

## 連に基づく高速2回走査ラベル付けアルゴリズム

## A High-Speed Run-Based Two-Scan Labeling Algorithm

何立風<sup>†1</sup>, 巢宇燕<sup>†2</sup>, 鈴木賢治<sup>†3</sup>,  
中村剛士<sup>†4</sup>, 伊藤英則<sup>†4</sup>

Lifeng He<sup>†1</sup>, Yuyan Chao<sup>†2</sup>, Kenji Suzuki<sup>†3</sup>, Tsuyosi Nakamura<sup>†4</sup> and Hidenori Itoh<sup>†4</sup>

**Abstract** This paper presents a run-based two-scan algorithm for labeling connected components in a binary image. Unlike conventional two-scan labeling algorithms which resolve label equivalences by processing pixels one by one, our algorithm resolves label equivalences by processing runs one by one. At any time, all provisional labels that are assigned to a connected component are combined in a set, and the smallest label is used as their representative label. The corresponding relation of a provisional label and its representative label are recorded in a table. Whenever different components are found to be connected, all provisional label sets related to these connected components are merged together, and the smallest provisional label is considered to be the representative label. When the first scan is finished, all provisional labels that were assigned to each connected component in the given image will have a unique representative label. During the second scan, we need only to replace each provisional label with its representative label. Experimental results on various types of images demonstrated that our algorithm outperforms all conventional labeling algorithms.

キーワード：ラベル付け, 連, 2回走査, 画像認識

## 1. ま え が き

パターン認識やコンピュータビジョンにおいて、2値画像中の対象物（連結成分）に別々のラベルを付けるラベル付けは、最も基本的かつ重要な処理の一つであり、コンピュータに対象物を認識させる上で欠くことができない<sup>1)</sup>。ラベル付けは、その対象物の連結性解決の複雑性により、エッジ検出やノイズ平滑化などの他の基本画像処理手法と比べて演算時間が長いいため、一般的にこれらを組合せて構成さ

れるパターン認識システムにおいて、そのオンライン実時間実現を妨げる決定的な要因となっていた。

これまでに多くのラベル付けアルゴリズムが提案されており、それらはラスタ走査型と輪郭追跡型に大別できる。ラスタ走査アルゴリズムには、マルチ走査アルゴリズム<sup>3)4)</sup>、2回走査アルゴリズム<sup>2)5)~9)</sup>、ハイブリッドアルゴリズム（最大4回走査<sup>13)</sup>）がある。これらのアルゴリズムでは、各画素をラスタ走査順に処理し、暫定ラベル間の連結性の検出および解析を行うことによりラベル付けを行う。これらの中で、ハイブリッドアルゴリズム<sup>13)</sup>が最速のものである。輪郭追跡アルゴリズム<sup>12)</sup>は対象物の輪郭を追跡することによりラベル付けを行うが、画像の走査が不規則となるため、並列処理やハードウェア実現に向かない。

本稿では、画素ごとに連結性分析する従来の2回走査ラベル付けアルゴリズムと異なり、連に基づく新しい2回走査ラベル付けアルゴリズムを提案する。様々な性質の画像を用いた比較評価実験により、本アルゴリズムは既存のすべてのラベル付けアルゴリズムより高速であることを示す。

## 2. 提案アルゴリズムの概要

対象物画素の値を  $V_O$ 、背景画素の値を  $V_B$  としたサイズ  $N \times N$  の2値画像において  $(x, y)$  にある画素の値を  $p(y \times N + x)$  で表す。ただし、 $V_O, V_B$  ( $V_O < V_B$ ) は任意の暫定ラベルよりも大きな値、画像の回り（最縁）の画

2008年2月15日受付, 2008年6月19日最終受付, 2008年7月8日採録

†1 愛知県立大学 大学院 情報科学研究科

(〒480-1198 愛知県愛知郡長久手町)

†2 名古屋産業大学 大学院 環境情報マネジメント研究科

(〒488-8711 愛知県尾張旭市新居町 3255-5)

†3 シカゴ大学 大学院 放射線医学研究科

(アメリカイリノイ州シカゴ市)

†4 名古屋工業大学 大学院 工学研究科

(〒466-8555 愛知県名古屋市長和区御器所町)

†1 Graduate School of Information Science and Technology, Aichi Prefectural University

(Nagakute-cho, Aichi-Gun, Aichi 480-1198)

†2 Graduate School of Environment Management, Nagoya Sangyo University

(3255-5, Arai-cho, Owariasahi, Aichi 488-8711)

†3 Department of Radiology, The University of Chicago

(Chicago, IL, USA)

†4 Graduate School of Information Science, Nagoya Institute of Technology

(Gokiso-cho, Showa-ku, Nagoya 466-8555)

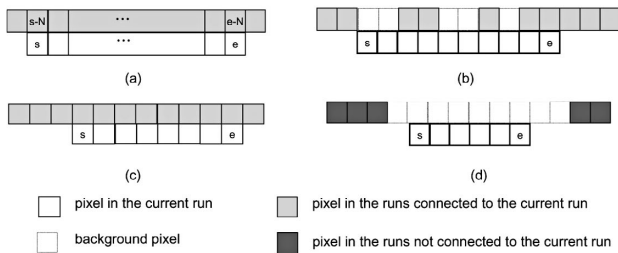


図 1 注目連の 8 連結性

Eight-connected connectivity for the current run. (a) Eight-connected pixel region for the current run; (b) and (c) samples of the current run that are eight-connected with some runs; (d) a sample of the current run that are not eight-connected with any runs.

素は背景画素であることとする。

画像中の同じ行にある  $p(s)$  から  $p(e)$  ( $s \leq e$ ) までの連続した対象物画素のブロックを連と呼び、 $r(s, e)$  で表す。処理中の連 (注目連と呼ぶ) が  $r(s, e)$  であるとき、図 1 に示すように、これまでの走査により発見した連のうち、 $a \leq e - N + 1$  または  $b \geq s - N - 1$  のすべての連  $r(a, b)$  が注目連  $r(s, e)$  と 8 連結する。

本アルゴリズムでは、走査中の各時点までに発見した同一対象物に属する連の暫定ラベルをその対象物に対応する同等ラベル集合にまとめ、それらの暫定ラベル中の、最小ラベルを代表ラベルとする。便宜上、暫定ラベル  $l$  の代表ラベルは  $r$  であることをテーブル  $R[l] = r$  で表し、また、代表ラベル  $i$  をもつ同等ラベル集合を  $S(i)$  で表す。

第一走査では、 $i = 0$  から画素  $p(i)$  をラスト走査順に処理する。新しい連 (注目連) を発見したときは、そのデータを記録し、注目連に 8 連結している連が存在するかどうかをチェックする。このような連が存在しない場合、注目連がこれまで発見した対象物のどれとも連結してないことを意味する。そのため、注目連に属するすべての画素に新しいラベル  $l$  を付ける。ここでは、 $R[l] = l$ 、 $S(l) = \{l\}$  となる。

一方、注目連に 8 連結している連が存在する場合、これらの連が注目連と同一対象物に属することがわかる。このとき、これらの連に対応する同等ラベル集合  $S(r_1), \dots, S(r_n)$  にあるすべての暫定ラベルは同等ラベルであるため、同等ラベル集合  $S(r)$  に合併する。ただし、 $r$  は  $r_1, \dots, r_n$  の最小のものであり、 $S(r)$  にある任意の暫定ラベル  $l$  において、 $R[l] = r$  である。また、注目連にあるすべての画素にラベル  $r$  を付ける。

図 2(a) に、ある対象物において注目連を処理する前のラベル付けの状況を示し、注目連を処理したあとの状況を図 2(b) に示す。

本アルゴリズムでは、同等ラベル集合の間に連結関係を見出すたびに、それらの同等ラベル集合を合併するとともに代表ラベルテーブルを更新することから、第一走査後、同一対象物に属するすべての暫定ラベルが同一同等ラベル集合にまとめられ、同一代表ラベルをもつことになる。そ

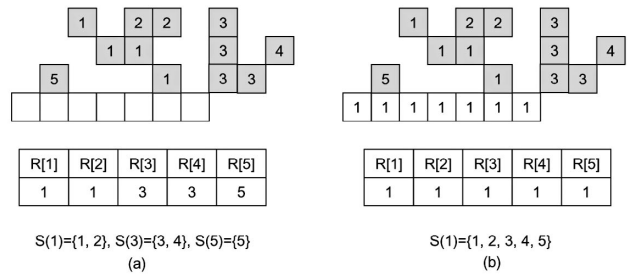


図 2 連結性の解決

Operations for label equivalence resolving: (a) before processing the current run; (b) after processing the current run.

のため、第二走査では、各対象物画素に付与された暫定ラベルをその暫定ラベルの代表ラベルに書き換えれば、各対象物にユニークなラベルを付けることができる。

### 3. アルゴリズムの実装

本アルゴリズムにおいては、同等ラベル集合の合併が主な演算であるが、同等ラベル集合を連結リストで実現すれば、その演算は定数回で終了する。便宜上、連結リストで実現した同等ラベル集合のことを同等ラベルリストと表現することにする。

任意の暫定ラベルは、一つの同等ラベルリストにしか属さないことから、同等ラベルリストは、 $V_0$  サイズの二つの 1 次元配列  $next[]$  と  $tail[]$  で実現できる。 $next[l] = j$  は、ある暫定ラベル  $l$  の次のラベル  $j$  を示す配列である。なお、 $next[l] = -1$  は暫定ラベル  $l$  の次のラベルは存在しない、つまり、 $l$  はその同等ラベルリストの末尾要素であることを示す。一方、 $tail[u] = v$  は、同等ラベルリスト  $S(u)$  の末尾要素が  $v$  であることを示す配列である。

また、暫定ラベルとその代表ラベルの関係を表す代表ラベルテーブルは 1 次元配列  $rtable[]$  により簡単に実現できる。すなわち、暫定ラベル  $l$  の代表ラベルを  $r$  にするときは、 $rtable[l] = r$  とすればよい。また、暫定ラベル  $l$  の代表ラベル  $r$  は  $r = rtable[l]$  で調べることができる。

$rtable[]$ 、 $next[]$  と  $tail[]$  を用いれば、同等ラベルリストの新設と合併は簡単に実現できる。同等ラベルリスト  $S(l) = \{l\}$  を新設するためには、次の操作を行えばよい。

$$rtable[l] = l, next[l] = -1, tail[l] = l.$$

一方、同等ラベルリスト  $S(v)$  を  $S(u)$  に合併するときには、 $S(v)$  の先頭要素  $v$  を  $S(u)$  の末尾要素  $tail[u]$  の次の要素につなげ、 $S(u)$  の末尾要素を  $S(v)$  の末尾要素に変更し、 $S(v)$  にあるすべての暫定ラベルの代表ラベルを  $u$  に書き換えればよい。これは次の操作により実現できる。

```

next[tail[u]] = v;
tail[u] = tail[v];
i = v;
while (i ≠ -1) do
    rtable[i] = u;
    i = next[i];
end while

```

また、 $rtable[V_B] = V_B$  とあらかじめセットしておけば、第二走査における各対象物画素  $p(i)$  に付与された暫定ラベル (つまり、 $p(i)$ ) をその暫定ラベルの代表ラベル ( $rtable[p(i)]$ ) に書き換える操作は次のように簡単に実現できる。

$$p(i) = rtable[p(i)]$$

本アルゴリズムでは、ラスタ走査順に処理を行うため、先に発見した連の連結性解析が先に行われる。そのため、二つのキュー  $s\_queue$  と  $e\_queue$  を用いて連  $r(s, e)$  のデータ  $s$  と  $e$  をそれぞれ記録することができる。また、注目連  $r(s, e)$  を処理したあと、 $n \leq e - N$  が成り立つようなすべての連  $r(m, n)$  がその後の連結性解析に用いられることがないので、それら連のデータを廃棄することができる。言い換えれば、 $r(s, e)$  を処理したあとは、 $p(e - N + 1)$  から  $p(e)$  までの画素を含む連のデータだけを保存すればよい。そのため、キュー  $s\_queue$  と  $e\_queue$  をそれぞれ  $(N/2 + 1)$  サイズの循環リストで実現できる。

#### 4. 比較評価実験

提案アルゴリズムの特性および性能を評価するため、ワークステーション (Intel Pentium D 930 3.0GHz + 3.0GHz CPUs, 2GB Memory, Mandriva Linux OS) 上での CPU 実行時間を用いた比較評価実験を行った。なお、実行時間はシングルコアを用いた 5,000 回の実行の平均値である。

実験には次の 5 種類の画像を用いた。(1) ノイズ画像: 六つのサイズ ( $32 \times 32$ ,  $64 \times 64$ ,  $128 \times 128$ ,  $256 \times 256$ ,  $400 \times 400$  と  $512 \times 512$  画素) の画像に、1,000 階調までの白色一様ノイズを加え、閾値を 0 から 1,000 まで 25 毎に変化させて 2 値化した合計 246 種類の 2 値画像 (各サイズ毎には 41 種類)。このような画像は自然画像に比べ、複雑な幾何学形状と複雑なラベル連結性を持つため、ラベル付けアルゴリズムをより厳格に評価するのに適している。(2) 自然画像: SIDBA (Standard Image Data Base) および USC (University of Southern California) の標準画像データベースから風景画像、航空画像、人物画像、静物画像、指紋画像、スナップ画像およびテキスト画像を含む 50 種類の自然画像を Otsu の閾値選択法<sup>10)</sup>により 2 値化した画像。(3) 医用画像: シカゴ大学医用画像データベースから 25 種類の医用画像を Otsu 法<sup>10)</sup>により 2 値化した画像。(4) Columbia-Utrecht Reflectance and Texture Database\* から 7 種の

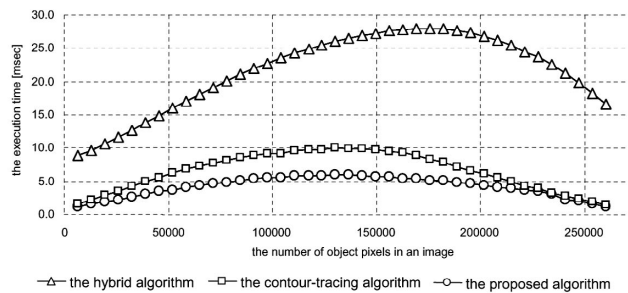


図 3 画像中の対象物画素数に対する実行時間の特性  
Execution time versus the number of object pixels in an image.

テクスチャ画像を Otsu 法<sup>10)</sup>により 2 値化した画像。(4) テストパターン画像: ラベル付けアルゴリズムを評価するために作成した、階段状、渦状、のこぎり状、チェッカーボード状、蜂巣状パターンを含むテスト用の人工画像<sup>13)</sup>。

ハイブリッドアルゴリズム<sup>13)</sup>が従来のラスタ走査型中最速であること<sup>13)</sup>、輪郭追跡アルゴリズム<sup>12)</sup>が最新のアルゴリズムであることから、この二つのアルゴリズムを比較対象とした。なお、実験に用いたハイブリッドアルゴリズムのプログラムは、著者から提供されたものであり、輪郭追跡アルゴリズムのプログラムは、著者のホームページ\*\*よりダウンロードしたものである。

$512 \times 512$  サイズのノイズ画像を用いて画像中の対象物画素数に対する各アルゴリズムの実行時間特性を図 3 に示す。すべての画像において、本アルゴリズムはハイブリッドアルゴリズムと輪郭追跡アルゴリズムより速いことが確認できる。

図 4 に、ノイズ画像における画像サイズに対する各アルゴリズムの実行時間特性を示す。三つのアルゴリズムとも線形特性をもつことが確認できる。画像サイズに対する線形性は、パターン認識アルゴリズムにとって望ましい特性である。なお、これら三つ以外のラベル付けアルゴリズムは、線形特性を持っていない<sup>13)</sup>。また、本アルゴリズムの最大実行時間が他の二つの手法の平均実行時間よりも短いことも確認できる。

最後に、自然画像、医用画像、テクスチャ画像とテストパターン画像を用いて代表的なマルチ走査ラベル付けの Lumia アルゴリズム<sup>5)</sup>、2 回走査ラベル付けの Rosenfeld アルゴリズム<sup>11)</sup>と白井アルゴリズム<sup>6)</sup>とも比較を行った。実験結果を表 1 に示す。すべての種類の画像において、本アルゴリズムが最速であることがわかる。

また、実験に用いたすべての画像において、本アルゴリズムが最速であることも確認した。

#### 5. 従来の 2 回走査手法との比較

連の連結性に基づいた提案手法は従来の画素の連結性に基づいた 2 回走査ラベル付け手法との違いとしては、次の 3 点が挙げられる。

\* <http://www1.cs.columbia.edu/CAVE/software/curet/index.php>

\*\* <http://dar.iis.sinica.edu.tw/Download>

表 1 自然画像, 医用画像, テクスチャ画像, 人工画像における実験結果 (ミリ秒)  
Comparison of various execution times [msec] with natural images, medical images, textural images, and special-shape artificial images.

Image type		Rosenfeld	Lumia	Shirai	Hybrid	Contour	Ours
natural	max.	2185.2	2045.9	52.4	18.4	3.8	2.9
	mean	524.6	173.3	40.0	13.7	2.4	1.8
	min.	22.7	11.7	33.9	9.6	1.2	1.0
medical	max.	650.2	807.7	41.7	14.9	2.6	1.8
	mean	378.9	192.1	38.8	13.4	1.9	1.5
	min.	75.5	15.3	37.2	11.4	1.5	1.2
textural	max.	1068.7	72.7	42.8	28.5	3.7	2.5
	mean	296.7	35.2	39.3	27.4	2.7	1.8
	Min.	13.8	15.6	38.5	26.4	1.5	1.1
artificial	max.	1061.3	748.9	39.3	15.8	7.4	2.5
	mean	316.8	164.9	22.5	7.9	3.9	1.5
	Min.	2.0	2.5	0.96	2.2	1.1	0.8

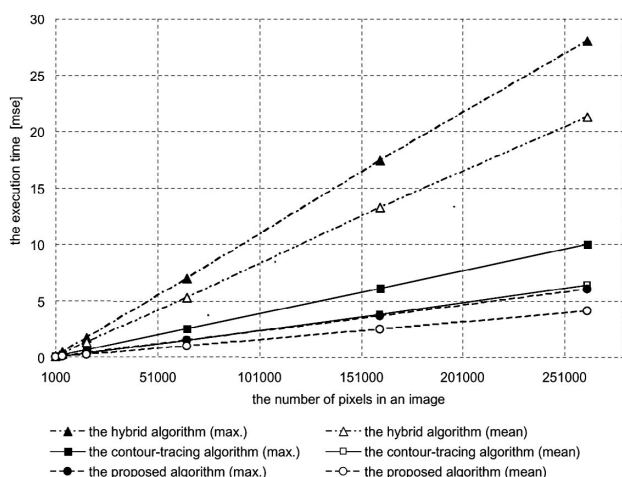


図 4 画像サイズに対する実行時間の線形性  
Linearity of the execution time versus image size.

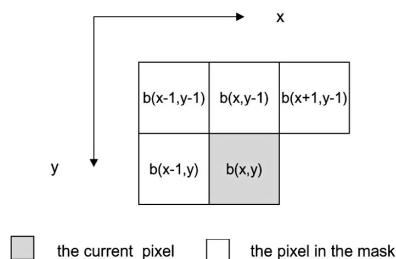


図 5 8 連結用走査マスク  
Mask for the eight-connected connectivity.

### 5.1 暫定ラベルの付け方

従来の 2 回走査ラベル付けアルゴリズムでは, 画素ごとに連結性分析を行う. 図 5 に示すマスクを用いてラスタ走査の順序で走査しながら, 画像中の注目対象物画素に対応するマスク中の最小暫定ラベルを暫定ラベルとして付ける. そのとき, マスクに最小暫定ラベルを計算する必要がある.

これに対して, 提案手法では, 画素のつながりから成る連 (ランとも言う) に基づきラベル付けを行う. 注目連の上すべての対象物画素に同一暫定ラベルを付けていくので, 2 番目以降の画素において最小暫定ラベルを計算する必要がない. 対象物画素数が  $P$ , 連数が  $L$  である画像において, 従来の手法では, 暫定ラベル付けに  $3 \times P$  回比較が必要であるのに対して, 提案手法では,  $3 \times L$  回比較で済む.

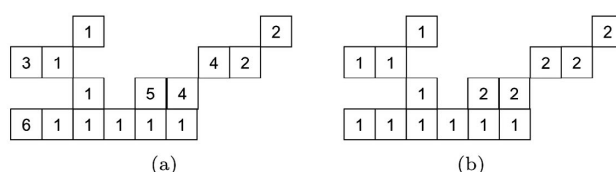


図 6 暫定ラベル付けの例  
Provisional labels after the first-scan: (a) assigned by conventional raster-scan algorithms; (b) assigned by our algorithm.

$P \geq L$  が成立するから, 提案手法が従来の手法より効率である. 特に  $P \gg L$  のとき, 提案手法が従来の手法より高速である.

また, 従来の手法では, 注目対象物画素に対応するマスクにはほかの対象物画素が存在しないとき, 注目対象物画素に新たな暫定ラベルを付ける. これに対して, 提案手法では, 注目連に 8 連結している連が存在しないときだけ, 注目連に属するすべての画素に新たな暫定ラベルを付ける. 連の画素が複数である場合, 連の 8 連結領域が, 従来の一つの画素の 8 連結領域より広いから, 同じ画像において提案手法で付けられた暫定ラベルの数は従来の手法で付けられた暫定ラベルの数より少なくなる.

図 6 は従来の手法と提案手法による暫定ラベル付けの状況の例である. 従来の手法は六つの暫定ラベルを付ける (図 6(a)) のに対して, 連に基づき処理をする提案手法は二つだけである (図 6(b)). このように, 連を単位として処理することにより, 暫定ラベルの数を大幅に削減できる.

暫定ラベルの数の減少は, その分の同等ラベルリストの新設に関連する操作を減少させるだけではなく, 同等ラベルリストの合併 (暫定ラベルの連結性の解析) に関連する操作も減少させるため, 本アルゴリズムの効率なラベル付けを可能としている.

512×512 サイズのノイズ画像を用いて同じ画像における従来の手法で付けられた暫定ラベルの数と提案手法で付けられた暫定ラベルの数を図 7 に示す. すべての画像において, 画像に付けられた暫定ラベル数は, 提案手法が従来の手法より少ないことがわかる.

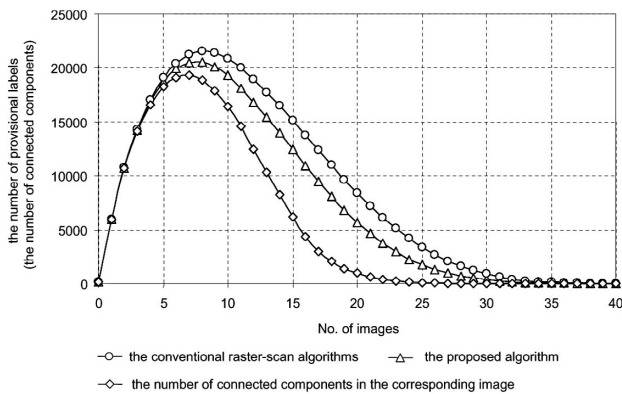


図 7 暫定ラベルの数の比較

Number of provisional labels assigned by different raster-scan algorithms.

## 5.2 連結性の解析のタイミング

従来の手法では、第一走査で発見した同等ラベル関係を解析せず記録しておく。第一走査が終了しない限り、同等ラベルの解析ができない。第一走査が終了後、同等ラベルの解析を行う。そのときも同様、同等ラベルの解析が完了しない限り、第二走査が開始できない。これに対して、提案手法では、第一走査において同等ラベル集合の間に連結関係を発見するたびに、それらの同等集合を合併することによって、同等ラベルの連結関係を解決する。第一走査が終了したとき、同等ラベルの解析も同時に完了している。そのため、第二走査がすぐ開始できる。また、提案手法において、暫定ラベル付けと同等ラベルの連結関係解決が独立に行える操作である。

## 5.3 連結性の解析手法

画像のサイズを  $N \times N$ 、暫定ラベル数を  $L$  とする。そのとき、最悪の場合、 $L$  は  $N \times N$  と比例する。従来の手法では、第一走査で発見した同等ラベル関係を記録するとき、同等ラベル関係が重複して記録されるか、 $L \times L$  に比例する巨大サイズの配列が必要となる<sup>7)</sup>。これらの問題は同等ラベルの解析の高速化に妨げる要因となっている。

提案手法では、同等ラベルを同等ラベル集合に記録する；同等ラベル集合中の最小ラベルを代表ラベルとする；ラベル間の連結性を発見したたび、対応するラベル集合を合併するという連結性の記録と解析における新しいアイデアにより、重複する同等ラベル関係の記録を避けながら、3つの  $L$  サイズの配列で実現した連結リストを用いて連結関係の記録と解析を効率的かつ高速に行うことができる。なお、従来の手法では、本論文で提案したデータ構造で実現できない。

## 6. むすび

本論文では、連に基づく新しい2回走査ラベル付けアルゴリズムを提案した。様々な性質の画像を用いた比較評価実験により、提案したアルゴリズムの有効性を確認した。

今後の研究課題として、提案したアルゴリズムの計算量の

完全解析や3次元画像のラベル付けへの拡張が挙げられる。本研究の一部は、豊秋奨学会の研究助成による。

## 〔文 献〕

- 1) 鳥脇純一郎: 画像理解のためのデジタル画像処理 [II], 昭晃堂, pp. 45-47 (1988)
- 2) Rosenfeld, A. and Pfaltz, J. L.: Sequential operations in digital picture processing, *Journal of ACM*, **13**, 4, pp.471-494 (1966)
- 3) Haralick, R. M.: Some neighborhood operations, *Real Time/Parallel Computing Image Analysis*, New York, Plenum Press, pp. 11-35 (1981)
- 4) 橋詰明英, 鈴木隆一, 横内久猛, 堀内秀之, 山本真司: 赤血球自動識別アルゴリズムとその評価, *医用電子と生体工学*, **28**, 1, pp. 25-32 (1990)
- 5) Lumia, R., Shapiro, L. and Zungia, O.: A new connected components algorithm for virtual memory computers, *Computer Vision, Graphics, and Image Processing*, **22**, 2, pp. 287-300 (1983)
- 6) Shirai, Y.: Labeling connected regions, *Three-Dimensional Computer Vision*, Springer-Verlag, pp.86-89 (1987).
- 7) 後藤敏行, 太田善之, 吉田真澄, 白井良明: 連結領域の高速ラベル付けアルゴリズム, *信学論*, **J72-D-II**, 2, pp.247-255 (1989)
- 8) 奥山良幸, 小林芳樹, 武長寛, 浅田和佳, 藤原和紀: ラスタ走査形ラベリングの高速化手法, *信学論*, **J73-D-II**, 1, pp.36-45 (1990)
- 9) 直井聡: 文字の形状特徴を利用した可変ウィンドウサイズによる高速ラベリング手法, *信学論*, **J80-D-II**, 7, pp.1793-1801 (1997)
- 10) Otsu, N.: A threshold selection method from gray-level histograms, *IEEE Trans. Systems Man and Cybernetics*, **9**, pp.62-66 (1979)
- 11) Rosenfeld, A. and Kak, A. C.: *Digital Picture Processing*, volume 2, Academic Press, San Diego, CA, 2nd edition (1982)
- 12) Chang, F., Chen, C.J., and Lu, C.J.: A linear-time component-labeling algorithm using contour tracing technique, *Computer Vision and Image Understanding*, 93:206-220 (2004)
- 13) Suzuki, K., Horiba, I., and Sugie, N.: Linear-time connected-component labeling based on sequential local operations, *Computer Vision and Image Understanding*, 89:1-23 (2003)



**何立風** 1982年中国西北軽工業学院自動制御工学科卒。同年同大助手。1987年同大講師。1994年名古屋工業大学大学院博士前期課程修了。1997年同大後期課程修了。博士(工学)。現在、愛知県立大学大学院情報科学研究科准教授。画像処理、定理証明、知識データベースなどに興味を持つ。



**宇燕** 1984年中国西北軽工業学院機械工学科卒。1997年名古屋工業大学大学院人間情報文化工学研究科博士前期課程修了。2000年同大後期課程修了。2000年4月から2002年9月まで名古屋工業大学研究員。現在、名古屋産業大学准教授。博士(学術)。画像処理、定理証明および図面理解、CADなどに興味を持つ。



**鈴木賢治** 1991年名城大・理工・電気電子卒。1993年同大大学院修士課程修了。現在、シカゴ大学放射線医学研究科准教授。博士(工学)(名古屋大学)。コンピュータ支援診断、医用画像処理・認識及び機械学習の研究に従事。



**中村剛士** 1992年名古屋工業大学情報工科学科卒。1995年同大大学院博士前期課程修了。1998年同大後期課程修了。博士(工学)。現在、同大情報工科学科准教授。感性情報、ソフトウェア工学などに興味を持つ。



**伊藤英則** 1969年名古屋大学工学部電気電子工科学卒。1971年同大学院工学研究科修士課程修了。1974年同大博士課程修了。博士(工学)。同年NTT入社、横須賀研究所勤務。1985年(財)新世代コンピュータ技術開発機構出向。1989年より名古屋工業大学教授。感性情報、自動推論、マルチメディアなどに興味をもつ。