

## 方向特徴場の方向性並列演算 (MAP) 法による 地形図の特徴抽出†

山田 博三<sup>††</sup> 山本 和彦<sup>††</sup>  
齊藤 泰一<sup>††</sup> 松井 伸二<sup>†††</sup>

現在地図のデジタル化のための認識処理において、入力の早い段階でベクトル化を行うのが主流になっている。しかし、ベクトル化特徴に対する逐次処理では、ある程度以上は処理を高速化できない上に、ベクトル化で失われた情報は復元できないという問題点を持つ。一方、並列演算法は、専用プロセッサにより高速処理が可能であり、元の画像への再現性にも優れているという長所を持つが、可能な処理の範囲に限られるという欠点を持つ。筆者らは、並列演算による処理の範囲の拡大を目的として、収縮膨張演算を拡張した、方向性特徴場と異方性近傍演算の組合せによる並列演算法を提案してきた。本論文では、この考え方を地形図からの幾何学的特徴の抽出に適用する。その過程で、実際の画像処理系で有用な演算を、MAP (Multi-Angled Parallelism) 演算として整理・体系化する。地形図からの建物 (点の集合)、ハッチング領域、道路、文字領域等の抽出により、手法の有効性を実証した。

### 1. はじめに

地形図には、さまざまな情報が重畳して描かれており、これらを色や幾何学的特徴により分離・抽出することは、マルチメディア理解を進めるうえで重要な課題である。地図の色情報による分離に関して、筆者らは先に、色相の排他性を用いた手法を提案し、各表現色の明確な分離に成功している<sup>1)</sup>。本論文では、幾何学的特徴に着目し、道路、建物、文字領域などの分離・抽出を行う。これらの抽出手法に関しては多くの研究がなされているが<sup>2)</sup>、これらのほとんどでは、処理の早い段階でベクトライズを行い、その後は逐次的な手法を用いている。その主な理由としては、地形図のような大画像の処理には時間がかかることがあげられる。しかし、ベクトライズによる方法では、黒地の内の厚い部分などで、情報が失われてしまうと、元の情報を再現できないという難点がある。

一方、収縮膨張演算などの2値画像における局所並列演算は、専用プロセッサによる高速処理が実現されており、目視検査装置などでひろく用いられている。しかし、専用プロセッサのみでは処理できる内容に限られるという欠点があった。特に、地形図のように複雑な形が重畳している対象では、並列処理を用いよう

としても、処理過程の途中で汎用プロセッサに制御やデータを戻さねばならず、専用プロセッサの持つ高速性が活かせなくなってしまう。この問題を解決するためには、並列演算の枠内で処理できる範囲を拡大する必要がある。そこで、筆者らは、局所並列演算を用いたより多様かつ高次の特徴の抽出を目的とし、方向特徴場に着眼した方向演算を提案した<sup>3)</sup>。そこで、従来の収縮膨張演算の拡張として、方向化を図ることにより、従来不可能であった点状特徴と線状特徴の区別が可能となり、並列演算の適用範囲を広げられることについて論じた。

本論文では、前論文<sup>3)</sup>で提案した考え方を、MAP (Multi-Angled Parallelism) 法として、整理・体系化する。さらに、地形図からの道路、建物、ハッチング領域、文字領域等幾何学的特徴の抽出にMAP法を適用する。実験結果から、MAP法を用いることにより、並列演算の枠組で、地形図の特徴を統一的に抽出できることを実証した。

### 2. 方向性局所並列演算

#### 2.1 エッジの方向定義

2値画像を次のように表す。

$$b(x) = b(i, j)$$

方向  $d$  を、右方向から時計まわりに  $0, 1, 2, \dots, 7$  の8方向に定義し、近傍画素を  $x + e_d$ 、パターン全体を単に  $b$ 、 $d$  方向の近傍点による ( $-e_d$  だけシフトした) パターンを  $b[d]$  で表す。また、方向  $d$  に関する演算は、 $\text{mod } 8$  で行うものとする<sup>3)</sup>。

本論文では主に輪郭点に沿った特徴抽出を行う。輪

† MAP: Multi-Angled Parallelism for Feature Extraction from Topographical Maps by HIROMITSU YAMADA, KAZUHIKO YAMAMOTO, TAIICHI SAITO (Machine Understanding Division, Electrotechnical Laboratory) and SHINJI MATSUI (Technical Engineering Department No. 5, Technical Engineering Group, FUJIFACOM Corp.).

†† 電子技術総合研究所知能情報部

††† 富士ファコム制御(株)技術本部第5技術部

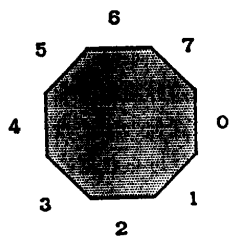


図 1 エッジの方向

Fig. 1 Direction of edge.

郭点上の方向特徴は、各方向に対応する 8 面の 2 値画像  $a_d, d=0\sim 7$  で表される。各方向面  $a_d$  は、 $d$  方向成分がある時 1、なければ 0 である。輪郭画素 (エッジ) の方向は黒から白への法線方向とする。8 角形画像のエッジの方向を図 1 に示す。以下、 $a_d$  の抽出手法について説明する。

まず、下記の演算により  $90^\circ$  ごとのエッジ面  $a_d^0 (d=0, 2, 4, 6)$  を求める。

$$a_d^0 = b \cap \overline{b[d]}$$

ここで、 $\bar{b}$  は、 $b$  の否定である。次に、 $a_d^0$  に対する下記の演算により、斜め方向のエッジ面  $a_d^1 (d=1, 3, 5, 7)$  を求める。

$$a_d^1 = (a_{d+1}^0 \cap a_{d-1}^0) \cup (a_{d+1}^0 \cap \{a_{d-1}^0[d-1] \cup a_{d-1}^0[d+2]\}) \cup (a_{d-1}^0 \cap \{a_{d+1}^0[d-2] \cup a_{d+1}^0[d+1]\})$$

ここまでで、8 方向のエッジ面が得られるが、方向精度が緩いことと、2 値化の際の量子化誤差などにより生ずるエッジの凹凸などノイズの影響を考慮し、2 画素以上の連続を方向定義の条件とする。すなわち、方向性伝播を

$$C_d: C_d b = b \cap \{b[d-2] \cup b[d+2]\}$$

とすれば、

$$a_d' = \begin{cases} S_d^1 M_{a_d^0}^1 C_d a_d^0 & (d=0, 2, 4, 6) \\ S_d^1 M_{a_d^1}^1 C_d a_d^1 & (d=1, 3, 5, 7) \end{cases}$$

ここで、 $S, M$  は、線分に対する短縮、復元を行うための演算で、詳細については後述する。さらに、方向特徴の安定性を増すため、孤立した 2 画素以下の線分を方向定義の段階で除去し  $a_d$  を求める。

$$a_d = M_{a_d'}^1 S_d^1 a_d'$$

以上の処理を行った場合、方向定義されない輪郭点が存在し、これらを無方向性輪郭点として定義し、利用することもできるが、本論文では特にその利用方法について論じていない。

## 2.2 演算子の定義

以下、実験に必要な演算子を定義する。

まず、最も基本的な無方向性の収縮、および膨張を

$E_{8 \text{ or } 4}, D_{8 \text{ or } 4}$  で表す。ここで、8 もしくは、4 の数字は、それぞれ 8 近傍 4 近傍の収縮膨張演算を表す。このような演算子において、右肩の数字は繰返し回数を示し、二つ以上の演算子を並べて書いた場合、右側から順に演算を行うことを示す。例えば、 $D_8^3 E_4^2$  は、4 近傍収縮演算 2 回の後 8 近傍膨張演算 3 回を行うことを示す。

次に、線の短縮などに用いる方向性を持った収縮演算を定義する。

$$S_d': S_d' b = b \cap \{b[d-1] \cup b[d] \cup b[d+1]\}$$

ここで、方向特徴面で線を両端から短縮することを考えれば、

$$S_d: S_d = S_{d-2}' \cap S_{d+2}'$$

と接線方向両側の演算として定義したほうが使いやすい。ここで、 $S_d'$  では、 $d$  方向を向いた扇状の隣接画素を調べているが、前述の演算  $C$  のような単一方向だけを見る演算も考えられる。ただ、輪郭点上の線の短縮を目的とするなら、線幅方向の条件を緩めた前者のほうが実用的である。

線の延長でも同様に、単一方向への延長と、方向精度を緩めた扇状延長が考えられる。前者として、

$$L_d': L_d' b = b \cup b[d]$$

が定義され、後者として、

$$X_d': X_d' b = b \cup b[d-1] \cup b[d] \cup b[d+1]$$

が定義される。これらの演算に関しても、処理の都合上、

$$L_d: L_d = L_{d-2}' \cup L_{d+2}'$$

$$X_d: X_d = X_{d-2}' \cup X_{d+2}'$$

を定義しておく。

局所演算としては、次式のような演算も可能である。

$$F_d': F_d' = L_{d+1}' \cap L_{d-1}'$$

この演算は、背景に対する閉じ状態の抽出に用いられる。すなわち、図形の外接四角形を求める演算として  $F$  を定義する。

$$F: F = F_1' \cup F_3' \cup F_5' \cup F_7'$$

収縮膨張演算を行う際、別の画像  $f$  をマスクとして用いることは、有効な手法である。演算子を以下のように定義する。

$$M_f: M_f b = f \cap D_8 b$$

$$H_f: H_f b = E_8(b \cup f)$$

同様な演算は、方向性演算にも定義可能である。ここでは、マスク付き扇状延長を定義しておく。

$$W_d: W_d f = f \cap X_d'$$

表 1 MAP 演算  
Table 1 MAP operations.

マ ス ク	膨 張		収 縮	
	無	有	無	有
無 方 向	$D$	$M_f$	$E$	$H_f$
扇 状	$X': X$	$W_{a:f}$	$S': S$	—
単 一 方 向	$L': L$	—	$C$	—
そ の 他	$F$	—	$Z$	—

$W_{a:f}$  は背景方向へ拡散を行う演算で、 $L_a$ ,  $X_a$  とは延長方向が異なる。

このように、マスクの有無、収縮膨張、方向精度の組合せにより、さまざまな演算子を定義することができる。ただし、輪郭上の線状特徴を対象とした場合、収縮と膨張の演算の間で対称性はない。これは、原画像での2値は黒と白に対応するのに対して、輪郭上での2値は、有と無に対応することを考えれば納得できる。このため、すべての組合せの演算子が意味を持つわけではない。

この他、論文中で用いる演算として、8近傍に連結画素がない孤立点を除去する局所演算  $Z$  を定義する。

$$Z: Zb = b \cap \left( \bigcup_{d=0}^7 b[d] \right)$$

表 1 に本節で定義した演算を整理する。ここで定義した演算は、画像間の論理演算および近傍論理演算であり、現状の画像処理専用プロセッサで高速処理が可能である。表中、未定義の演算は、本論文では用いていないが、扱う画像の性質によっては有用となるであろう。

なお、前論文<sup>3)</sup>では、ツイスト形演算による 22.5° 方向伝播の提案を行ったが、本論文では、処理の単純化のため、8方向のみを用いている<sup>4)</sup>。

### 3. 基本特徴の抽出

#### 3.1 多色刷り地形図の色分離

国土地理院発行の3色刷り 1/25,000 地形図を実験対象とする。300 dpi のフルカラーキャナ JX-450 を用いて入力された画像は、あらかじめ、色相の排他性をういた色分離手法<sup>1)</sup>により、各表現色ごとに分離される。この手法は、多色刷りの印刷物を処理対象とし、画像の中には特定の表現色以外は存在しないという色の排他性を考慮し、さらに表現色の周辺部分に存在する背景色との間の混合色は、色相空間内で背景色と各表現色を結ぶ直線上に分布することを考慮にいたれたものである。色分離結果は、各代表色と入力色と

の位置関係から、'その色らしさ'を表す濃淡画像として得られる。本論文では、道路、文字、建物、記号を含む 512×512 の黒色の分離結果を2値化し、各種特徴の抽出を行う。この画像の中には、他の表現色で描かれた画像の影響は見受けられない。図 2 はこの手法の処理結果であり、以後の処理では、特に断りが無い限り、この画像を原画像として用いている。

#### 3.2 線状特徴の抽出

まず、線分に関する特徴として、 $a_d$  において、極端に短い線分の除去を行う。

$$f_d^{\text{short}} = M_{a_d}^3 S_{a_d}^3 a_d$$

ここでは、6画素以下の線分を除去するため、振動演算の繰返し回数を設定したが、これらのパラメータは扱う画像の性質によって最適な値を選ぶ必要がある。

図 3 は  $f_d^{\text{short}}$  の 8 面の OR をとった画像である。図 3 の意味は、エッジに垂直な方向に 3 画素以上方向成分の引き続いている点である。処理は各点独立に行われるから、それ以上の線状特徴はすべて抽出されることになる。

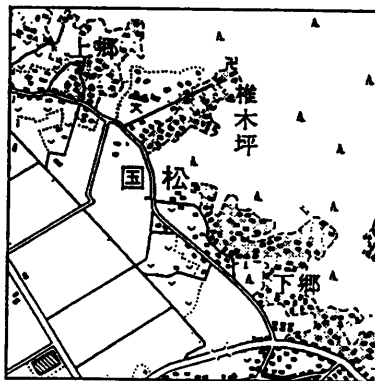


図 2 原画像  $b$   
Fig. 2 Original image  $b$ .

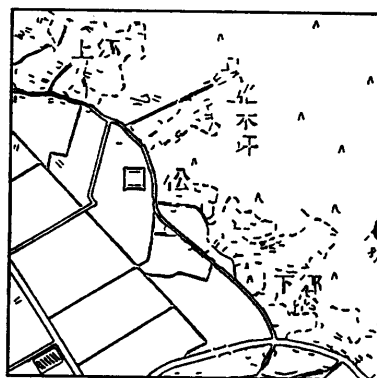


図 3 短い線分  
Fig. 3 Short line segments.

次に、ある程度以上長い線分  $f^{long}$  の抽出を行う。この際  $f^{short}$  での線分を接線方向に延長して、破線やノイズや交差点付近で途切れた線分の連結を行っている。局所並列演算で線分の延長を行うため、まず、線分の端点を抽出する。  $a_d$  における線は、線幅 1 と考えられるため、各画素において隣接画素の状態を示す方向  $h$  の関数  $g(h)$  を以下のようにして求め、

$$g(h) = \begin{cases} 1 & \text{if } f_d^{short}[h] \cap f_d^{short}[h+1] = 1 \\ 0 & \text{otherwise} \end{cases} \quad (h=0 \sim 7)$$

線分上で  $\sum_{d=0}^7 g(d)$  が 1 の点を端点として抽出する。ここで、 $g(h)$  は、点の周りの交差数を求めるための関数である。さらに抽出された端点  $f_d^{end}$  に対し延長を行い、線の連結を行う。

$$f_d^{connect} = f_d^{short} \cup L_d^3 f_d^{end}$$

次に、方向精度の緩和を行う。これは、連続した線ならば、 $90^\circ$  以上曲っていない限り、1本の線として扱うためである。

$$f_d^{relax} = \begin{cases} f_{d-1}^{connect} \cup f_d^{connect} \cup f_{d+1}^{connect} & (d=0, 2, 4, 6) \\ f_d^{connect} & (d=1, 3, 5, 7) \end{cases}$$

$f_d^{relax}$  の中から長い線を抽出する。この際、文字を構成する線分より長いことを条件としている。ここで、縦横方向のみで方向性の緩和を行っているのは、斜めの方向性を緩和し、縦または横方向の伝播も斜め成分として許してしまうと、例えば国構えの外側で(文字幅+文字高さ)の長さの斜め線が定義されてしまい、文字幅または高さで伝播の回数を制御してそれ以上の長さの線を抽出することができなくなってしまうためである。

$$f_d^{long} = M_{f_d^{relax}}^{17} S_d^{17} S_d^3 f_d^{relax}$$

図 4 (a) に 8 面の処理結果を足し合わせて示す。

さらに下記の演算を用いれば、原画像の中から長い線を取り出すことができる。図 4 (b) にこの結果を示す。

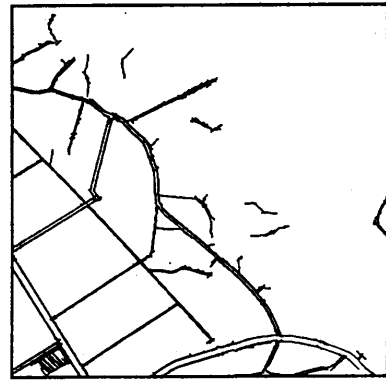
$$f^{line} = M_i^2 \left( \bigcup_{d=0}^7 f_d^{long} \right)$$

### 3.3 点状情報の抽出

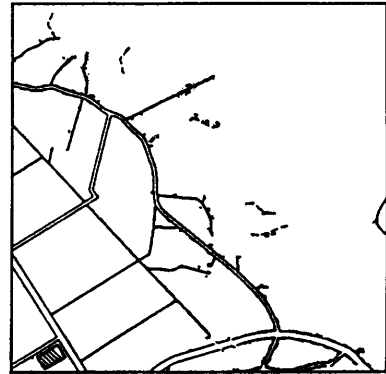
建物の抽出を例にとり、ある面積以上の黒領域を抽出する。

$$f^{block} = M_b^3 E_b^1 E_b^1 b$$

$f^{block}$  では、文字の交差部分を含む場合があり、不十分である。そこで、以下の振動演算により建物を抽出する。



(a)  $f_d^{long}$



(b) Long lines.

図 4 長い線  
Fig. 4 Long lines.

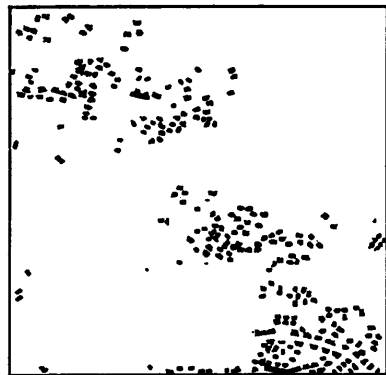


図 5 建物

Fig. 5 Buildings.

$$f^{buil} = f^{block} \cap D_b^5 E_b^5 Z(b \cap f^{block})$$

図 5 に処理結果を示す。

以上のような場合には、方向性演算を必要としないが、点線等の小さな点を抽出しようとする場合、このような無方向性の振動演算では、収縮の際、線分と同時に消えてしまうため、これらを分離することはできない。このため、点の抽出を、 $a_d$  のすべての面で、長

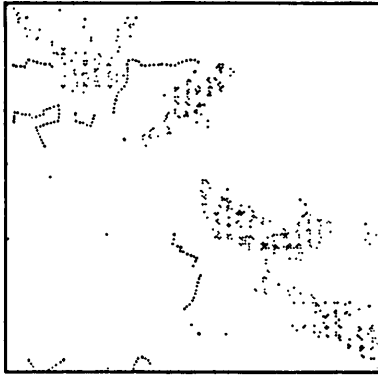


図 6 小さな点  
Fig. 6 Small dots.

さがあるしきい値以下のものとして行う。

$$f^{\text{non-dot}} = M_b^4 \bigcup_{d=0}^7 (M_{a_d}^2 S_d^2 a_d)$$

$$f^{\text{dot}} = b \cap \overline{f^{\text{non-dot}}}$$

図 6 に処理結果を示す。集落内のドットと点線が抽出されている。従来の方向性のない収縮膨張演算では、線と点の区別は不可能であったが、方向性演算を用いることにより、線の抽出が可能となった。その結果、点の抽出も可能となった。

#### 4. 組合せ特徴

手法有効性を示すため、前章で求めた基本特徴をくみあわせ、いくつかの特徴を抽出する。

##### 4.1 ハッチング領域の抽出

建物密集地などを表すハッチングは、右下がりのななめ線の連続として表される。よって、方向特徴面上では、 $a_3$  と  $a_7$  に存在する。

$$f^{3U7} = f_3^{\text{short}} \cup f_7^{\text{short}}$$

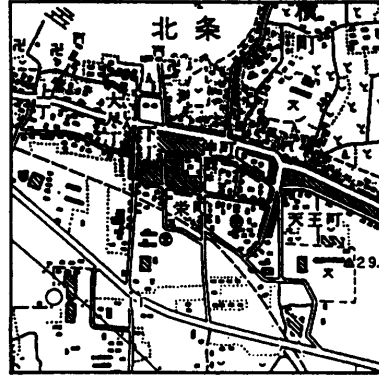
さらに、無方向性の振動演算により、連続したななめ線を抽出する。

$$f^{\text{hatch}} = D_8^4 E_8^6 D_8^2 f^{3U7}$$

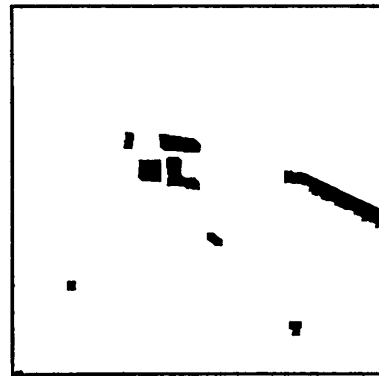
図 7 (a) にハッチング領域を含む原画像を示し図 7 (b) にその処理結果を示す。

文献 5) でも、並列処理を用いたハッチング領域の抽出が行われているが、それに比べて、テンプレートマッチングが不要になるなど、方向別要素による高次特徴表現の効果が現れている。

なお、図 7 (b) では、幅の狭いハッチング領域の検出に失敗している。これは、ハッチングの線分長が足りなかったために生じたものである。解決するためには、ハッチングの線に対して直角方向の平行性を見る必要があるが、このような、より強力な特徴抽出は



(a) Original image.



(b) Hatching region.

図 7 ハッチング領域  
Fig. 7 Hatching region.

今後の一つの課題である。

##### 4.2 道路の抽出

平行線で描かれた道路を長い線分ではさまれた領域として抽出する。扇状の方向性を持った拡散を用いるため、 $f^{\text{long}}$  でいったん緩められた方向性を回復させる。

$$f_d^{\text{narrow}} = f_d^{\text{long}} \cap f_d^{\text{connect}}$$

$f_d^{\text{narrow}}$  を背景方向に向けて、扇状に拡散させる。

$$f_d^{\text{proad}} = W_d \cdot \delta^6 f_d^{\text{narrow}}$$

2本の線にはさまれた領域は、対向する二つの方向特徴面で拡散を受ける。そこで、対向する面に存在する情報を  $f_d^{\text{road}}$  として抽出する。

$$f_d^{\text{road}} = f_d^{\text{proad}} \cap f_{d+4}^{\text{proad}}$$

図 8 に 8面の処理結果を足し合わせて示す。

平行線による道路については、文献 6) で並列処理による抽出がされているが、本論文では、方向性演算を導入しているため、交差点付近での接続を線の“延長”として考えられる等、処理が単純になっている。次節では、さらに複雑な特徴として、文字領域の抽出を行う。

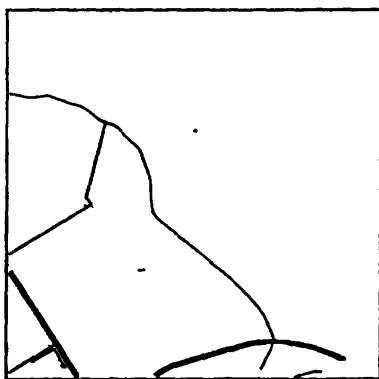


図 8 平行線で描かれた道路  
Fig. 8 Road by parallel lines.

#### 4.3 文字領域の抽出

地形図の高次理解を行ううえで、文字は必要不可欠な情報源である。

地形図のような複雑に重畳した画像の中からの文字認識においては、切り出しと認識の同時処理が必要となるが、そのための基礎実験としてわれわれは全面マッチング文字認識を行った<sup>7)</sup>。その中では、画像内のすべての点で、辞書とのマッチングをとり、類似度の高い点を文字として認識した。複雑な画像に重畳して描かれた文字の認識法としては、本質的な考え方が、計算量が膨大なものになる。ここで、文字が含まれる領域を限定できれば、計算量を削減することができる。

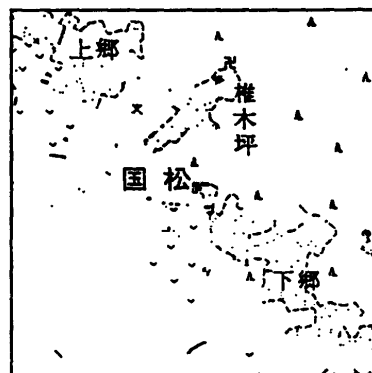
このような“文字らしい領域”は、ある範囲の長さの線分が密集する部分として抽出される。ここでは、原画像から前章で求めた長い線、小さな点、建物を引くことにより求める。原画像  $b$  で  $f^{line}$  に対応する  $f^{line}$  と、 $f^{buil}$ 、 $f^{dot}$  とを合わせて  $b$  から除去する。

$$f^{mod} = b \cap \overline{f^{line}} \cap \overline{f^{dot}} \cap \overline{f^{buil}}$$

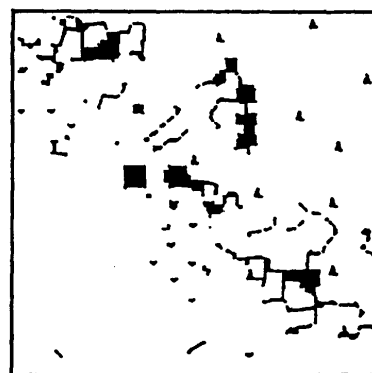
図 9 (a) に  $f^{mod}$  を示す。この図で線分が密な部分を抽出するため  $E_8^6 D_8^6 Z f^{mod}$  の振動演算を行った結果を図 9 (b) に示す。図 9 (b) から、文字よりも小さい記号を取り除こうとする場合、構造の簡単な“上”や“下”などの文字が同時に消えてしまうことは容易に想像がつく。このため、背景に対して“閉じ”状態の部分を文字の一部として埋めた後、さらに、振動演算を施す。文字領域は、次式より求められる。

$$f^{char} = E_8^6 F^{32} E_8^6 D_8^6 Z f^{mod}$$

ここで求めた領域は、文字の中心部を示すもので、高次処理に対して文字の存在する位置を示すためには十分であろう。これにより、前述の全面マッチング文字



(a)  $f^{mod}$



(b)  $E_8^6 D_8^6 Z f^{mod}$

図 9 文字領域抽出過程  
Fig. 9 Process of character extraction.

認識の計算量は、1/20 程度になる。図 10 (a) に  $f^{char}$  を示し、効果をわかりやすく示すため、 $b \cap D_8^6 f^{char}$  を図 10 (b) に示す。

今回の実験では、結果の安定性を重視するという観点から、なるべく、それぞれの特徴を独立に抽出する、という方針をとった。例えば、図 10 (b) において、“下郷”の周辺などで建物が残っているが、これは建物の抽出結果とは独立に演算  $F$  を行ったためである。これに対して、図 11 に示すように、建物や小さな点の抽出結果を考慮に入れて演算  $F$  を行えば、もっときれいな結果が得られる。

しかし、処理を逐次形に並べてしまうと、以前の処理で起こった欠陥がそのまま次の結果へ引き継がれてしまい、結果の安定性が損なわれてしまうのである。並列処理の一つの長所として、各点で処理を独立に行うために、1カ所の欠損が他に影響せず結果が安定であるという点があげられるが、ここでも、並列処理の持つそのような長所を活かした結果を得るようにした。並列処理によりこのようにして得られた結果に対する補填は、次節で述べるような逐次処理により可能

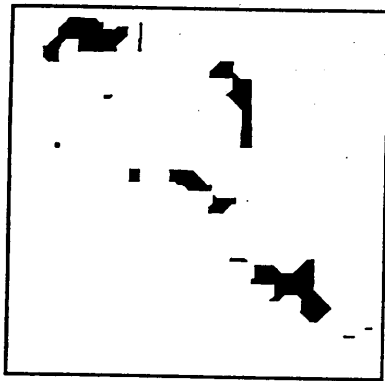
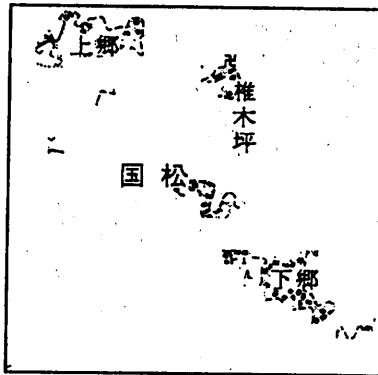
(a)  $f^{char}$ (b)  $b \cap D_s^* f^{char}$ 

図 10 文字領域

Fig. 10 Character region.

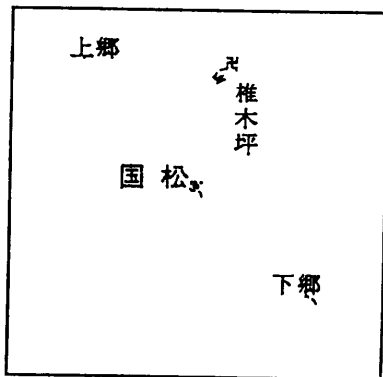


図 11 文字領域の整形

Fig. 11 Refined character region.

であろう。このような接続により、従来に比べて高いレベルでの並列処理と逐次処理の役割分担が可能である。

#### 4.4 逐次処理への接続

前論文<sup>3)</sup>では、方向性並列演算の一つの拡張として、線状特徴の組合せ特徴としての角(カド)への適用について述べた。高次処理へのもう一つの重要な接

点として、逐次処理への接続がある。すなわち、方向性表現の導入により、並列処理の範囲が従来に比べて広がったとはいえ、各点での均一処理では不可能になる場合や、極端に効率が落ちる場合が生ずる。今回の例で言えば図8の道路の抽出において曲がりの強い部分は方向別特徴の足し合わせでは道路としての抽出が困難である。このような部分はむしろその部分に着目した個別の処理を用いたほうが効率がよいと考えられる。そのような現象はおもに線状特徴の端点部分で生じる。したがって、線状特徴の端点部分を抽出することは、高次処理との接続のためには重要である。手法としては、細線化処理を用いれば、並列演算を用いた線状特徴の端点抽出は容易である。

#### 5. おわりに

MAP法(方向特徴場による方向性並列演算)を地形図の特徴抽出に適用し、並列処理の枠内で、道路・建物・ハッチング領域・文字領域等が抽出できることを確認した。この際、手法の評価という観点から、逐次的な処理を用いなかったが、線状特徴の端点の抽出により逐次処理への接続が可能であることを考察した。本論文で用いる並列演算の枠組は、面間の論理演算、近傍演算であり、いわゆる画像処理専用プロセッサで高速計算が可能である。今後の課題として、演算反復回数等パラメータの安定性の評価、(半)自動学習があげられる。また、地形図以外の対象に対する適用により、方向特徴以外の高次特徴場への拡張も今後の課題である。

**謝辞** 日頃ご討論いただき電子技術総合研究所画像研究室の皆様へ感謝します。なお本研究は通産省大型プロジェクト「電子計算機相互運用データベースシステムの研究開発」の一環として行われた。

#### 参考文献

- 1) 松井伸二, 山田博三, 斉藤泰一, 村木 茂, 山本和彦: 色相の排他性を用いた地図の色分離手法, 第37回情報処理学会全国大会論文集, 3V-4 (1988).
- 2) 山田豊通, 小杉 信: 地図認識, テレビジョン学会誌, Vol. 41, No. 10, pp. 928-931 (1987).
- 3) 山田博三, 山本和彦: 方向特徴場における2値画像局所並列演算, 信学論D, Vol. J72-D-II, No. 5, pp. 678-685 (1989).
- 4) 松井伸二, 山田博三, 斉藤泰一, 村木 茂, 山本和彦: 2値画像方向性局所並列演算を用いた地形図の特徴抽出, 信学技報, PRU 88-76 (1988).
- 5) 松沢成子, 宮武孝文, 松島 整: ハッチング領域

抽出アルゴリズムとその地形図への応用, 信学総全大, No. 1558 (1984).

- 6) 宮武孝文, 松島 整, 江尻正員: 平行線抽出手法を用いた地図からの道路情報自動抽出, 信学論 D, Vol. J 68-D, No. 2, pp. 153-160 (1985).
- 7) 瀬野英二, 山田博三, 齊藤泰一, 村木 茂, 山本和彦: 全面マッチングによる地形図の文字認識, 昭和 63 年電気関係学会関西支部連合大会, No. G 9-10 (1988).

(平成元年 7 月 5 日受付)

(平成 2 年 4 月 17 日採録)



山田 博三 (正会員)

昭和 47 年早稲田大学大学院修士課程修了。同年電子技術総合研究所入所, 以来手書文字認識, 動的計画法による形の認識, 並列演算による特徴抽出アルゴリズムの研究に従事。現在, 同所知能情報部画像研究室主任研究官。昭和 60 年 7 月から 1 年間カナダ NRC 電気工学部客員研究員。工学博士。共著「文字の科学」(法大出版)。



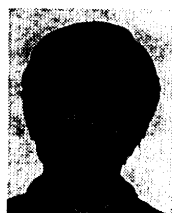
山本 和彦 (正会員)

昭和 44 年東京電機大学工学部電気通信卒業。昭和 46 年同大学院修士課程修了。同年電子技術総合研究所入所, 手書き文字認識の研究に従事。昭和 54~55 年米国メリーランド大学のコンピュータビジョン研究所で並列分散処理による画像理解の研究。現在電子技術総合研究所知能情報部画像研究室室長。人工知能, パターン認識, 特に図形の柔軟な対応付けの研究に従事。工学博士。電子情報通信学会, 電気学会, IEEE, AVIRG 各会員。



齊藤 泰一

昭和 49 年東京都立大学工学部電気科卒業。昭和 44 年電子技術総合研究所入所。現在, 知能情報部画像研究室にて, 文字・画像認識の研究に従事。電子情報通信学会会員。



松井 伸二

昭和 60 年埼玉大学工学部電気工学科卒業。同年富士電機(株)入社。以来漢字 OCR の研究・開発に従事。昭和 63 年 4 月より 1 年間電総研画像研究室で技術指導を受ける。

この間, 並列演算による特徴抽出アルゴリズムの研究に従事。現在富士ファコム制御(株)技術本部第 5 技術部第 2 開発課に勤務。