

高能率ラベリングプロセッサ のための一手法

藤原 秀人、 磯貝 文彦

三菱電機 応用機器研究所

本報告は、二値画像を幾つかの領域に分割するために行われるラベリング処理に関するものである。

ラベリングプロセッサの開発にあたっては、リアルタイム処理に向くラスタ走査型のアルゴリズムが一般に用いられているが、従来のラスタ走査型では仮ラベルが極端に多く発生するために能率が悪かった。本報告では、ラスタ走査を処理の基本とし、従来の方式に比べ仮ラベル数を大幅に低減できる能率の良いアルゴリズムを提案する。また、このアルゴリズムを用いたラベリングプロセッサについても述べる。ラベリングプロセッサに関しては現在検討段階であるが、比較的小規模な回路でリアルタイム処理を実現できる。

A method for high-performance labeling processor

Hideto FUJIWARA, Fumihiko ISOGAI

Products Development Laboratory
MITSUBISHI ELECTRIC CORP.

1-1, Tsukaguchi-honmachi 8-chome,
Amagasaki, Hyogo, 661 JAPAN

In this paper, we propose a high-performance labeling algorithm based on raster scanning. We also propose a labeling processor using this algorithm.

Conventional algorithms based on raster scanning have a problem which generates a lot of temporary labels.

Our algorithm can reduce number of temporary labels, and it consists of following two steps: (1)define run's label, (2)define temporary label using the run's label.

1. はじめに

近年の半導体技術の進歩によって、画像処理分野においても各種機能のハードウェア化による実時間処理が実現されつつある。二値画像内の対象物の個数を計数したり、各対象物の面積や重心位置などの特徴を抽出するうえでラベリング処理は必要不可欠な処理である。それだけにラベリングの高速化への要求が高い。

従来提案されてきたラベリングアルゴリズムは、(1)ラスタ走査型、(2)ランコード型(3)境界追跡型の3種類に大別できるがハードウェア化(高速化)に際してはそのリアルタイム処理の可能性や、回路規模、実現の容易性等からラスタ走査型が一般的に用いられている。

ラスタ走査型は1回の走査では対象物と1対1で対応したラベルを付けていくことができないうので、まず1回目の走査で仮ラベルを割り付けながら異なる値を持つ仮ラベル同士を連結を示す情報(統合情報)を保持し、次いでその統合情報を解析することによって対象物と1対1で対応した最終ラベルを得る。最後に、1回目の走査で得られた仮ラベル画像を再度走査し、仮ラベルを最終ラベルで置き換える。1回目の走査(仮ラベル付け処理)と2回目の走査(最終ラベル付け処理)に関しては比較的簡単にハードウェア化を行えるが、統合情報の解析(統合処理)に関してはハードウェア化が困難でありソフトウェア処理に頼っているのが現状である。

そして、従来のラスタ走査型アルゴリズムでは仮ラベル付け時に大量の統合情報が発生するのために統合処理の負荷が大きく、ラベリング処理全体で見ると処理性能が上がらないという問題点があった。

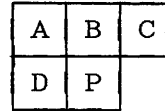
従って、ラベリング処理をより高速に実行するためには仮ラベル付け処理の段階で一塊の対象物に対する複数仮ラベルの割り当てを極力減らすこと、統合処理自体の高速化が重要である。

本稿では上記複数仮ラベルの割り当てを削減するための手法を提案する。本稿で提案する手法は、ラスタ走査によってラベルの連結性を判断してラン単位のラベルを求めた後に、そのラン内のラベル(ランラベル)をその構成画素に対して割り付けることによって従来の仮ラベル付けアルゴリズムに比して複数仮ラベルの割り当てを大幅に減らすことができるものである。筆者らはこれを『ラスタ走査型ランラベル方式』と名付けた。

2. 従来手法とその問題点

2.1 従来手法

従来手法は、水平3画素×垂直2画素のマスキ(第1図)を二値画像上でラスタ走査させ、注目点(第1図P)近傍(第1図A~D)のラベルを伝播させることによって画像のラベル付けを行うもので、次の3つのステップに分かれる。

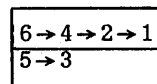
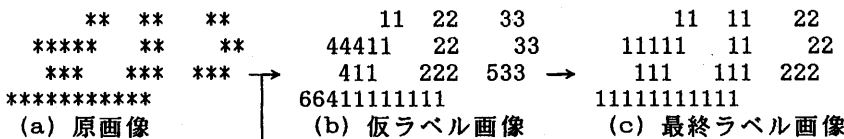


第1図 処理マスク

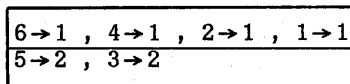
<1> 仮ラベル付(第2図(b))と

統合テーブルの作成
第1図に示す様なマスクを画像の左上からラスタ走査させ、注目点(P)が1のときに近傍画素(A~D)に付けている仮ラベルを注目点(P)に伝播させる。この時、近傍画素に全く仮ラベルが付けられていなければ(A~Dが全て0)注目点(P)に新しい仮ラベルを与える。

なお、近傍画素に2個の異なる仮ラベルが存在している場合があるが、この場合はどちらか一方を注目点(P)に伝播させると共に両仮ラベルが連結している(同一対象物内



解析



第2図 従来手法によるラベリング処理

に存在している)ことを示す統合テーブルを作成する。(第2図(b)では、仮ラベル4が1に、5が3に、6が4に、2が1にそれぞれ連結している)

<2> 統合テーブルの解析

<1>で得られた統合テーブルを解析し、一塊の対象物には1つのラベルが付けられるように仮ラベルと最終ラベルの間の変換テーブルを作成する。

<3> 最終ラベル付 (第2図(c))

仮ラベル画像を左上からラスト走査し、上記変換テーブルを用いて仮ラベルを最終ラベルに置き換える。

2.2 高速化における問題点

ラベリング処理の時間(最終ラベル画像を必要とする場合)は、上記3ステップの総和と異なる場合がある。<2>の統合テーブル解析では、なぜなら仮ラベルの付付け処理にアルタイル実行が比較的に容易にはハードウェア化が困難である。また、複数仮ラベル割り当ての大量発生によって仮ラベルの整理オーバーフローの問題が発生し、ラベルの整理に余分な処理が必要になる。

従って、ラベリング処理の高速化を図るうえで統合テーブル解析処理の高速化が必要不可欠になってくる。そのためのアプローチとしては2種類考えられる。

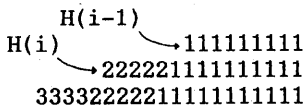
- (1) 解析処理自体の高速化
- (2) 複数仮ラベル割り当ての削減

本稿ではこの中の(2)を取り上げ、3章でその解決手法を示す。その解決手法を説明する前に、統合が発生する場合の画像状態について述べる。統合が発生する画像は次の2種類に分類できる。

(1) 階段状画像 (第3図)

ある行*i* (例えば、図中の2行目)のランについて見たとき、1つ上の行*i-1* (例えば、図中の1行目)の隣接するランの左端の水平座標(H(*i-1*))と*i*行のランの左端の水平座標(H(*i*))を比較し、その関係が以下の状態になっている場合を階段状画像と呼ぶ。

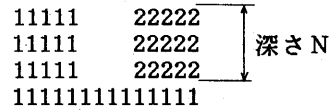
$$H(i) \leq H(i-1) - 2$$



(b) 仮ラベル画像
第3図 階段状画像

(2) U字型画像 (第4図)

U字型そのものの形をしており1つの対象物がその上端で2つ以上に分離しているが深さNのところまでそれらが結合しているような場合をU字型画像と呼ぶ。



(b) 仮ラベル画像
第4図 U字型画像

3. ラスタ走査型ランラベル方式

2章で述べた問題点を解決するための手法について述べる。

本方式は、ラスタ走査によってランの連結性を判断してラン単位のラベルを求めた後、その構成画素に対して割り付けることによって複数の仮ラベル割り当てを大幅に減らすことができる。

以下、本手法の動作について述べる。処理は第5図に示したような垂直2画素、水平3画素のマスクを画像の左上からラスト走査させて行う。

M11	M12	M13	仮ラベルマスク
M21	M22	M23	ランラベルマスク

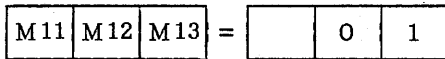
第5図 処理マスク

処理マスクの上段(M11からM13)を「仮ラベルマスク」と呼び、下段(M21からM23)を「ランラベルマスク」と呼ぶ。ランラベルマスクが走査している行を「注目行」、ランラベルマスクが走査しているランを「注目ラン」と呼ぶ。

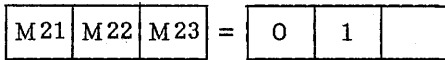
注目ランの水平始端座標を*j*_s、水平終端座標を*j*_eとしたときに1行上の水平座標*j*_{s-1}から*j*_{e+1}をこの注目ランに対する「隣接領域」と呼び、注目ランの1行上に存在し隣接領域にその1部でも重なっているようなランを「隣接ラン」と呼ぶ。また、本手法ではランの始端位置を認識する必要があるので、その始端条件を第6図に示す。

ランラベルマスクは注目ランに対するランラベルを決定すると共にそのランの終端でそのランラベルを記憶し、仮ラベルマスクは注目行の1行上に存在するラン(1行前のマスク走査で既にランラベルが決定している)に対してそのランの始端でランラベルを参照することによって仮ラベルを決定する。

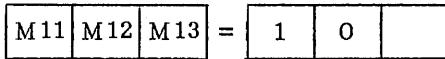
Pマスクの始端条件



Rマスクの始端条件



Rマスクの終端条件



第6図 始終端条件 (空白は0 or 1)

<アルゴリズム説明>

ランラベルマスクは注目ランの始端と終端を認識し、注目ランの終端に到達した時点でこの注目ランのランラベルをランラベルバッファに記憶する。ランラベルは、隣接ランが存在しなければ0であり、隣接ランが存在すればその隣接ランに対して仮ラベルマスクが決定した仮ラベルである。

このランラベルの決定は、ランラベルマスクが注目ランの始端で仮ラベルマスクをチェックしM11からM13が全て0ならその時点では隣接ランが存在するかどうか解らないのでランラベルを0に初期化し、M11からM13のながに1画素でも1があればその時点で隣接ランが存在していることが解るのでランラベルをその隣接ランの仮ラベルで初期化する。また、注目ランの始端以降その終端に至る途中で仮ラベルマスクが隣接ランの始端を認識し、その隣接ランに対する仮ラベルを決定したら注目ランのランラベルをその値で置き換えることによって実現できる。

仮ラベルマスクはランの始端を認識し、そのランをランラベルマスクが走査した時(1水平時間前)に記憶したランラベルを読みだし、そのランの構成画素に対する仮ラベルを決定し記憶する。決定した仮ラベルはM22=1になった時にラベルメモリに書込む。

仮ラベルは以下の条件で決定する。

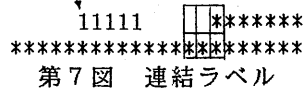
[仮ラベル決定条件]

- 1)ランラベル=0、連結ラベル=0
なら仮ラベル=新たなラベル
- 2)ランラベル≠0、連結ラベル=0
なら仮ラベル=ランラベル
- 3)ランラベル=0、連結ラベル≠0
なら仮ラベル=連結ラベル
- 4)ランラベル≠0、連結ラベル≠0、
ランラベル=連結ラベル
なら仮ラベル=連結ラベル
- 5)ランラベル≠0、連結ラベル≠0、
ランラベル≠連結ラベル
なら仮ラベル=連結ラベル

そして、条件5)の場合はランラベルと連結ラベルの統合処理を行う。

ここで、連結ラベルとは注目ラン1個に対して2個以上の隣接ランが存在した時に異なる仮ラベル同士で連結を検査するが、仮ラベルの直後にその仮ラベルで置き換えたとき(M13=0、M23=0)に0にクリアすることによって、実現できる。すなわち、ある注目ランに対して水平方向に2個目以降の隣接ランの各始端位置では1つ前の隣接ランの仮ラベルを記憶しており、注目ランに対して水平方向に最初の隣接ランであるなら連結ラベルは0である。(第7図)

連結ラベル



第7図 連結ラベル

第8図に本手法による仮ラベル付け動作を示す。

<複数仮ラベル割り当て削減の効果>

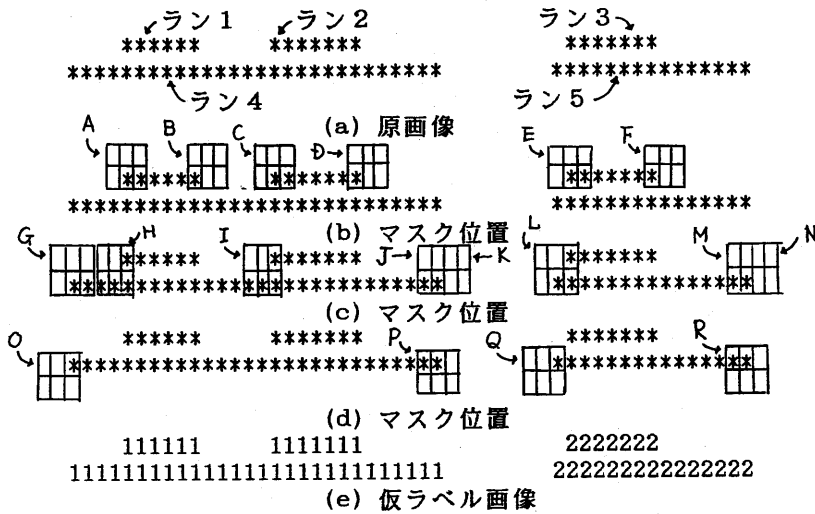
本方式を用いる効果に関して述べる。まず2章で述べた階段状画像に関しては、仮ラベル決定の前にランによるラベル付けを行っているので完全に複数割り当てを回避できる。

次にU字型画像に関しては、マスクの垂直サイズが2画素なのでU字の深さが2画素以上になると複数割り当てが発生するが、深さが1画素の場合には連結ラベルの導入と隣接するランが存在しないことを示すランラベル0の導入によって完全に回避できる。

4. ハードウェア化検討

本方式をハードウェア化した場合のプロック図を第9図に示す。

- ・マスク構成部
二値画像を入力として垂直2画素×水平3画素のマスクを構成する。
- ・ランラベル記憶制御部
ランラベルマスクの始端条件(M21=0、M22=1)の時、M11からM13を検査して全て0なら仮ラベルレジスタを0に初期化するためのリセット信号を出力する。また、ランラベルマスクの終端条件(M21=1、M22=0)の時、仮ラベルレジスタの内容をランラベル記憶部に書込むためのライト信号を出力する。
- ・ランラベル記憶部
Kビット幅(Kは仮ラベルが持つビット幅を表す)のFIFOによって構成され、各ランの出現順にランラベルを記憶する。



マスク位置	動作
A	ラン1のランラベルを0に初期化。
B	ラン1のランラベルとして0を記憶。
C	ラン2のランラベルを0に初期化。
D	ラン2のランラベルとして0を記憶。
E	ラン3のランラベルを0に初期化。
F	ラン3のランラベルとして0を記憶。
G	ラン4のランラベルを0に初期化。
H	ラン1の仮ラベルを決定。 ラン1のランラベル=0、連結ラベル=0 なので仮ラベル=新たなラベル(1) 同時にラン4のランラベルと連結ラベルを ラン1の仮ラベル(1)に変更。
I	ラン2の仮ラベルを決定。 ラン2のランラベル=0、連結ラベル=1 なので仮ラベル=連結ラベル(1) 同時にラン4のランラベルと連結ラベルを ラン2の仮ラベル(1)に変更。
J	連結ラベルを0にリセット。
K	ラン4のランラベルとして1を記憶。
L	ラン3の仮ラベルを決定。 ラン3のランラベル=0、連結ラベル=0 なので仮ラベル=新たなラベル(2) 同時にラン5のランラベルをラン3の仮ラベル(2)に初期化し、 連結ラベルをラン3の仮ラベル(2)に変更。
M	連結ラベルを0にリセット。
N	ラン5のランラベルとして2を記憶。
O	ラン4の仮ラベルを決定。 ラン4のランラベル=1、連結ラベル=0 なので仮ラベル=ランラベル(1) 同時に連結ラベルをラン4の仮ラベル(1)に変更。
P	連結ラベルを0にリセット。
Q	ラン5の仮ラベルを決定。 ラン5のランラベル=2、連結ラベル=0 なので仮ラベル=ランラベル(1) 同時に連結ラベルをラン5の仮ラベル(1)に変更。
R	連結ラベルを0にリセット。

(f) 各マスク位置での動作

第8図 ラスタ走査型ランラベル方式

- ・連結ラベル制御部
M13=M23=0の時、連結ラベルレジスタを0にする為のリセット信号を出力する。
- ・連結ラベルレジスタ
1個の仮ラベルを記憶するKビットのレジスタ。
- ・新ラベルカウンタ
Kビット幅のカウンタで、新たな仮ラベルが使われるたびにカウントアップして次の新ラベルを用意する。
- ・仮ラベル決定部
仮ラベルマスクの始端条件(M12=0、M13=1)の時、ランラベル記憶部からランラベルを読み出すためのリード信号を出力すると共に、上記仮ラベル決定条件から仮ラベルを決定しMPX1に対する選択信号を出力する。また、仮ラベルレジスタと連結ラベルレジスタに対するラッチ信号も出力する。
仮ラベルの決定は、ランラベル記憶部、連結ラベルレジスタの各出力の最上位1ビットをもって決定する。
- ・MPX1
仮ラベルを選択する為のkビット幅3入力1出力マルチプレクサ。
- ・仮ラベルレジスタ
仮ラベル決定後にそれをラッチするkビット幅のレジスタ。仮ラベルレジスタの内容は、同時にそのときランラベルマスクが走査しているランのランラベルでもある。上記ア

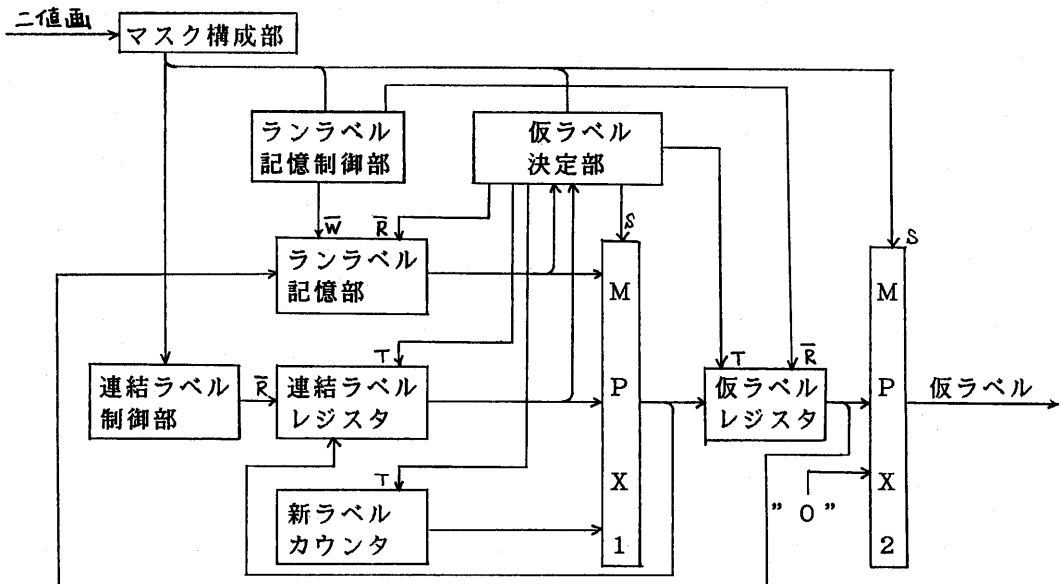
ルゴリズム説明では便宜上仮ラベルとランラベルを別のもので述べたが、ランラベルマスクがランの始端でランラベルを0に初期化する場合以外はランラベルと仮ラベルは常に同じ値であり、ランラベルが0に初期化される場合でもその条件から仮ラベルレジスタがラン上にならないことが解っており、仮ラベルレジスタが0にクリアされてもなんら問題はない。

- ・MPX2
M12の状態によって仮ラベルレジスタの内容をラベルメモリに送出するか、0を送出するかを選択する為のkビット幅2入力1出力マルチプレクサ。

<回路規模>

- 本回路は、
- kビット幅のレジスタ：2
- 3ビットシフトレジスタ：2
- 1ビット×水平画素数ラインバッファ：1
- kビット×TのFIFO：1
- kビット×2入力MPX：1
- kビット×3入力MPX：1
- kビットカウンタ：1
- AND、OR等のランダムロジック：数十gate

程度の規模になる。



第9図 ハードウェア・ブロック図

5. おわりに

仮ラベル付け時に一塊の対象物に対する複数ラベル割り当てを大幅に削減できる手法『ラスト走査型ランラベル方式』を提案し、そのハードウェア化についても述べた。

本手法の効果としては複数割り当てが発生する2種類の画像形態(階段状、U字型)の内、階段上画像に関してはその統合を完全に0にすることができ、U字型画像に関してはその深さが1画素の時に統合を0にすることができると示した。

また、そのハードウェア化に関しては比較的小規模な回路構成で本手法を実現できることを示した。

今後は本手法を用いたラベリング処理を実行するリアルタイムプロセッサを開発していく予定である。

6. 参考文献

- (1) Rosenfeld, A (長尾真監訳) : "デジタル画像処理", 近代科学社, '78, page360
- (2) 奥山ら : "ラスト走査型ラベリングの高性能化の検討", 信学会技術研報PU87-32, '87, Page35-42
- (3) 太田ら : "ビデオレート・ラベリングプロセッサ", 情処学会CV研CV54-1, '88, Page1-8
- (4) 榎井ら : "内容照合LUTを用いたラベリング方式", 信学会技術研報PU87-79, '87, Page17-24
- (5) 服部ら : "ビデオ・レートで連続画像処理可能な高速ラベリング・プロセッサ", 画像工学コン(第18回), '87, Page295-298