

濃淡画像の等高面トリーとその地形図検索への応用<sup>†</sup>杉 山 明<sup>††</sup> 鳥 脇 純一郎<sup>†††</sup>

本論文では、画像の各画素の濃度が標高値を表す地形図画像より、等高面トリーとよばれる構造情報を自動抽出する方法を提案する。等高面トリーは、頂点と鞍部の標高、および、一定高度以上の領域の包含関係を木構造を用いて表現したものである。地形図画像より等高面トリーを抽出する方法として、(i)しきい値処理による領域の包含関係を用いる方法、および(ii)頂点を中心とした領域分割と領域の隣接関係を用いる方法、を開発した。この結果抽出される等高面トリーは原地形図画像に比べてそのデータ量が数%以下となることが知られた。次に、等高面トリーの応用として、等高面トリーに標高値と座標値を付属情報として加えたものを原地形図画像のスケッチとして扱い、このスケッチを検索のキーとして地形図データベースへの適用を試みた。検索の例として、国土数値情報の標高値メッシュデータから切り出した種々の大きさの検索画像に対して、検索画像が40×40画素のとき76%、60×60画素以上であれば90%以上の高率で正しく検索された。等高面トリーはデータ量にして原画像の数%であるが、画像の大局的構造をよく表しており、簡単なマッチングで検索への応用も可能であるため、画像の解析に有効であることが確かめられた。

## 1. ま え が き

近年、画像処理の適用分野が広がるに伴い、大量のまたは大規模な画像データを蓄積・管理し、利用者が必要とする情報を効率よく検索できるような画像データベースシステムの開発・研究が強く望まれている。

画像データベースの検索方式としては、(i)画像識別子による検索、(ii)付属情報による検索、(iii)画像特徴による検索、があるが、画像独自のものとして(iv)の方式の研究が重要である。筆者らは、その一例として“スケッチ”の利用を提案し<sup>1)</sup>、胸部X線像処理に適用した<sup>2)</sup>。本文では、これとは異なるスケッチの例として、等高面トリーをとりあげ、それをデジタル化地形図へ適用する。

本文で扱う等高面トリーという概念は、地形表現の手段としての等高線図の研究に基づき、Markによって導入された<sup>4)</sup>。等高線は、地形をある標高値で切ったときにできる等高面の縁を表す線であり、等高線図は、ある間隔(等高線間隔)で求めた等高線の集まりである。等高線図において、一つの等高線は他の多くの等高線を囲むことがあるが、それを直接に囲む

等高線はただ一つである。このような等高線の包含関係は、グラフで表現でき、このグラフは木となる。Freemanら<sup>3)</sup>は等高線を枝、包含関係を節点とするグラフを、Mark<sup>4)</sup>は、逆に等高線を節点、包含関係を枝とするグラフ(contour tree; 図1(b))をそれぞれ提案した。

さらに、Markは地形表面を連続曲面とみなしてその等高線間隔を0に近づけたときの極限として、無限の節点と無限の枝をもつ contour tree を考察した。この極限的な木で意味をもつのは、最も端の節点(次数1の節点)と分岐の起こる節点(次数3以上の節点)であり、これらはそれぞれ頂点と鞍部点という地形の特徴点(critical point)に相当する。そこで、この地形の特徴点にのみ注目したとき、頂点を終端節点、鞍部点を中間節点とするグラフを等高面トリー(surface tree—以下 s-tree と書く; 図1(c))とよぶ<sup>4)</sup>。しかるに、contour tree が等高線図の解析にしばしば用いられてきたのに対して<sup>5)~7)</sup>、s-tree は連続な曲面に対して定義されたのみで、実際のデジタル画像処理への応用例はなく、また、その性質の組織的な検討もなされていない。

そこで、本文では濃淡画像の構造抽出の一方法として s-tree を利用することを試み、まず、デジタル濃淡画像からその s-tree を抽出する手順を明らかにする。次に、その具体的応用として、画像データベースの検索におけるスケッチとしての利用を提案する。

このときの対象画像としては、デジタル濃淡画像という意味で、国土数値情報(いわゆるメッシュデー

<sup>†</sup> The Surface Tree of a Digitized Gray Picture and Its Application to Retrieval of Digitized Terrain Elevation Data by AKIRA SUGIYAMA (Department of Information and Computer Sciences, Toyohashi University of Technology) and JUN-ICHIRO TORIWAKI (Department of Electronics, School of Engineering, Nagoya University).

<sup>††</sup> 豊橋技術科学大学情報工学系

<sup>†††</sup> 名古屋大学工学部電子工学科

\* 現在 (株)キャノン

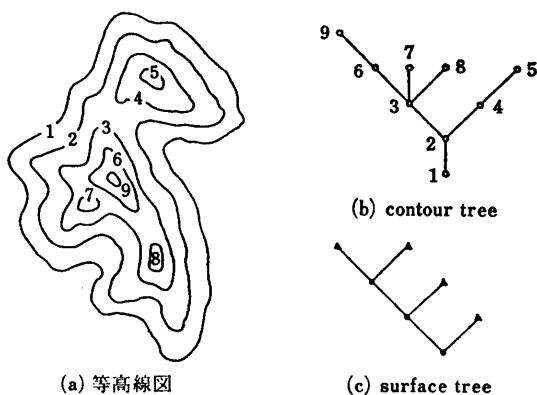


図 1 等高線図とグラフ表現

Fig. 1 A contour line map and its graph representations.

タ)<sup>11),12)</sup>の一つである標高値情報を用いる。なお、従来の地図情報処理は、もっぱら等高線図または通常の地図を対象としており<sup>14)</sup>、正方形メッシュの各画素に情報を割り当てたデジタル化地形図の処理はほとんどみられない。これに対して、筆者らは画像の構造特徴（検索のキーとなるスケッチ）として構造線の抽出法<sup>8),9)</sup>を開発しており、本文の s-tree はそれとは相補の関係にある異なる種類の構造情報を付け加えるものである。

以下2章で s-tree の定義とそのデジタル化地形図からの抽出法等について述べた後、3章では、s-tree を用いた地形図データベースの検索手順について述べる。そして、4章では、検索実験の結果を示し、最後に5章で種々の問題点を考察する。

## 2. 等高度面トリー

### 2.1 基礎的諸定義

本節では、以下の説明に必要な用語を定義し、さらに基本的な考え方を述べる。画素、8近傍、画素間の連結、連結成分等は従来の慣用に従う<sup>10)</sup>。なお、以下の説明では、直観的な理解を容易にするために、対象画像としてデジタル化地形図（以下たんに地形図という）を想定した用語を用いるが、s-tree 自体は任意の濃淡画像に適用できる。地形図  $F = \{f_{ij}\}$  の第  $i$  行  $j$  列の画素の濃度  $f_{ij}$  はその点の標高値を表す。

〔定義1〕（頂点）地形図  $F = \{f_{ij}\}$  において  $f_{ij} \geq f_{mn}$ ,  $\forall (m, n) \in N_{8,i,j}^{(8)}$  ( $= (i, j)$  の8近傍)なる画素  $(i, j)$  を局所極大点とよぶ。 $F$  の局所極大点のつくる連結成分のなかで、その連結成分に隣接するすべての画素よりも濃度値が大なる連結成分のおのをおの  $F$

の頂点とよぶ。

〔定義2〕（分枝）地形図  $F$  で、しきい値  $T_i$  以上の標高値の画素からなる連結成分の一つを  $C^{(i)}$  とする。しきい値を  $T_i$  から順に増やしていくとき、あるしきい値  $T_j (> T_i)$  で、初めて  $C_1^{(j)} \subseteq C^{(i)}$ ,  $C_2^{(j)} \subseteq C^{(i)}$ , ...,  $C_m^{(j)} \subseteq C^{(i)}$  を満足する2個以上の連結成分  $C_1^{(j)}, C_2^{(j)}, \dots, C_m^{(j)}$  が生じたとき、領域  $C^{(i)}$  は標高値  $T_j$  で  $m$  個の連結成分に分岐したという。

〔定義3〕（統合）定義2とは逆に、しきい値  $T_j$  で2個以上の連結成分  $C_1^{(j)}, C_2^{(j)}, \dots, C_m^{(j)}$  が存在し、しきい値を  $T_j$  から順に減じていくとき、あるしきい値  $T_i (< T_j)$  で初めて  $C_1^{(i)} \subseteq C^{(j)}$ ,  $C_2^{(i)} \subseteq C^{(j)}$ , ...,  $C_m^{(i)} \subseteq C^{(j)}$  なる一つの連結成分  $C^{(i)}$  が生じたとき、領域  $C_1^{(j)}, C_2^{(j)}, \dots, C_m^{(j)}$  は標高値  $T_i$  で1個の連結成分に統合されたという。また、 $C^{(i)}$  は  $C_1^{(j)}, C_2^{(j)}, \dots, C_m^{(j)}$  を覆うという。

〔定義4〕（等高度面トリー）定義2の連結成分  $C^{(i)}$  のおのをおの節点に対応させる。 $C^{(i)}$  が初めて2個以上の連結成分  $C_1^{(j)}, C_2^{(j)}, \dots, C_m^{(j)}$  に分岐したとき、 $C^{(i)}$  に対応する節点と  $C_1^{(j)}, C_2^{(j)}, \dots, C_m^{(j)}$  に対応する節点を枝で結ぶ。分岐の生じるすべての節点にこの操作を施して得られるグラフ  $G$  を等高度面トリー (surface tree—s-tree とかく) という。s-tree の名称は次の性質に基づく。

〔性質1〕 定義4のグラフ  $G$  は、頂点を終端節点、分岐する直前の連結成分（すなわち、統合結果の連結成分）を中間節点とする木となる。この中間節点には3本以上、終端節点には1本の枝が結ばれる。

これらの概念の直観的な説明は次のとおりである。定義1は、文字どおり山の頂点である。メッシュデータの場合、頂点は必ずしも1点に定まるとはいえず、複数の画素からなる領域として存在することもある。また定義1は、「しきい値  $T_i$  に対応する連結成分  $C^{(i)}$  がすべてのしきい値  $T_j (> T_i)$  に対して、 $C_m^{(j)} \subseteq C^{(i)}$  となる連結成分をもたないとき、この連結成分  $C^{(i)}$  を頂点とする」と定義することもできる。定義2は、たとえば二つの頂点をもつ山を下から上へ眺めていくと、2頂点の間の鞍部点\*の標高より高い部分は、別の頂点をもつ二つの山とみなされることを意味する。定義3は、逆に上から下へ眺めていくと別々の頂点をもつ二つの山が鞍部点の標高以下では一つの山に含まれることを意味する。このように s-tree は地形の頂点と鞍部点にのみ注目して、その隣接関係をグ

\* 鞍部点も頂点と同様に領域として存在しうる。

ラフとして表現したものである。

## 2.2 s-tree の抽出

ここではデジタル化地形図より s-tree を抽出する手順を示す。その手順は、(1) 終端節点—すなわち頂点 (定義1) の抽出、(2) 中間節点 (分岐・統合の発生標高) の抽出、の二つのステップに分けられる。

### 2.2.1 終端節点 (頂点節点) の抽出法

入力地形図画像 ( $F = \{f_{ij}\}$ ) より頂点画像  $G = \{g_{ij}\}$  を抽出して、各頂点を終端節点として s-tree へ登録する。頂点 ((定義1) 参照) の抽出手順にはとくにむずかしい点はないため、手順の詳細は省略する。頂点画像  $G = \{g_{ij}\}$  では、画素  $(i, j)$  が頂点ならば  $g_{ij} = 1$ 、その他は  $g_{ij} = 0$  である。

抽出された各頂点の位置 (連結成分はその重心の位置) と標高を s-tree を記憶する2次元配列に入れる。仮頂点\*の場合は、あらかじめ定められた符号を s-tree 上にも記入しておく。

### 2.2.2 中間節点 (分岐節点) の抽出

[方法1] しきい値処理による領域の包含関係を用いる方法

入力地形図画像  $F = \{f_{ij}\}$  中の最大(小) 値を  $f_{\max} = \max_{ij} \{f_{ij}\}$  ( $f_{\min} = \min_{ij} \{f_{ij}\}$ ) とおく。

[アルゴリズム1] (1) 初期しきい値の決定: あらかじめ設定されたしきい値間隔  $\Delta T$  の整数倍で  $f_{\max}$  を超えない最大の整数  $T_0$  を選び、初期しきい値  $T = T_0$  とする。

(2) しきい値処理: 画像  $F' = \{f'_{ij}\}$  を次式で求める。

$$f'_{ij} = \begin{cases} 1, & f_{ij} \geq T, \text{ または } g_{ij} \geq 1 \\ 0, & \text{その他} \end{cases}$$

ただし、 $G = \{g_{ij}\}$  は 2.2.1 項で得られた頂点画像。

(3) 画像比較:  $F'$  と  $G$  の正值画素がつくる各連結成分間の包含関係を調べる。 $G$  の2個以上の連結成分が、 $F'$  の1個の連結成分で覆われる場合、標高値  $T$  ( $T$  は現在のしきい値) で統合が生じる (定義3) として s-tree へ登録する。

(4) 終了判定:  $F'$  に1個の連結成分しか存在しないか、または  $T \leq f_{\min} + \Delta T$  ならば停止。このとき  $F'$  に2個以上の連結成分が存在するならば、それを統合するものとして仮想の節点を根として s-tree へ登録する。 $F'$  に2個以上の連結成分が存在し、かつ、

$T > f_{\min} + \Delta T$  ならば(5)へ。

(5) しきい値の更新:  $T \leftarrow T - \Delta T, G \leftarrow F'$ 。(2)へ。

この方法で抽出される s-tree は、厳密に言えばしきい値間隔  $\Delta T = 1$  のとき以外 ( $\Delta T \geq 2$  のとき) は、定義4を満足しないが\*、その場合でも  $\Delta T$  が小さければ s-tree のよい近似を与えられるので、とくに区別はしない。

[方法2] 頂点を中心とした領域分割と領域の隣接関係を用いる方法

[アルゴリズム2] (1) 頂点を中心とした領域分割: 2.2.1 項で得られる頂点画像の連結成分にラベル付けしたものを  $G^{(0)}$ 、入力地形図画像を  $F$  とし、次の手順によって画像系列  $G^{(k)} = \{g_{ij}^{(k)}\}$ ,  $k=1, 2, \dots$  を生成する。

(i)  $g_{ij}^{(k)} = g_{pq}^{(k-1)}$ 。ただし、 $f_{pq} = \max \{f_{kl}; (k, l) \in N_{(i, j)}^{(8)}\}$ 。すなわち  $(i, j)$  の8近傍のなかで標高値の最大の点  $(p, q)$  のラベル  $g_{pq}^{(k-1)}$  を  $(i, j)$  に入れる。

(ii) ある整数  $n$  に対して  $G^{(n)}$  と  $G^{(n+1)}$  が一致するとき反復を終了する。そうでなければ  $k \leftarrow k+1$ , (i)へ。

この結果  $G^{(n)}$  は  $L_{\max}$  個の排反な領域に分割され、各領域はラベル  $l$  ( $1 \leq l \leq L_{\max}$ ) をもつ。これを再び  $G$  とおく\*\*。

(2) 領域間の隣接関係を示す隣接表の作成

隣接表 ADJ は2次元配列で表現され、その要素  $\text{ADJ}(i, j)$  はラベル  $i$  をもつ領域とラベル  $j$  をもつ領域が隣接しているならば隣接標高 (二つの領域の接する最高標高—鞍部の標高) が入り、隣接していなければ0が入る。隣接表は次の手順で作成される。

(i) 隣接表の全要素を0に初期化する。

(ii) 画像  $G$  を走査し、ラベル  $l_1$  をもつ画素  $(i, j)$  とその8近傍で  $l_1$  と異なるラベル  $l_2$  をもつ画素  $(p, q)$  が存在するとき、画素の組  $((i, j), (p, q))$  において低いほうの標高値を  $h$  ( $h = \min \{f_{ij}, f_{pq}\}$ ) とする。隣接表において、もし  $h > \text{ADJ}(l_1, l_2)$  ならば、 $\text{ADJ}(l_1, l_2)$  を  $h$  で置き換える。

この手順で画像  $G$  のすべての画素を走査した結果、求める隣接表が得られる。

(3) 分岐関係の抽出

上で述べた隣接表 ADJ から s-tree を作る。その

\* 頂点に8隣接する画素の一部が入力画像外にあるため、当面の図面内部では頂点と同様の性質をもつが、それだけでは頂点と断定できないものを仮頂点と呼ぶ。

\* 標高値が整数で与えられているとする。

\*\* (1)の処理は横井茂樹氏 (名古屋大学) の提案による。

具体的な手順にはとくにむずかしい点はないため詳細は省く。

### 2.3 計算機内での s-tree の表現

s-tree は節点情報のリストとして大きさ  $n \times 4$  語の 2 次元配列 ( $n$  は節点の総数) に記憶される。各節点の入る要素の行番号がその節点のラベルとして用いられる。詳細は文献 13) を参照。

## 3. s-tree の検索への応用

### 3.1 検索の概要

s-tree は、原画像の大局的な構造の情報をもち、画像の構造解析に利用できる。ここでは、s-tree に標高値と位置を付属情報として加えたものを原画像のスケッチとみなし、これを検索のキーとして用いることを試みる。ここで検索とは、入力された画像（地形図の一部—以下検索画像とよぶ）が原画像のどこに位置するか、あるいはどの部分に対応するかを調べることをいう。その手順は、まず検索の対象となる原画像全体の s-tree（以下、原トリーとよぶ）をあらかじめ抽出して、データベースシステム側に記憶しておく。検索画像が入力されると、それより s-tree（以下、検索トリーとよぶ）を抽出して原トリーと検索トリーのマッチングを行い、その結果を用いて原画像内での位置決めを行う。

### 3.2 s-tree のマッチング

s-tree では頂点節点のみが位置の情報をもち、s-tree 自体が画像の大局的な構造を表現したものである。頂点節点を中心とした簡単なマッチングを行う。原トリーと検索トリーのマッチングにおいて、頂点節点から見て  $i$  段階上位までの照合を行うとき、レベル  $i$  のマッチングとよぶ。ただし、 $i=0, 1, 2, \dots$ 、 $i=0$  は頂点節点のみのマッチングを表す（図 2）。以下図の記号を用いる。

s-tree の節点は標高値の情報をもつから、種類と標高値の一致する節点是对应するものとみなす。具体的

には下記の条件を満足するとき、検索トリーの頂点  $P_R$  と原トリーの頂点  $P_0$  は対応するものとみなす。ただし、 $h_i, h'_i$  は、それぞれ、該当頂点節点より  $i$  段階上の節点の標高値を示す。

レベル  $n$  : 頂点節点とその上位  $n$  段階の分岐節点標高によるマッチング

条件 : すべての  $i (i=0, 1, \dots, n)$  について  $h_i = h'_i$  するとき

上記のマッチングを行った結果、検索トリーのある一つの頂点節点にマッチする原トリーの頂点節点の数（同状態頂点数とよぶ）を  $A$  とする。  $A$  の値によって次のような場合が考えられる。①  $A=0$  のとき : マッチする頂点節点がない場合である。レベル 0 のマッチングでは対応する頂点が原画像内に存在しない。レベル 1 以上のマッチングでは、同一標高の頂点節点であってもその上位の分岐節点の標高が異なる場合である。これは検索領域の切り出しの副作用で偽の分岐が起きた場合に生じる。この種の頂点節点は検索には用いられない。②  $A=1$  のとき : 検索トリーの頂点節点に適合する頂点節点が原画像中で一意に定まる場合であり、この種の頂点節点が多いほど検索に有効となる。③  $A \geq 2$  のとき : マッチング候補頂点節点が  $A$  の値に等しい個数だけ出る。

### 3.3 検索画像の位置決め

通常は検索画像内には複数個の頂点が存在するため、すべての頂点に対して上記のマッチングを行う。その結果によって、次のような処理を行う。

(1)  $A=1$  の頂点節点がただ 1 個の場合 : 対応する原トリーの頂点節点から検索画像の位置が定まる。ただし、s-tree のみではこの結果を確認することはできない。

(2)  $A=1$  の頂点節点が 2 個以上ある場合 : 検索トリーの頂点節点と原トリーの頂点節点の相互位置関係を調べることによって確認を行う。いま、 $A=1$  の頂点節点が  $K$  個あるとし、それらの検索画像内の位置を  $(i_{R1}, j_{R1}), (i_{R2}, j_{R2}), \dots, (i_{RK}, j_{RK})$ 、対応する原トリーの頂点の座標（原画像内での位置）を  $(i_{01}, j_{01}), (i_{02}, j_{02}), \dots, (i_{0K}, j_{0K})$ 、とする。まずこのなかより  $|i_{Rm} - i_{Rn}| = |i_{0m} - i_{0n}|$ 、 $|j_{Rm} - j_{Rn}| = |j_{0m} - j_{0n}|$  を満足する二つの頂点節点  $(i_m, j_m)$ 、 $(i_n, j_n)$  を任意に選ぶ。次にこの二つの一方と残りの各頂点節点との間で同様の関係を満足するかどうかを調べる。このときこの関係を満足しない頂点節点は、 $A=1$  であるが誤って適合したものであり、検索には用いない。このようにし

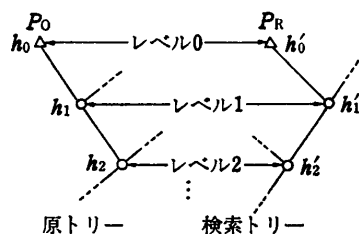


図 2 マッチングのレベル

Fig. 2 Level of matching.

て選ばれた  $A=1$  の頂点節点群を用いて、原画像内での位置を定める。

(3)  $A \geq 2$  の頂点節点のみの場合：対応する原トリーの頂点節点の位置すべてを検索画像の位置の候補として出力する（検索画像内にも複数個の頂点が存在しうる）。当然出力は一意には定められない。

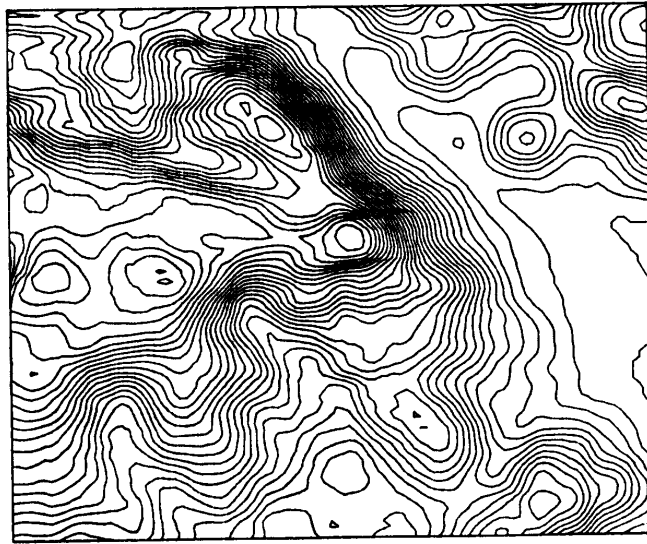
なお、この段階の処理は本論の主旨ではないので、これ以上精密化は行わないが、検討の余地はきわめて大きい。たとえば、(1)の場合でも  $A \geq 2$  の頂点節点があれば補助手段として利用できる。同様に(3)の場

合にもより複雑な手順は容易に考えられよう。

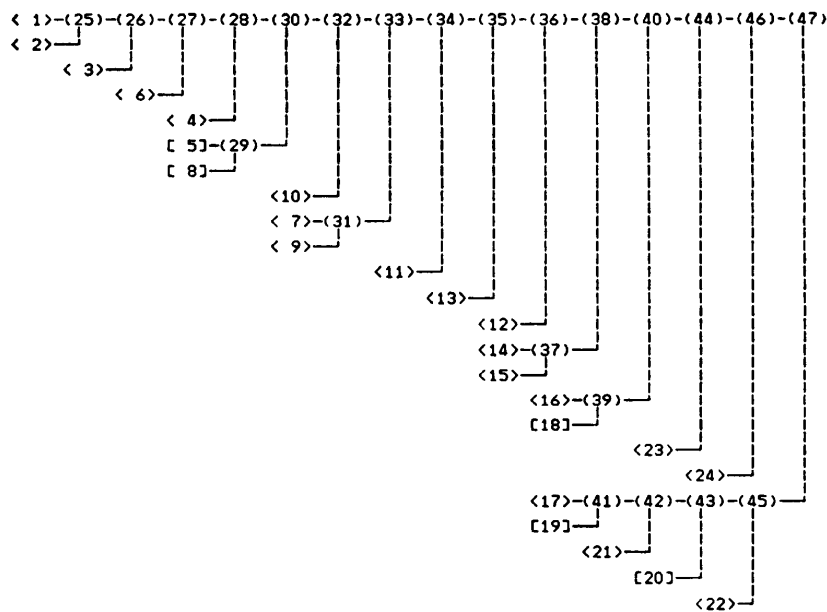
## 4. 実験結果

### 4.1 試料画像とその s-tree

国土地理院作成“国土数値情報”<sup>11),12)</sup>の一つである標高データファイル KS110 より 20 万分の 1 地形図 4 枚分に相当する  $640 \times 640$  点の領域 3 種をとり出し、特殊標高（海域、内水域、海面下の標高）を修正したのち、 $3 \times 3$  局所領域で一様重み平滑化を施したものをおおの原画像とする（以下での原画像とはすべて



(a) 入力画像の等高線図（等高線間隔 20m）



< >, [ ], ( ) 内数字は節点ラベル  
 < >: 頂点節点  
 [ ]: 仮頂点節点  
 ( ): 分岐節点

(b) 等高面トリー

入力画像：5 万分の 1 地形図「御油」付近。  
 $100 \times 100$  画素  
 一様重み平滑化 ( $9 \times 9$  局所領域)

図 3 等高面トリーの抽出例

Fig. 3 Example of an extracted surface tree.

表 1 原画像と s-tree のデータ量

Table 1 Size of original images and their surface trees.

| 原 画 像                     | s-tree*                            |
|---------------------------|------------------------------------|
| 4 バイト/画素                  | 16 バイト/節点                          |
| 全面素数 (640×640)<br>409,600 | 節点数 約 4,600 個<br>(内頂点節点数約 3,100 個) |
| 計 1,600 kB                | 計約 72 kB                           |

\* 3 枚の原画像についての平均値.

この画像を表す).

まず, これらの原画像より s-tree (検索のときは原トリーとなる) を抽出した. 表 1 にその概要を示す.

s-tree の抽出法は方法 1 を用い, しきい値間隔は  $\Delta T=10$  である. これより s-tree は, 原画像の約 1/22 (4.5%) のデータ量となり, これをデータベースに付加することは記憶容量上はほとんど問題がないことが知られる. なお, s-tree の抽出時間は FACOM M-200 で約 2,300 秒程度 (方法 1 の場合) である. 図 3 に s-tree の抽出例を示す.

#### 4.2 s-tree の抽出方法の比較

2.2.2 項で述べた分岐節点の抽出の二つの方法の比較を表 2 に示す.

方法 2 は方法 1 の約 1/76 の処理時間であり, かなり速い. 方法 1 の処理時間は画像中の最高標高と最も低い分岐標高との差, およびしきい値間隔に依存するので画像によりかなりの差が生じる. 方法 2 の処理時間は, その大部分が領域分割に費やされ, 画像内の頂点数に若干依存するが, 画像による差はあまりない. s-tree 抽出に必要な記憶領域は, 方法 1 では画像 3 枚分で一定であるが, 方法 2 では隣接表の大きさが頂点

表 2 s-tree 抽出方法の比較

Table 2 Comparison of two methods for extracting the s-tree.

|            | 方 法 1                            | 方 法 2                              |
|------------|----------------------------------|------------------------------------|
| 処 理 時 間*   | 106.7 秒**                        | 1.4 秒                              |
| 記 憶 領 域    | 入力画像 3 枚分<br>作業領域                | 入力画像 2 枚分<br>隣接表<br>作業領域           |
| 標高値のしきい値間隔 | 処理過程の最初に設定. 間隔を変えると原則として全処理をやり直す | 画像処理の過程はしきい値に無関係. 間隔変更は隣接表処理の段階で可能 |

\* 大きさ 100×100 画素の画像 25 枚の平均値 (FACOM M-200).

\*\* しきい値間隔  $\Delta T=1$ .

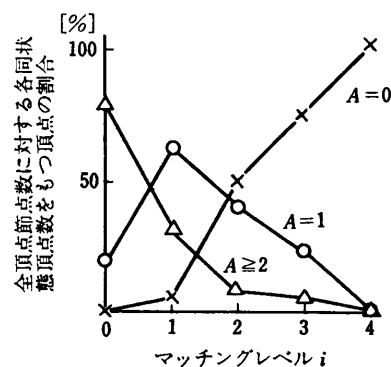


図 4 マッチングレベルに対する同状態頂点数

Fig. 4 Relationship between numbers of the same status peaks and matching level.

数の 2 乗に比例する. 方法 1 ではしきい値間隔を大きくとれば s-tree は近似になるが処理時間は短縮できる. 方法 2 でも若干の短縮はできるが効果は少なく, むしろ, 高精度の s-tree を高速で求めるに適する.

したがって, 隣接表が原画像の大きさより小さい範囲では方法 2, その他のときは方法 1 を使うのがよい. 今回の実験によれば, 方法 1 で  $\Delta T=10$  程度までならば失われる情報量は少ないようである.

#### 4.3 マッチングレベルと同状態頂点数

3.2 節で示したマッチングレベル  $i$  に対する同状態頂点数の変化を図 4 に示す.

図 4 は, 検索トリーの全頂点節点数に対する, 同状態頂点数  $A$  の各値をもつ頂点節点の割合を示し, 前記の 3 枚の原画像より大きさ 80×80 画素の検索画像を各 25 枚 (計 75 枚) 取り出したときの平均値である. これより, マッチングレベルは,  $A=1$  の割合が比較的大きいレベル 1 またはレベル 2 で,  $A=1$  となる頂点節点を用いる方法が適当と思われ, 以下の検索実験

表 3 検索結果の一例

Table 3 Experimental results of retrieval.

| 検索画像の大きさ* | 正検索率** (%) | 検索時間 [秒] |        | 誤検索率† (%) |
|-----------|------------|----------|--------|-----------|
|           |            | 抽出方法 1 ‡ | 抽出方法 2 |           |
| 20×20     | 22.7       | 0.21     | 0.26   | 1.3       |
| 40×40     | 76.0       | 1.53     | 0.38   | 8.0       |
| 60×60     | 92.0       | 2.95     | 0.59   | 10.7      |
| 80×80     | 94.7       | 8.10     | 1.00   | 10.7      |
| 100×100   | 97.3       | 13.09    | 1.79   | 12.0      |

\* 検索画像は各大きさで 25 枚, 計 75 枚.

\*\* 正しい位置に検出されたものの割合.

† 誤った原画像中に対応位置を検出したものの割合.

‡ しきい値間隔  $\Delta T=10$ , 計算機は FACOM M-200.

でもこの方法を用いた。このようにマッチングレベルを上げることが必ずしも正確なマッチングにはつながらないのは、検索画像が原画像に比べて小さいためその s-tree の構造（分岐の起こる標高）が原画像とは異なってくるためである。

#### 4.4 検索画像

各原画像より、種々の大きさの部分画像を切り出し、これを検索画像としたときの検索率と検索時間を表3に示す。ここで検索率とは全入力画像に対して正しい位置が検出されたものの割合である。また検索時間は、検索トリーの抽出、マッチング、位置決めに要した時間の合計である。

この結果によれば検索画像は 40×40 画素以上でないといふ検索はできない。これは検索画像が小さいためにそのなかの頂点数が少なく、s-tree が十分な情報をもたないためである。すなわち、この大きさでは大局的情報は把握されないことを意味している。60×60 画素以上の入力の場合の誤りは、位置決め過程で誤って排除されたり、A=1 の頂点がなかったりするものであり、原画像中の誤った位置で検出されたものはなかった。一方、原画像以外の地形図中に誤って位置を検出してしまったものの割合も表3に示してある。また検索時間の大部分は s-tree の抽出に要する時間である。なお、原トリーはディスクファイルにおかれ、その読み込みには 0.9 秒程度必要である。これらの結果から、大局的構造のマッチングによる大まかな検索という本方法の目的は十分達せられると思われる。

#### 5. む す び

画像の基本的な構造情報（スケッチ）として s-tree の利用を試み、それが原画像の数%のデータ量であるにもかかわらず有効な大局的情報をもつことを、画像データベースの検索への応用を通して確かめた。s-tree 自体は任意の濃淡画像に適用できるが、本文で用いた地形図への適用は最も自然なものであろう。今回の実験は初めての試みであり、本文で論じなかった多くの興味ある問題が提起される。そのおもなものについて以下で簡単に述べておこう。

(1) 今回の検索実験では、入力検索画像と原画像の該当部分がもし存在すれば厳密に一致する。このような状況では、直接的パターンマッチングでももちろん検索は可能である。ちなみに、今回の資料で入力検索画像の左上の 3×3 画素のすべてが一致する位置で検出を試みたところ、75 個の入力中 69 個が正しく検

出され、マッチングに要した時間は、平均 0.2 秒程度であった。しかし、このときは原画像に直接アクセスするため、原画像をディスクから読み込むのに 6 秒程度必要とする。

(2) s-tree も検索法もともに標高値そのものに依存するため、雑音に対して敏感である。実際、加法的ガウス雑音を付加した画像による予備の実験では、入力画像の平滑化、および標高値のマッチングにおける許容区間の導入は必ずしも決め手にはならない。したがって、ランダム雑音を含む画像への本方法の適用性には限界があり、よい補助手段の開発が望まれる。

(3) 限られた記憶容量で十分大きい画像を扱えるためには、s-tree の分割作成と統合の効率よい方法を開発する必要がある（原理的には本文の方法の若干の修正でよい）。

謝辞 終りに、ご討論いただいた名古屋大学横井茂樹助教授および豊橋技術科学大学の研究室の皆様へ感謝する。なお、本文の実験には名古屋大学大型計算機センターと画像処理ライブラリ SLIP を利用した。また、本研究の一部は文部省科研費 (No. 57380004) による。

#### 参 考 文 献

- 1) 柴田, 鳥脇, 福村: 胸部 X 線写真データベースの開発と素描の応用, 信学研資, PRL 78-70 (1979).
- 2) 長谷川, 福村, 鳥脇: 胸部 X 線写真データベースのためのスケッチの応用, 信学論, Vol. J 65-D, No. 9, pp. 1121-1128 (1982).
- 3) Freeman, H. and Morse, S. P.: On Searching a Contour Map for a Given Terrain Elevation Profile, *J. Franklin Inst.* Vol. 284, No. 1, pp. 1-25 (1976).
- 4) Mark, D. M.: Topological Randomness of Geographic Surface, Ph. D. Thesis, Simon Fraser Univ. (1977).
- 5) 安居院, 松原, 中嶋: 閉曲線群の記述と構造解析, 信学研資, PRL 80-48 (1980).
- 6) 安居院, 松原, 中嶋: 閉曲線群の計算機処理とその応用, 信学研資, PRL 80-49 (1980).
- 7) 安居院, 伊藤, 中嶋: 山岳地図情報処理に関する研究, 信学研資, PRL 81-37 (1981).
- 8) 鳥脇, 福村: 標本化された濃淡画像の構造情報抽出の一方, 信学研資, PRL 76-15 (1976).
- 9) 李, 長谷川, 福村, 鳥脇: 地形図情報の構造化の一実験, 信学研資, PRL 80-102 (1981).
- 10) Rosenfeld, A. and Kak, A. C.: *Digital Picture Processing* (2nd ed.), Vol. 2, Academic Press, New York (1982).

- 11) 矢口：国土数値情報，電気学会誌，Vol. 98, No. 11, pp. 1032-1035 (1978). PRL 82-68 (1983).
  - 12) 野々村：国土数値情報整備事業について，情報処理学会データベース管理システム研資，20-1 (1980). 14) Nagy, G. and Wagle, S.: Geographic Data Processing, *Comput. Surv.* Vol. 11, No. 2, pp. 139-181 (1979).
  - 13) 杉山，鳥脇：スケッチをキーとする地形図データの検索への *surface tree* の応用，信学研資，  
(昭和58年4月27日受付)  
(昭和58年7月19日採録)
-