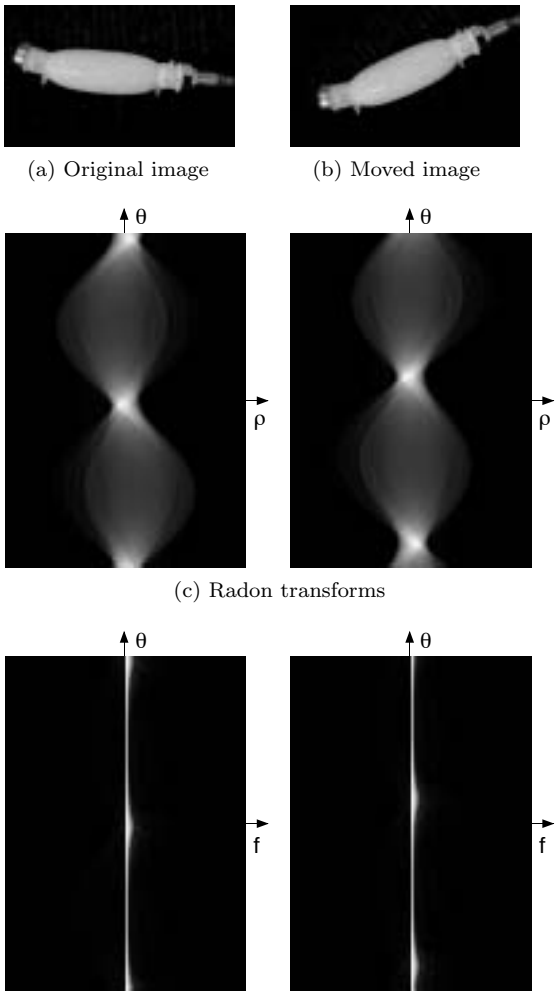
$$R_0(\rho,\theta)\equiv R_1(\rho,\theta+\alpha),\quad \forall\rho,\theta, \quad (4)$$

$$R_0(\rho,\theta)\equiv R_2(\rho-d_0\sin(\theta-\beta),\theta),\forall\rho,\theta. \quad (5)$$

原画像 g_0 を角度 α 回転させ，角度 β 方向に距離 d_0 並進移動させて得られる画像を g_3 とする．画像 g_3 のラドン変換を $R_3(\rho,\theta)$ とすると，

$$R_0(\rho,\theta)\equiv R_3(\rho-d_0\sin(\theta-\beta),\theta+\alpha),\quad \forall\rho,\theta. \quad (6)$$

すなわち，回転はパラメータ θ に関するシフトで，並進はパラメータ ρ に関するシフトで表される．パラメータ θ に関するシフト量は一定であり，回転角度 α に一致するが，パラメータ ρ に関するシフトは，角度 θ によって変化する．図 8-(a), (b) に示す二つの画像のラドン変換を，図 8-(c) に示す．図 8-(b) に示す画像は，図 8-(a) に並進変位と回転変位を与えた画像である．



(a) Original image　(b) Moved image

(c) Radon transforms

(d) Power spectrums of Radon transforms

Fig. 8　Radon transforms and their power spectrums

パラメータ ρ に関するシフトを消去するために，ラドン変換 $R(\rho,\theta)$ の ρ に関するパワースペクトルを計算する．すなわち，

$$F(f,\theta)=\left|\int_{-\infty}^{\infty} R(\rho,\theta)e^{i\rho f}d\rho\right|^2. \quad (7)$$

ラドン変換 $R_0(\rho,\theta), R_3(\rho,\theta)$ のパワースペクトルを，それぞれ $F_0(f,\theta), F_3(f,\theta)$ とする．(6) 式より，

$$F_0(f,\theta)\equiv F_3(f,\theta+\alpha),\quad \forall f,\theta. \quad (8)$$

したがって，パワースペクトル $F_0(f,\theta)$ と $F_3(f,\theta)$ の相関関数を計算することにより，回転角度 α を算出することができる．すなわち，ラドン変換のパワースペクトルは，回転角にのみ依存するので，それらの一次元マッチングを計算することにより，回転角のみを検出することができる．図 8-(a),

(b) に示す二つの画像のラドン変換のパワースペクトルを，図 8-(d) に示す．図 8-(d) に示す二つの画像の縦方向のシフトが，回転角に一致する．

回転角度 α が求められると，回転運動を除去することができる．そこで，原画像と回転運動を除去した並進画像を比較し，並進移動量を算出する．原画像のラドン変換を $R_0(\rho,\theta)$，並進画像のラドン変換を $R_2(\rho,\theta)$ とする．パラメータ ρ に沿ってラドン変換 $R_2(\rho,\theta)$ と $R_0(\rho,\theta)$ を比較する．そのために相関関数

$$\sigma[R_2,R_0](d)=\int_{-\infty}^{\infty}\{R_2(\rho+d,\theta)-R_0(\rho,\theta)\}^2\,d\rho \quad (9)$$

の最小値を求める．角度 θ に対して，相関関数が最小となる d の値を求める．これを $d_{min}(\theta)$ で表す．原画像のラドン変換と並進画像のラドン変換の関係は (5) 式で表されるので，

$$d_{min}(\theta)=-d_0\sin(\theta-\beta) \quad (10)$$

が成り立つ．したがって，関数 $d_{min}(\theta)$ の振幅と位相を求めることにより，並進移動距離 d_0 と並進移動方向 β を計算することができる．

以上のようにラドン・フーリエ変換法では，重心や輪郭を使わず，画像全体の変換を比較するため，物体が重なっている場合や画像が欠けている場合に対しても，物体の回転運動と並進運動をロバストに計測することができる．

ラドン・フーリエ変換法には，角度 α と $\alpha+\pi$ が区別できないという欠点がある．これは，パワースペクトルを正負に分離する手法や，図 6-(b) に示す積分路を有する片側ラドン変換により解決することができる．

4.2 FPGA と C/C++ベースのハードウェア記述

本プロジェクトでは，実際のハンドリングで使用されるビジョンチップを試作することを目的としている．そのためには，ハンドリングに必要な機能をローコストで提供することが望まれる．そこで，可能な限り汎用の技術と部品を用いる．

画像の入力には，CCD カメラと Bt シリーズに代表されるビデオキャプチャチップを用いる．このとき，CCD カメラからの NTSC 信号は，順次 A/D 変換され，後段の画像処理部に送られる．画像処理部においては，ディジタル画像を扱うので，論理回路が構成できれば十分である．そこで，FPGA 上に，画像処理用のハードウェアを実現する．ただし，ラドン・フーリエ変換法では，ラドン変換やフーリエ変換の結果を保存しておくメモリが必要である．そこで，SRAM 内に変換の結果を保存し，必要に応じて参照する．

ハードウェア記述では，Verilog HDL や VHDL に代表されるハードウェア記述言語を使うのが，現在の主流である．一方，ロボティクスにおけるアルゴリズムの開発には，C/C++が広く用いられている．本プロジェクトにおけるラドン・フーリエ変換法のアルゴリズムも，C/C++で記述されている．アルゴリズムを VLSI に実装するためには，C/C++で記述したアルゴリズムをハードウェア記述言語で書き換える必要がある．しかしながら，アルゴリズムの規模が大きく，複雑になるにしたがって，アルゴリズムの開発とハードウェアの開発を繰り返し行う必要性が高くなる．特に，知能ロボットにおける信号処理アルゴリズムや運動制御アルゴリズムは，開発と検証を繰り返し行うことが多い．この繰り返しに要する時間を短縮するため，C/C++をベースとするハードウェア記述が提案されている．C/C++ベースのハードウェア記述言語として，SystemC，SpecC，CycleC，Handel-C 等がある．

本プロジェクトでは，CycleC (C Level Design, Inc.) と Handel-C (Celoxica Ltd.) を用いて，ラドン・フーリエ変換のロジックを記述する．CycleC は，ANSI C/C++の文法のサブセットである．サブセットである CycleC の記述スタイルに従って，ハードウェアを記述する．論理回路の挙動は C++のクラスとメソッドで記述する．実際の論理回路は，インスタンスに対応する．CycleC により記述されたロジックは，通常のコンパイルを行うことにより，ソフトウェアとして実行することができる．一方，CycleC により記述されたロジックは，トランスレータにより HDL へ変換することができる．すなわち，通常のコンパイルと実行によるアルゴリズムの検証と，HDL への変換が両立しており，アルゴリズムの開発とハードウェアの記述を効率よく進めることが期待できる．一方，Handel-C は，Oxford 大学における並列計算に関する研究の成果をもとに開発された言語である．Oxford 大学では 1980 年代，並列計算を記述するプログラミング言語として，CSP 理論に基づき Occum を開発した．Handel-C には，ANSI C/C++に並列計算を記述する Occum の構文，すなわち，並行プロセスを記述する par，並行プロセス間のチャネルを記述する!や?等が追加されており，ハードウェアの並列性を明確に記述することができる．Handel-C により記述されたロジックから，直接ネットリストが生成される．

以上のように，FPGA と C/C++ベースのハードウェア記述を用いて，ラドン・フーリエ変換法を実行するチップを実現する．

5. 将来の方向

ロボティクスにおいては，並列性が高く，システム LSI 化することにより高速化が期待できる分野が多く，システム LSI 技術が浸透する可能性が高い．第 3節で述べたように，視覚信号は二次元の時系列信号であり，その情報量は極めて大きい．さらに，ロボティクスにおいてはリアルタイムの画像処理が要求される．画像処理は一般に並列性が高く，システム LSI 化による高速化が期待できる．すでに，フィルタリングやエッジ検出等の初期処理，相関演算に関しては，実用化されている．現在の研究開発が進むにしたがって，

- ステレオビジョンにおける対応点探索
- ハフ変換，ラドン変換，フーリエ変換等の画像変換
- マルチベースラインステレオビジョン [27]

等において，VLSI 技術が適用され実用化されると考えられる．並列画像処理に関する研究は，従来より数多く成されて

おり[28),29)]，システム LSI への実装が期待できる．また，高速な視覚センサである CMOS イメージセンサは，すでに人工網膜 LSI の名称で市販されており[30)]，今後さらなる大規模化が期待できる．このような高速視覚センサを用いる場合には，画像処理を高速化する必要があり，システム LSI 技術が期待される．

ビジョンと同様の二次元時系列信号処理が今後必要になる分野として，オーディオ信号処理とハプティック信号処理が挙げられる．たとえば，単一のマイクロホンではなく，マイクロホンアレイにより音声信号を二次元時系列として取得し，それを処理することにより音源定位や話者認識が可能になると考えられる．また，ロボットの表面に分布的に配置される触覚センサ，圧力センサ，近接センサからの信号をもとに，ロボットの運動を制御することが試みられている[31)]．このようなオーディオ信号処理やハプティック信号処理をリアルタイムで実行するためには，システム LSI 化が不可欠である．今後，マイクロフォンアレイやハプティックセンサ等の集積センサが実現するにしたがって，集積センサから得られる大量の時系列信号を，リアルタイムで処理しフィードバックする必要性が高まる．このような二次元時系列の処理は高い並列性を有することが多く，信号処理アルゴリズムのシステム LSI 化は，リアルタイム信号処理への有効なアプローチである．

VLSI 技術が有望なもう一つの分野として，シミュレーションがある．たとえば，GRAPE プロジェクトでは，天文学における大規模多体問題を専用に解くコンピュータを開発している[32)]．天文学における多体系とは，ニュートンの万有引力が支配する系であり，二つの天体間の引力は，単純に天体の質量と天体間の距離で決まる．ただし，銀河系内の恒星のように天体の数が膨大な系を対象とするため，シストリックアレイとよばれる並列演算により計算速度を高めることが試みられている．ロボティクスにおけるシミュレーションも，個々の処理は単純であるが，処理全体の個数が多いため，計算時間を要する場合が多い．このような場合は，計算アルゴリズムをシステム LSI 化することにより，高速にあるいはリアルタイムでシミュレーションを実行することが可能になる．たとえば，物体の変形シミュレーションにおいては，複数の質点を力学要素で結合した格子モデルが使用される．二つの質点間に作用する力は一般に，質点の位置と速度で決まる．ただし，計算精度を上げるためには質点の数を増やすことが必要である．また，バーチャルリアリティ応用ではリアルタイム計算が必要である．したがって，変形を計算するアルゴリズムをシステム LSI 化し，高速のシミュレーションを実現することが望まれる．このように，計算は単純であるが処理個数が多いシミュレーションにおいて，システム LSI 化はリアルタイムシミュレーションに有効である．

ロボティクスにシステム LSI を導入するためには，システム LSI の設計手法を発展させる必要がある．従来より，並列計算のプログラミングとデバッグの困難さは指摘されており，アルゴリズムのシステム LSI 化にも同様の課題が生じる．ロボティクスにおけるシステム LSI においては，設計と検証を繰り返すことが多く，設計と動作検証が容易な手法が望まれる．たとえば，データフロー計算機上での画像処理アルゴリズムから，画像処理用 ASIC を直接導く手法が提案されている[33)]．また，演算遅れ時間を小さくするために，動的に再構成可能な VLSI が提案されている[34)]．このような VLSI においても，並列アルゴリズムを設計し実装する手法が必要である．並列アルゴリズムの開発とシステム LSI 化を，統一的に進めることができる手法を期待したい．

6. おわりに

30 年前，コンピュータは特定の場所にあり，それを使うのはプログラマを始めとする限られた技術者であった．マイクロプロセッサの登場は，このような状況を劇的に変えた．すなわち，マイクロプロセッサが様々な機器に組み込まれ，メカトロニクスという分野が誕生した．プログラミングは工学における基本技術になり，工学の様々な分野でコンピュータが日常的に使われている．筆者は，システム LSI 技術において，同様の現象が生じる可能性が高いと考えている．従来，VLSI の開発は，限られた場所でスペシャリストが進める仕事であった．FPGA や CPLD (Complex Programmable Logic Device) に代表される再構成可能な VLSI と論理設計ツールの一般化は，VLSI スペシャリストでない技術者がシステム LSI を開発することを可能にしつつある．これに伴い，システム LSI がこれまで以上に様々な機器に組み込まれ，工学の広い分野でシステム LSI が使われることになると予想している．システム LSI 技術は，メカトロニクスを，知能集積機械というべきシステムに変貌させるであろう．

ロボティクスは，1) 膨大なセンサ情報を扱う，2) リアルタイム性が要求されるという特性を有しており，システム LSI が活躍する格好の舞台を提供する．反面，1) 逆動力学計算のように，研究の進展に伴って必要性が変化する可能性がある，2) 情報端末やネットワークデバイスのような，大量のニーズは期待できないという背景がある．したがって，ロボティクスにおける VLSI を発展させるためには，FPGA に代表される再構成可能なハードウェア，並列アルゴリズムの開発環境，システム LSI の設計・検証ツールのさらなる発展が望まれる．

参 考 文 献

1) 黒田, システム LSI の可能性と課題, 電子通信情報学会誌, Vol.84, No.8, pp.552–558, 2001

2) Luh, J. Y. S, Walker, M. W., and Paul, R. P. C., *On-line Computational Scheme for Mechanical Manipulators*, ASME J. of Dynamic Systems, Measurement, and Control, Vol. 102, pp.69–76, 1980

3) Hollerbach, J. M., *A Recursive Lagrangian Formulation of Manipulator Dynamics and a Comparative Study of Dynamics Formulation Complexity*, IEEE Trans. on Systems, Man, and Cybernetics, Vol.10, No.11, pp,730–736, 1980

4) Rahman, M. and Meyer, D. G., *A Cost-effective High Performance Bit-serial Architecture for Robot Inverse Dy-*

namics Computation, IEEE Trans. on Systems, Man, and Cybernetics, Vol.17, No.6, pp,1050–1058, 1987
5) Kittichaikoonkit, S., Kameyama, M, and Higuchi, T., *Design of a Matrix Multiply-addition VLSI processor for Robot Inverse Dynamics Computation*, Trans. IEICE, Vol.74, No.11, pp.3819–3828, 1991
6) Yeung, T. B., and Lee, C. S. G, Efficient Parallel Algorithms and VLSI Architectures for Manipulator Jacobian Computation, IEEE Trans. on Systems, Man, and Cybernetics, Vol.19, No.5, pp.1154–1166, 1989
7) 藤岡, 亀山, 樋口, 冗長マニピュレータ制御用座標変換 VLSI プロセッサ, 電子情報通信学会論文誌, Vol. J75-D-1, No.10, pp.909–916, 1992
8) Arimoto, S., *Control Theory of Nonlinear Mechanical Systems – A Passivity-based and Circuit-theoretic Approach*, Oxford Science Publications, 1996
9) 大須賀, メカニカルシステムに対する制御系設計理論, 計測と制御, Vol.40, No.6, pp.403–410
10) 井上, 稲葉, 森, 立川, 局所相関演算に基づく実時間ビジョンシステムの開発, 日本ロボット学会誌, Vol.13, No.1, pp.134–140, 1995
11) Bugeja, A. and Yang, W., *A Reconfigurable VLSI Coprocessing System for the Block Matching Algorithm*, IEEE Trans on VLSI Systems, Vol.5, No.3, pp.329–337, 1995
12) 岡林, 沢崎, 内山, 佐藤, 組込型トラッキングビジョンの開発, 第5回ロボティクスシンポジア, pp.369–374, 2000
13) 徐, コンピュータビジョンにおけるエピポーラ幾何, 松山, 久野, 井宮 編, コンピュータビジョン:技術評論と将来展望, 新技術コミュニケーションズ, pp.80–96, 1998
14) Kanade, T. and Okutomi, M., *A Stereo Matching Algorithm with an Adaptive Window: Theory and Experiment*, IEEE Trans on Pattern Analysis and Machine Intelligence, Vol.16, No.9, pp.920–932, 1989
15) Okutomi, M. and Kanade, T., *A Locally Adaptive Window for Signal Matching*, Int. J. of Computer Vision, Vol.7, No.2, pp.143–162, 1990
16) Hariyama, M., Takeuchi, T., and Kameyama, M., *VLSI Processor for Reliable Stereo Matching Based on Adaptive Window-Size Selection*, Proc. 2001 IEEE Int. Conf. on Robotics and Automation, pp.1168–1173, Seoul, May, 2001
17) 張山, 竹内, 亀山, 高性能ステレオビジョン VLSI プロセッサの試作, 日本機械学会ロボティクスメカトロニクス講演会 CD-ROM, 2001
18) 石川, 超並列・超高速ワンチップビジョンとその応用, 日本ロボット学会誌, Vol.13, No.3, pp.335–338, 1995
19) 小室, 鈴木, 石井, 石川, 汎用プロセッシングエレメントを用いた超並列・超高速ビジョンチップの設計, 電子情報通信学会論文誌, Vol.J81-D-I, No.2, pp.70–76, 1998
20) 小室, 石井, 石川, 吉田, 高速対象追跡用ビジョンチップの設計と試作, 日本機械学会ロボティクスメカトロニクス講演会 CD-ROM, 2001
21) 小室, 石井, 石川, 吉田, 高速対象追跡ビジョンチップ, 電子情報通信学会論文誌, Vol.J84-D-II, No.1, pp.75–82, 2001
22) Eklund, J.-E., Svensson, C., and Aström, A., *VSLI Implementation of a Focal Plane Image Processor – A Realization of the Near-Sensor Image Processing Concept*, IEEE Trans. on VLSI Systems, No.4, Vol.3, pp,322–335, 1996
23) Ballard, D., H., *Generalizing the Hough Transform to Detect Arbitrary Shapes*, Pattern Recognition, Vol. 13, No. 2, pp.111–122, 1981
24) Onishi, H. and Suzuki, H., *Detection of Rotation and Parallel Translation Using Hough and Fourier Transforms*, Proc. 1996 Int. Conf. on Image Processing, Vol.3, pp.827–830, 1996
25) 大西, 鈴木, 有本：ハフおよびフーリエ変換を用いた拡大・回転・平行移動検出法の部品位置決めへの応用, 日本ロボット学会誌, Vol. 16, No. 2, pp.232–240, 1998
26) Tsuboi, T., Masubuchi, A., Hirai, S., Yamamoto, S., Ohnishi, K., and Arimoto, S., *Video-frame Rate Detection of Position and Orientation of Planar Motion Objects using One-sided Radon Transform*, Proc. IEEE Int. Conf. on Robotics and Automation, Vol. 2, pp.1233–1238, Seoul, May, 2001
27) 金出, 蚊野, 木村, 川村, 吉田, 織田, ビデオレートステレオマシンの開発, 日本ロボット学会誌, Vol.15, No.2, pp.261–267, 1997
28) Levialdi, S. eds., *Multicomputer Vision*, Academic Press, 1988
29) 美濃, 並列画像処理, コロナ社, 1999
30) 久間, 田中, 太田, 田井, 岩附, 人工網膜チップの開発と事業化, 応用物理, Vol.67, No.4, pp.424–430, 1998
31) Lumelsky, V. J., Shur, M. S., and Wagner, S., *Sensitive Skin*, IEEE Sensors Journal, Vol.1, No.1, pp.41–51, 2001
32) Makino, J. and Taiji, M., *Scientific Simulations with Special-Purpose Computers – The GRAPE Systems*, John Wiley and Sons, 1998
33) Kraljić, I. C., Quénot, G. M, and Zavidovique, B., *From Real-Time Emulation to ASIC Integration for Image Processing Applications*, IEEE Trans on VLSI Systems, Vol.4, No.3, pp.391–404
34) 藤岡, 亀山, ビットシリアルアーキテクチャに基づくロボット制御用再構成可能 VLSI プロセッサの構成, 電子情報通信学会論文誌, Vol.J81-D-I, No.2, pp.85–93, 1998

［著 者 紹 介］

平 井 慎 一（正会員）

1963 年 3 月 19 日生．90 年 京都大学大学院工学研究科 博士課程数理工学専攻単位取得退学．同年 大阪大学工学部電子制御機械工学科助手．95 年同助教授．96 年立命館大学理工学部ロボティクス学科助教授となり，現在に至る．89 年 米国マサチューセッツ工科大学 客員研究員．90 年度計測自動制御学会論文賞．96 年度日本機械学会ロボティクスメカトロニクス部門業績賞．柔軟物ハンドリング，リアルタイムビジョン，分散マニピュレーションなどの研究に従事．日本ロボット学会，電子通信情報学会，日本機械学会，IEEE などの会員．（工学博士）．