

高速画像処理用並列計算機RIPEの 計算精度と処理時間に関する評価

金川 英一[†] 中西 弘泰[†] 本沢 邦朗[†]
塩見 彰睦[†] 今井 正治[†] 長谷川 純一^{††}

[†] 豊橋技術科学大学情報工学系, 豊橋市
^{††} 中京大学情報科学部, 豊田市

複雑で計算量の多いフィルタをRIPE上で実現することにより、RIPEシステムの性能評価および問題点の解析を行った結果について述べる。RIPEは、1次元プロセッサ・アレイ上で、画像データに対する空間並列性を持った局所処理を実行するSIMD型並列計算機である。今回、計算量の多いフィルタを実現することで計算精度と処理時間に関して評価を行なった。評価の結果、RIPE(512PE, 基本構成, 10GOPS)は汎用EWS(15.8MIPS)と比べて処理時間でおおよそ100倍から1000倍高速であることが推定される。このとき、汎用EWSと比べて処理結果の画素値に本質的な違いは見られないことを確認した。またこの評価をもとに問題点を解析した結果、RIPEに改良を加えることで、おおよそ700倍から4000倍の高速化が図れると推定された。

Computational Precision and Execution Times of Filtering Operations on RIPE

Eiichi KANAGAWA[†], Hiroyasu NAKANISHI[†] Kunio HONSAWA[†]
Akichika SHIOMI[†], Masaharu IMAI[†], Jun-ichi HASEGAWA^{††}

[†] Department of Information and Computer Sciences,
Toyohashi University of Technology, Toyohashi, 441 Japan
^{††} School of Computer and Cognitive Sciences,
Chukyo University, Toyota, 470-03 Japan

RIPE is a SIMD type parallel processor with one-dimensional array structure for very high speed image processing. In this paper, we describe the computational precision and execution times of filtering operations on RIPE. The filtering operations which contain large amount of computation were executed on RIPE system and typical EWS. According to the experimental results, it is estimated that the execution times of RIPE system are one hundred to one thousand times as short as those of typical EWS. We have analyzed the experimental results and found that the execution times of RIPE can be reduced seven hundred to four thousand times as short as those of typical EWS by enhancing the hardware of RIPE.

1 はじめに

従来の画像処理では、処理時間の制約などにより、計算量の少ない処理によって目標を達成する方法が研究されてきた。従って、主に3×3画素程度の小さなサイズの局所領域における処理が用いられてきた。しかしながら、近年では計算機の性能向上にともない、大規模な2次元濃淡画像に対して、比較的大きなサイズの局所領域での処理や、計算精度を要する処理、複雑で計算量の多い処理などが行なわれるようになってきた。従来の小さなサイズの局所領域での積和演算に重点をおいた画像処理システムでは、このような処理を高速に実行することが困難である。そこで著者らは、(1)大規模な2次元濃淡画像に対する、(2)比較的大きなサイズの局所領域での、(3)複雑で計算量の多い処理の、高速実行が可能な並列画像処理システムの実現を目的として、高速画像処理用並列計算機 RIPE (Real-time Image Processing Engine) の研究開発を行ってきた^{1), 2)}。これまでに、プロセッサ・アレイを実現する専用VLSI (RIPEチップ) をASIC技術を用いて実現し³⁾、これを用いてシステムの試作を行なった⁴⁾。現在の試作システムは、原理検証および性能評価を主目的としている。そのため、実現の容易性を重視して、目的を達成できる範囲内で機能を制限した仕様となっている。具体的には、1つのRIPEチップに内蔵できるプロセッサ数を多くするために、局所メモリの容量を制限し、乗算器の内蔵を見合わせた。

これまでのRIPEシステムの評価では、主に平滑化処理や強調処理等に用いられる比較的単純な処理を行うフィルタを用いてきた。中にはエッジ保存平滑化やメディアンフィルタのように、計算量の多い処理を行なうフィルタもあるものの、RIPEの機能を制限したことによる大きな問題点はなかった。そこで、複雑で計算量の多いフィルタをRIPE上で実現し、RIPEシステムの性能評価および問題点の解析を行なった。本稿ではその結果について示す。

複雑で計算量の多いフィルタとしては、集中度フィルタ⁵⁾を実現した。このフィルタは、胃がんを自動検出する目的で開発され、画像の各点に線がどの程度集中しているかを定量的に示す。また、計算量の多いフィルタとしては、尺度空間フィルタ⁶⁾を実現した。このフィルタは構造解析等に広く応用されており、比較的大きなサイズの局所領域における積和演算によって実現される。今回これらのフィルタを実現するにあたって、処理時間ももちろんのこと、計算精度も重視した。

以下本稿では、まず2.でRIPEシステムの概要を述べ、3.で集中度フィルタによる評価について、4.で尺度空間フィルタによる評価について述べる。

なお、本稿では特にことわらない限り、画像サイズをN×N画素、局所領域のサイズをM×M画素とする。また、1行を横1行、1列を縦1列の意で用いる。

2 RIPEシステムの概要

RIPEシステムの概要について述べる。まずRIPEのアーキテクチャについて述べ、次に基本構成のRIPEシステムを示し、最後にRIPE上でのアルゴリズムについて述べる。

2.1 RIPEのアーキテクチャ

RIPE^{1), 2)}は、1次元プロセッサ・アレイ上で2次元濃淡画像に対する空間並列性を持った局所処理を実行する方式を採用したSIMD型並列計算機である。処理は、入力、演算、出力の3つのステージにパイプライン化されて、行単位で行われる。プロセッサ・アレイは、複数個のRIPEチップから構成される。RIPEチップは8個のPE (Processing Element) を内蔵している専用VLSIである。各PEは画像の縦1列の処理を受け持ち、演算に必要な局所領域内の画素データの縦1列分を保持するための局所メモリが内蔵されている。また、16bit幅のデータ転送用の双方向入出力ポートを2ポート持ち、隣接するPE間でのデータ転送が可能である。従って、演算に必要な局所領域内の画素データを得るためには、縦方向は局所メモリを参照し、横方向のみPE間データ転送を行えばよい。

2.2 基本構成RIPEシステム

基本構成RIPEシステムでは、次のようにフレーム・メモリとプロセッサ・アレイを構成する。

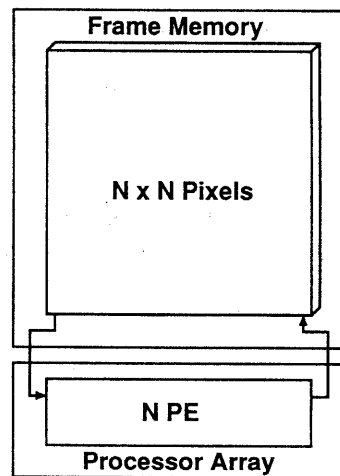


図1: 基本構成RIPEシステム

- (1) フレーム・メモリは処理前の画像と処理後の画像を格納するために、複数のプレーンを持つ。各プレーンには1画面分の画像データを格納できる。
- (2) N個のPEでプロセッサ・アレイを構成する。
- (3) フレーム・メモリとプロセッサ・アレイを図1のように接続する。

(4) 処理を行う場合は、処理前の画像データを格納したプレーンから画素データがラスタスキャン順に読み出され、プロセッサ・アレイで処理されたデータが処理結果を格納するプレーンにラスタスキャン順に書き込まれる。

2.3 RIPE 上での局所処理のアルゴリズム

2次元の局所領域での処理を1次元の処理に分解することにより、計算量を減らせる場合がある。これをRIPEに適用した場合の処理の流れとその効果を示す。ここではまず2次元の局所領域での処理の1次元分解について述べ、次にRIPEで実現した場合について述べる。

2.3.1 1次元分解の概要

2次元の局所領域での処理を1次元の処理に分解することは、2次元フィルタリングの1次元分解として横井らによって報告されている⁷⁾。図2のように2次元の局所領域での処理が、1次元の局所領域での処理に分解して実現できる場合、K段分解可能という。局所領域のサイズを $M \times M$ 画素とし、その局所領域における処理を $P[M, M]$ とする。また、横1画素 \times 縦 M 画素の1次元の局所領域における処理を $P_i[1, M]$ 、横 M 画素 \times 縦1画素の1次元の局所領域における処理を $P_i[M, 1]$ で表す。“ A_i ”は点処理を、“*”は画像間処理を示す。図2は、原画像データに対して順番に $P_i[1, M]$ 、 $P_i[M, 1]$ 、 A_i を施し、 K 個の結果を合成することにより、2次元の局所領域での処理を実現することを示している。

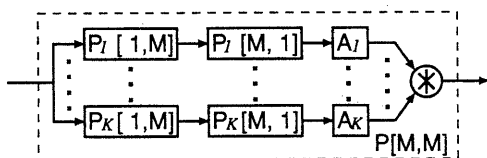


図2: 2次元の局所領域での処理の1次元分解

1画素当たりの演算回数は、定義式通りに2次元の局所領域での処理を行う場合、およそ M^2 に比例する。これに対して、2次元の局所領域での処理を1次元の局所領域での処理に分解した場合の演算回数は、およそ $K(M+M)$ に比例する。従って、2次元の局所領域での処理が1次元の局所領域での処理に分解できる場合、演算回数が減少できる可能性がある。

2次元の局所領域における加重マトリクスを用いた積和演算の1次元分解を考える。加重マトリクスを行列 W と見なすと、2次元の局所領域での処理はrank W 段に1次元分解できる。例えば、加重がすべて一樣(rank $W = 1$)ならば1段に、左右対称(rank $W \leq [M/2]$)ならば、 $[M/2]$ 段以下に分解可能である。表1に、2次元の局所領域での処理を2次元のまま実行した場合の演算回数と、1次元に分解して実行した場合の

演算回数を示す。表1より、2次元の局所領域での処理を1次元分解することにより演算回数を減らせる場合があることが知られる。

表1: 演算回数の比較

rank W	2次元		1次元分解	
	乗算	加算	乗算	加算
M	M^2	M^2-1	M^2	M^2-1
$[M/2]$	M^2	M^2-1	$M(M+1)/2$	$(M-1)(M+3)/2$
1	1	M^2-1	1	$2(M-1)$

加重マトリクス W , W のサイズ $M \times M$

2.3.2 RIPE への1次元分解の適用

RIPEは、2次元の局所領域での処理を1次元の局所領域での処理に分解するのに適した構造をもっている。RIPE上で1次元に分解して処理する場合は次のような流れとなる。まず、 $P_i[1, M]$ では局所メモリ内の画素データのみを用いて演算が行われる。次に、 $P_i[M, 1]$ では演算に必要なデータを得るためにPE間データ転送を行い、得たデータに対して演算を施す。その結果に A_i を施した後、“*”で自PE内の K 個の結果を合成する。このように、RIPEはまず列方向、次に行方向の1次元の処理を実行することになる。この場合、2.3.1で述べたように1画素当たりの演算回数を減らすことができる。同時にRIPEでは、1次元に分解して処理を行うことにより、PE間での転送回数も削減できる。RIPEで2次元の局所領域における積和演算を2次元のまま処理を行った場合と、1次元分解して実現した場合のPE間転送回数を表2に示す。表2より、RIPE

表2: RIPEのPE間転送回数の比較

rank W	2次元	1次元分解
M	$M(M-1)$	$(M^2-1)/4$
$[M/2]$	$M(M-1)$	$(M^2-1)/4$
1	$M(M-1)$	$M-1$

加重マトリクス W , W のサイズ $M \times M$

へ1次元分解を適用するとPE間転送回数が減少することが知られる。このようにRIPEへ1次元分解を適用すると、PE間転送回数および演算回数が減少する場合があり、処理の高速化を図ることができる。

3 集中度フィルタによる評価

複雑で計算量の多いフィルタの1つである集中度フィルタを実現し、RIPEシステムの評価および問題点の解析を行う。以下では、まず集中度フィルタの概要について述べる。次に、今回計算機上での実現に用いた方法と、その計算精度および処理時間について述べる。

そして最後に、RIPE 上でより高速に処理するための改良点について考察を行う。

3.1 集中度フィルタの概要

集中度 (convergence index)⁵⁾とは、画像の各点でその点へ向かって線がどの程度集中しているかを定量的に示した尺度である。集中度は、胃がん、特に早期がんに特有な胃壁に現れるひだの集中パターンに注目し、これを X 線像から自動検出する目的で開発された。胃がんの多くが胃壁のひだの集中を伴うという事実に着目し、そのような集中の度合を集中度という尺度で定量化することによって病変部と疑わしい場所を検出する。

線群がある点に集中している程度を表すには、各線におけるその点へ向かう成分の大きさの総和を知らねばよい。そこで、1本の線が局所的に定義できる多数の線素 (line primitive) の連なりで構成されているものと考え、各線素が被評価点への程度集中しているかを調べ、各線におけるその点へ向かう成分の大きさの総和を求める。ここで線素とは、8 連結線図形上の 1 点 (端点、分岐点、交差点は除く) に対する、その前後 2 点を結んでできる線分と同じ方向で、かつその半分の長さをもつベクトルと定義されている。この定義より線素は 8 種類の方向と 3 種類の長さを取り得ることになり、文献 5) の線素決定手順より得られる線素画像を入力として用いることにより集中度を求める。

ある点 P に対する集中度 $C(P)$ の説明図を図 3 に示し、定義式を示す。集中度は、0 以上 1 以下の値をとる。

$$C(P) = \frac{\sum_R \frac{dx |\cos \alpha|}{r}}{\sum_R \frac{dx}{r}} \quad (1)$$

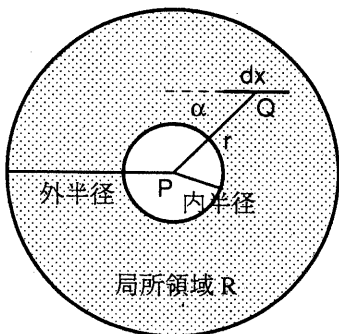


図 3: 集中度の概要

3.2 集中度フィルタの実現

集中度を求めるには、線形画像の局所領域内の各点に対して平方根や三角関数を求めたり、乗除算や加減算などの様々な計算が必要である。しかし、このような計算のすべてを原画像のそれぞれの点で行うのは効

率的ではない。そこで、あらかじめ計算可能な計算を行い、局所領域内の各画素で取り得る画素値の中間結果を求めておく。実行時には、中間結果を原画像のそれぞれの点で画素値により選択し、選択した値に対してさらに演算を施して集中度を求める。

3.2.1 RIPE 上での実現

集中度フィルタに関しても、図 2 に示したように 1 次元分解することにより高速化を図ることができる。集中度フィルタは M 段に分解できる。この場合の 1 画素当たりの演算回数は分解する前と変わらない。これに対して、RIPE 上での 1 画素当たりの PE 間転送は、2 次元の局所領域で処理を行った場合は $M(M-1)$ 回必要であるが、1 次元分解した場合は $(M^2-1)/4$ 回となる。このように、RIPE 上での集中度フィルタは 1 次元分解することによって処理が高速化される。

実数で表現される集中度を RIPE 上では整数に置き換えて出力しなければならない。集中度は定義式より 0~1 の間の実数値を取り得るが、現在の RIPE システムでは処理結果を 256 階調の画像で出力している。そこで、RIPE では集中度 0~1 の間の実数を 256 分割して整数で表現している。RIPE 上での集中度フィルタの処理画像における濃度差 1 は、式 (1) で与えられる集中度では約 0.004 の差に相当する。RIPE の集中度は 32bit 整数演算により求めている。

3.2.2 汎用 EWS 上での実現

RIPE との処理結果の比較を行うために、集中度フィルタを汎用 EWS 上でも実現した。汎用 EWS 上の集中度フィルタは、集中度を求めるために 64bit の実数型の演算を行い、結果を実数で出力する。

3.3 実験結果

集中度フィルタを RIPE 上および汎用 EWS 上で実現し、計算精度と処理時間について実験を行った。実験に用いた条件を表 3 に示す。

表 3: 実験の条件

画像サイズ	512×512×8bit
原画像	胃 X 線 2 重造影像 (線素画像)
基本構成 RIPE	512PE 20MHz 10GOPS 内部演算に 32bit 整数型を使用
汎用 EWS	NS SUN4/40 15.8MIPS 内部演算に 64bit 実数型を使用

3.3.1 計算精度の比較

まず RIPE 上と汎用 EWS 上での集中度フィルタリングを行い、どの程度の違いが現れるかを実験を行った。

表 4: RIPE と汎用 EWS の処理結果の差

局所領域 (外半径, 内半径)	2 画像間の濃度差		
	集中度パターン		2 値化後
	画素数	最大値	画素数
(50, 25)	2026	1	62
(60, 30)	4408	1	89
(70, 35)	4654	1	58

RIPE 上での集中度フィルタはシミュレータで動作させた。シミュレータは処理結果を 0~255 の間の整数として出力する。また汎用 EWS は、実数として出力する。従って、計算精度の比較は汎用 EWS の結果を 0~255 の間の整数に置き換えて行った。胃がんの候補領域を得るには、集中度を 2 値化する必要があるので、集中度に 2 値化処理を施した結果についても比較を行った。同一の画像に対する RIPE および汎用 EWS での処理結果を比較して、濃度値が異なっている画素数と濃度差の最大値を表 4 に示す。表 4 より、処理結果の差は高々濃度 1 (集中度で約 0.004) であり、濃度差の生じた画素数も全体の画像の画素数と比較すると数%である。以上の結果より、RIPE 上での計算結果は汎用 EWS での結果と本質的な差はないと考えられる。

3.3.2 処理時間の比較

次に、RIPE と汎用 EWS で集中度フィルタリングの処理時間の比較を行った。局所領域のサイズの制限で

表 5: RIPE と汎用 EWS との処理時間の比較

局所領域 (外半径, 内半径)	処理時間 [sec]		性能比
	RIPE	汎用 EWS	
(50, 25)	8.49	8.95×10^3	1054
(60, 30)	12.17	11.68×10^3	960
(70, 35)	16.53	14.41×10^3	872

現在の RIPE では処理ができないため、RIPE の処理時間はプログラムのステップ数から算出した。各処理時間には、原画像をメモリに読み込む時間と、処理結果をメモリから書き出す時間は含まれていない。処理時間の比較を表 5 に示す。表 5 の性能比とは、汎用 EWS の処理時間を 1 とした時の RIPE の高速度を示した数字である。RIPE による集中度フィルタリングは、汎用 EWS と比較しておよそ 800 倍から 1000 倍高速であることが知られた。性能比は局所領域が大きくなるにつれて落ちている。これは汎用 EWS が画像の縁の演算を行わないのに対して、SIMD 型の並列計算機である RIPE は同じプログラムですべての画素に対する演算を行っており、局所領域が大きくなればなるほど RIPE の方が演算量が多くなるのが原因である。

3.4 考察

RIPE でより高速に集中度フィルタを処理するための改良点について述べる。現在の RIPE では大幅な処理時間の短縮は望めないで、新たな機能を追加することによって処理時間の短縮が図れないか考察を行った。

集中度フィルタは、処理時間の短縮のために前もって計算可能なものは計算して中間結果を求めておき、画素値によってその中間結果を選択している。RIPE では取り得る画素値の種類だけ条件を並べて中間結果を選択するようにしている。そのため、RIPE の処理のおよそ 80% 以上の時間をこの選択処理に費やす結果となっている。この選択処理の大幅な短縮を図れば、RIPE 上での処理時間も大幅に短縮できることが推測される。そこで、条件を並べずに中間結果を選択可能とするために、RIPE にレジスタ間接でアクセス可能なテーブルを内蔵することを考えた。テーブルへの値のセットは、集中度プログラムを実行する前に行い、テーブルへのアクセスは 1 ステップで行われるものと仮定して、RIPE 上での処理時間を予測した。レジスタ間接でテーブルへのアクセスが可能ならば、取り得る画素値の種類だけ条件を並べる必要がなくなり、画素値によって中間結果を選択することができる。処理時間の比較を

表 6: テーブルを内蔵した RIPE と汎用 EWS との処理時間の比較

局所領域 (外半径, 内半径)	処理時間 [sec]		性能比
	RIPE	汎用 EWS	
(50, 25)	2.29	8.95×10^3	3904
(60, 30)	3.27	11.68×10^3	3571
(70, 35)	4.43	14.41×10^3	3253

表 6 に示す。表 6 より、現在の RIPE にレジスタ間接でアクセス可能なテーブルを内蔵すれば、汎用 EWS と比較しておよそ 3000 倍から 4000 倍の処理の高速化が図れるものと推定された。

4 尺度空間フィルタによる評価

計算量の多いフィルタの 1 つである尺度空間フィルタ (scale-space filter) を実現し、RIPE システムの評価および問題点の解析を行う。以下では、まず尺度空間フィルタの概要について述べる。次に、今回計算機上で実現した時に用いた方法と、その計算精度および処理時間について述べる。そして最後に、RIPE 上でより高速に処理するための改良点について考察を行う。

4.1 尺度空間フィルタの概要

尺度空間フィルタ⁶⁾とは、原画像を様々な尺度でとらえた画像集合に拡張するフィルタである。この画像集合は、帯域幅が可変のガウシアンによって原画像を

様々な段階にはかすことで得られる。この時、ガウシアン帯域幅は画像を観測する際の尺度となっている。さらに、ガウシアンによって得られた画像の2次偏微分(ラプラシアン)の零交差を求めることで、原画像の階層構造を知ることができる。

原画像を $f(x, y)$ 、処理結果を $g(x, y; \sigma)$ とすると、ガウシアンは次のように定義される。

$$g(x, y; \sigma) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(u, v) w(x-u, y-v; \sigma) du dv$$

$$w(x, y; \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right), \quad \sigma > 0 \quad (2)$$

ガウシアン帯域幅は、パラメータである σ によって決まる。 σ が大きくなるにつれてガウシアン帯域幅は広がるので、フィルタリングした結果、原画像の微細な構造が失われていく。つまりパラメータ σ が、画像を観測する際の尺度となる。このことから、パラメータ σ は“尺度パラメータ”と呼ばれる。

また、ガウシアンによって得られた画像 $g(x, y; \sigma)$ は、次の偏微分方程式を満足する。

$$\left(\frac{\partial}{\partial \sigma} - \sigma \nabla^2\right) g(x, y; \sigma)$$

$$= \left(\frac{\partial}{\partial \sigma} - \sigma \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\right) g(x, y; \sigma)$$

$$= 0 \quad (3)$$

この方程式の解集合として得られる画像は、原画像の山と谷の領域を表している。

尺度空間フィルタを実現する方法としては、上記のガウシアンとラプラシアンを用いる方法の他に、LoG (Laplacian of Gaussian) フィルタを用いる方法がある。LoG フィルタは、原画像を $f(x, y)$ 、処理結果を $l(x, y; \sigma)$ とすると次のように定義される。

$$l(x, y; \sigma) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(u, v) w(x-u, y-v; \sigma) du dv$$

$$w(x, y; \sigma) = -\frac{1}{\pi\sigma^4} \left(1 - \frac{x^2+y^2}{2\sigma^2}\right) \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

$$\sigma > 0 \quad (4)$$

4.2 尺度空間フィルタの実現

計算機で離散的に尺度空間フィルタリングを行う場合は、それぞれの重み関数に従う加重マトリクスを使った積和演算に置き換えることができる。

ガウシアンとLoGフィルタの重み関数は、中心からの距離が遠くなるほど0に漸近していく。しかし、計算機で離散的にフィルタリングする際には、局所領域のサイズを有限にしなければならない。そこで、ガウシアンとLoGフィルタの局所領域のサイズは、それぞれ直径 6σ , 8σ を用いることにした。これは、ガウシアンでは直径 6σ の部分、LoGでは直径 8σ 部分までで全体積の99%以上を確保するためである。

加重マトリクスを用いた積和演算は、2.3で示した1次元分解を利用することで処理の高速化を図ることができる。ガウシアンとLoGフィルタは $[M/2]$ 段に分解できるので、2次元の局所領域での処理を行うアルゴリズムに比べて、乗算回数、加算回数ともおよそ半分になる。

4.2.1 RIPE 上での実現

RIPEでは整数型しか使用することができないため、マトリクスの重みを表すのに32bitの符号付き整数型を用いた。重みを整数化する際には、局所領域の全ての濃度値が最大である場合の処理結果が、32bitの最大値になるように定数倍している。また、1次元分解を利用することで演算回数だけでなく、PE間転送回数も減らすことができた。なお、RIPE上での尺度空間フィルタはシミュレータ上で動作させた。

4.2.2 汎用 EWS 上での実現

処理結果の比較を行うために、汎用 EWS 上でも尺度空間フィルタを実現した。RIPEの場合とは異なり、マトリクスの重みを表すのに64bit実数型を用いた。

4.3 実験結果

RIPEおよび汎用 EWS で尺度空間フィルタリングを行なったときの、計算精度および処理時間について実験を行なった。この実験の条件を図7に示す。また、測

表 7: 実験の条件

画像サイズ	512×512×8bit
原画像	SIDBA GIRL
基本構成 RIPE	512PE 20MHz 10GOPS 内部演算に32bit整数型を使用
汎用 EWS	NS SUN4/40 15.8MIPS 内部演算に64bit実数型を使用

定した σ の値の範囲は、ガウシアンでは1.0~10.0、LoGフィルタでは1.0~7.8で、きざみ幅はともに0.2である。

4.3.1 計算精度の評価

マトリクスの重みを表すのに整数を用いた場合、実数を整数化したことによる誤差が生じる可能性がある。そこで、マトリクスの重みを整数で表した場合と実数で表した場合とで、処理結果にどの程度違いが現れるのかを測定した。整数型と実数型の2つのマトリクスを、同じ原画像に適用し、その結果を1画素ずつ比較することで影響を調べた。

ガウシアン処理結果は256階調の画像になる。そこで、重みを実数で表した場合と整数で表した場合の処理結果の差を取り、この時に差の生じた画素数と濃度差の最大値を調べた。その結果、表8に示したように、2画像間での差は高々濃度1であり、濃度差の生

じた画素数も少ない (0.01%以内) ことから、ガウシアン加重マトリクスを整数で表したことによる影響は微小であると考えられる。

表 8: 重みの整数化誤差 (ガウシアン)

σ	局所領域	2 画像間の濃度差	
		画素数	最大値
1.0	7×7	0	0
3.0	19×19	3	1
5.0	31×31	4	1
7.0	43×43	5	1

一方、LoG フィルタの処理結果は正負で 2 値化して出力している。そこで、ガウシアン同様、重みを実数で表して 2 値化した場合と整数で表して 2 値化した場合の処理結果の差を取り、この時に差の生じた画素数を調べた。その結果、表 9 に示したように、それぞれの加重マトリクスで処理した画像に差が生じた。しかし、その画素数が全体の画素数に対してごくわずか (0.03%以内) であることから、LoG フィルタの場合もその重みを表すのに整数を用いたことによる影響は微小であると考えられる。

表 9: 重みの整数化誤差 (LoG フィルタ)

σ	局所領域	2 画像間で値が異なる画素数
1.0	9×9	0
3.0	25×25	11
5.0	41×41	45
7.0	57×57	33

4.3.2 処理時間の比較

次に、RIPE と汎用 EWS を使って尺度空間フィルタリングを行ない、その処理時間を比較した。汎用 EWS の処理時間については、実測した 10 回の処理時間の平均値を用い、RIPE の処理時間については、ステップ数より算出した予測値を用いた。また各処理時間には、原画像をメモリに読み込む時間と、処理結果をメモリから書き出す時間は含まれていない。

表 10 にガウシアンとラプラシアンを用いた場合の処理時間の比較結果を示す。この時、汎用 EWS と RIPE の処理時間は、ガウシアンとラプラシアンの処理時間を加えた値である。また、表 11 に LoG フィルタを用いた場合の処理時間の比較結果を示す。ここで、表中の性能比とは、汎用 EWS の処理時間を 1 とした時の RIPE の高速度を示した数字である。性能比を見て明らかのように、RIPE では汎用 EWS に比べておよそ 100 倍の性能を持っていることが知られた。

表 10: 処理時間の比較 (ガウシアン, ラプラシアン)

σ	局所領域	処理時間 [msec]		性能比
		RIPE (乗算器 無)	汎用 EWS	
1.0	7×7	155.3	16.26×10 ³	105
3.0	19×19	811.4	80.42×10 ³	99
5.0	31×31	1731.9	193.47×10 ³	112
7.0	43×43	2936.0	348.73×10 ³	119

表 11: 処理時間の比較 (LoG フィルタ)

σ	局所領域	処理時間 [msec]		性能比
		RIPE (乗算器 無)	汎用 EWS	
1.0	9×9	205.8	21.58×10 ³	105
3.0	25×25	1302.6	128.51×10 ³	99
5.0	41×41	2740.9	316.40×10 ³	115
7.0	57×57	4855.1	578.77×10 ³	119

4.4 考察

今回実現した尺度空間フィルタでは、汎用 EWS を用いて処理するのに比べてある程度の高速度が図れた。しかしながら、これまでの評価に用いてきたフィルタと比べると、性能比がかなり落ちてしまうことが分かった。その原因は、現在の RIPE チップに乗算器が内蔵されていない点にある。今回の尺度空間フィルタのように、比較的大きな局所領域を用いて、しかも積和演算で用いるマトリクスの重みの桁が大きい場合には、乗算をシフトと加算に展開しステップ数が増加するため、結果的に処理時間が低下する。マトリクスの重みを整数化する際の精度を落すことで、乗算部分のステップ数を減らし高速度を図ることも可能である。いずれにせよ、乗算部分のステップ数が全体に対して 90% 以上を占めていることを考えると、いかにして乗算に費やすステップ数を減少させるかが重要になる。

そこで、RIPE チップに 16bit×16bit の乗算器 (1 クロック動作) が内蔵されていると仮定して処理時間の予測と比較を行なった。表 12 にガウシアンとラプラシアンを用いた場合の処理時間の比較結果、表 13 に LoG フィルタを用いた場合の処理時間の比較結果を示す。ラプラシアンに関しては、乗算器を使うよりもシフトを使った方が高速度のため、乗算器内蔵前と処理時間は変わらない。また、ガウシアンおよび LoG フィルタにおける乗算は、32bit×8bit の演算となるので、1 回につき 5 ステップとして換算した。この結果、乗算器を内蔵した RIPE では、汎用 EWS に対しておよそ 700 倍から 2000 倍の高速度が可能なのと推測される。

参 考 文 献

表 12: 処理時間の比較 (ガウシアン, ラプラシアン)

σ	局所領域	処理時間 [msec]		性能比
		RIPE (乗算器有)	汎用 EWS	
1.0	7×7	22.9	16.26×10 ³	710
3.0	19×19	57.3	80.42×10 ³	1403
5.0	31×31	121.3	193.47×10 ³	1595
7.0	43×43	214.7	348.73×10 ³	1624

表 13: 処理時間の比較 (LoG フィルタ)

σ	局所領域	処理時間 [msec]		性能比
		RIPE (乗算器有)	汎用 EWS	
1.0	9×9	9.9	21.58×10 ³	2180
3.0	25×25	68.2	128.51×10 ³	1884
5.0	41×41	177.2	316.40×10 ³	1786
7.0	57×57	337.9	578.77×10 ³	1713

5 むすび

本稿では、複雑で計算量の多いフィルタを RIPE 上で実現することにより、RIPE システムの問題点の解析および性能評価を行った結果について述べた。

複雑で計算量の多いフィルタを RIPE 上で実現することで、これまで用いてきたような比較的単純な処理を行うフィルタでは評価しきれなかった点や、明らかにならなかった問題点を明確にすることができた。これらの問題点の中には、設計段階で予想されたことも含まれているが、今回の解析によってその重要性を知ることができた。

RIPE チップは現状でも汎用 EWS と比較して高速であるが、乗算器やレジスタ間接でアクセス可能なテーブルに参照できる機能を内蔵することで、さらなる高速化を図ることができる。これらによって集中度フィルタリングや尺度空間フィルタリングは、汎用 EWS のおよそ 700 倍から 4000 倍の高速化が図れることが知られた。

謝辞 本研究に御協力いただいた、新日本製鐵(株)、矢崎総業(株)、(株)豊橋テクノサイエンス、および適切な助言を頂いた、名古屋大学の鳥脇純一郎教授、横井茂樹助教、豊橋技術科学大学の山本真司教授、および豊橋技術科学大学 VLSI 設計研究室の諸賢に深謝します。集中度フィルタの実験試料は国立がんセンター中央病院放射線診断部より提供頂いた。なお本研究の一部は、新技術事業団先端的研究成果展開事業による。

- 1) 本沢邦朗, 佐藤淳, 富田穰太, 今井正治, 長谷川純一: 超高速画像処理システム RIPE のアーキテクチャ, 電子情報通信学会技術研究報告, CPSY89-12, pp. 43-50 (1989).
- 2) Imai, M., Honsawa, K., Yamada, N., Tomita, J., Sato, J. and Hasegawa, J.: The Architecture and Efficiency of RIPE : A Real-time Image Processing Engine, 電子情報通信学会技術報告, CPSY90-31, pp. 113-118 (1990).
- 3) 松下昭司: 超高速画像処理システム RIPE のハードウェアの実現, 豊橋技術科学大学修士学位論文 (1993).
- 4) 平岡久和: 超高速画像処理システム RIPE の性能評価, 豊橋技術科学大学修士学位論文 (1993).
- 5) 長谷川純一, 筒井武敏, 鳥脇純一郎: 胃 X 線 2 重造影像におけるひだ集中を伴うがん病変部の自動抽出, 電子情報通信学会論文誌 (D-II), Vol. J73-D-II, No. 4, pp. 661-669 (1990).
- 6) Witkin, A. P.: Scale-Space filtering, in *Proc. of 8th International Joint Conference on Artificial Intelligence*, pp. 1019-1022 (1983), Karlsruhe.
- 7) 横井茂樹, 鳥脇純一郎, 福村晃夫: 画像処理のための 2 次元フィルタリングの 1 次元分解について, 電子情報通信学会論文誌 (D), Vol. J61-D, No. 7, pp. 512-513 (1978).