

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号  
特許第5854802号  
(P5854802)

(45) 発行日 平成28年2月9日(2016.2.9)

(24) 登録日 平成27年12月18日(2015.12.18)

(51) Int.Cl.

G 0 6 T 7 / 6 0 ( 2 0 0 6 . 0 1 )

F 1

G 0 6 T 7 / 6 0 2 5 0 A

請求項の数 11 (全 24 頁)

(21) 出願番号	特願2011-263422 (P2011-263422)	(73) 特許権者	000001007
(22) 出願日	平成23年12月1日 (2011.12.1)		キヤノン株式会社
(65) 公開番号	特開2013-114655 (P2013-114655A)		東京都大田区下丸子3丁目30番2号
(43) 公開日	平成25年6月10日 (2013.6.10)	(74) 代理人	100126240
審査請求日	平成26年9月12日 (2014.9.12)		弁理士 阿部 琢磨
		(74) 代理人	100124442
			弁理士 黒岩 創吾
		(72) 発明者	石田 良弘
			東京都大田区下丸子3丁目30番2号キヤ
			ノン株式会社内
		(72) 発明者	常松 祐一
			東京都大田区下丸子3丁目30番2号キヤ
			ノン株式会社内
		審査官	板垣 有紀
			最終頁に続く

(54) 【発明の名称】 画像処理装置、画像処理方法、及びコンピュータプログラム

(57) 【特許請求の範囲】

【請求項 1】

所定サイズの画素マトリクスで多値画像をラスタ走査し、当該ラスタ走査したときの画素マトリクス内の複数画素に対応する前記多値画像における値に基づいて、異なる値をもつ画素間の境界を示す輪郭を区分すべき点である色交点の判定と、当該異なる値をもつ画素間の境界を示す前記輪郭を定義する輪郭点の抽出とを行う色交点判定・輪郭点抽出手段と、

前記色交点判定・輪郭点抽出手段で判定された色交点と抽出された輪郭点とを用いて、当該色交点ごとに区分した輪郭線で構成される輪郭情報を生成する輪郭情報再構成手段とを有することを特徴とする画像処理装置。

【請求項 2】

前記色交点判定・輪郭点抽出手段は、  
前記ラスタ走査したときの画素マトリクス内の複数画素に対応する前記多値画像における値に基づいて、前記異なる値をもつ画素間の境界を示す前記輪郭を区分すべき点である色交点を判定する色交点判定手段と、

前記ラスタ走査したときの画素マトリクス内の複数画素についての前記多値画像における値と、前記色交点判定手段の判定結果とに基づいて、異なる値をもつ画素間の境界を示す前記輪郭を定義する輪郭点を抽出する輪郭点抽出手段と

を有することを特徴とする請求項 1 に記載の画像処理装置。

【請求項 3】

前記色交点とは、3種以上の異なる値を持つ画素が会合する点、または、複数の色が会合する点で且つ斜め方向の隣接する画素が同じ値でかつ左右方向及び上下方向に隣接する画素が異なる色である点、のいずれかを満たす点である、請求項1または2に記載の多値画像の画像処理装置。

【請求項4】

前記画素マトリクスは、3画素×3画素サイズのマトリクスであることを特徴とする請求項1乃至3のいずれか1項に記載の画像処理装置。

【請求項5】

前記色交点判定・輪郭点抽出手段は、前記画素マトリクスに含まれる3画素×3画素における中心画素の値と同じ値を有する画素の領域の輪郭を定義する輪郭点を抽出することを特徴とする請求項4に記載の画像処理装置。

10

【請求項6】

前記画素マトリクスは、2画素×2画素サイズのマトリクスであることを特徴とする請求項1乃至3のいずれか1項に記載の画像処理装置。

【請求項7】

入力画像に対して、同色であると判定した画素ごとに同じ領域番号が付与することにより、各画素に付与された領域番号を値として有する前記多値画像を生成する領域番号付与手段を更に有することを特徴とする請求項1乃至6のいずれか1項に記載の画像処理装置。

【請求項8】

20

前記入力画像は、スキャン入力された画像に対してノイズ除去を行った後の画像であることを特徴とする請求項7に記載の画像処理装置。

【請求項9】

前記輪郭情報再構成手段で生成された前記色交点ごとに区分した輪郭線で構成される輪郭情報に対して、当該区分された輪郭線ごとに関数近似を行う関数近似手段を更に有することを特徴とする請求項1乃至8のいずれか1項に記載の画像処理装置。

【請求項10】

抽出手段が、所定サイズの画素マトリクスで多値画像をラスタ走査し、当該ラスタ走査したときの画素マトリクス内の複数画素に対応する前記多値画像における値に基づいて、異なる値をもつ画素間の境界を形成する輪郭を区分すべき点である色交点の判定と、

30

当該異なる値をもつ画素間の境界を形成する輪郭点の抽出とを行う色交点判定・輪郭点抽出ステップと、  
輪郭情報再構成手段が、前記色交点判定・輪郭点抽出ステップで判定された色交点と抽出された輪郭点とを用いて、当該色交点ごとに区分した輪郭線で構成される輪郭情報を生成する輪郭情報再構成ステップと

を有することを特徴とする画像処理方法。

【請求項11】

コンピュータを、

所定サイズの画素マトリクスで多値画像をラスタ走査し、当該ラスタ走査したときの画素マトリクス内の複数画素に対応する前記多値画像における値に基づいて、異なる値をもつ画素間の境界を形成する輪郭を区分すべき点である色交点の判定と、当該異なる値をもつ画素間の境界を形成する輪郭点の抽出とを行う色交点判定・輪郭点抽出手段、

40

前記色交点判定・輪郭点抽出手段で判定された色交点と抽出された輪郭点とを用いて、当該色交点ごとに区分した輪郭線で構成される輪郭情報を生成する輪郭情報再構成手段、として機能させるためのコンピュータプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

多値画像から、異なる色領域間を隔てる境界（画素境界の頂点境界線）を高速に漏れなく、所定の順番通りに抽出する画像処理装置、画像処理方法、コンピュータプログラムに

50

関する。

【背景技術】

【0002】

画像のベクトル化に関連する技術は文字を対象としたものからスタートし、輪郭情報の抽出方法（特許文献1）や座標点列の関数近似方法（特許文献2）が提案された。さらに一般的なイラスト画像へと対象が広がり、線画像に対するベクトル化（特許文献3）や、カラー画像に対するベクトル化（特許文献4）等、様々な種類の画像にベクトル化処理を施す技術が提案されている。

【0003】

ベクトル化されたデータは、所望のサイズに変倍してもジャギーの無いスムーズな輪郭表現が可能である。色数が少ないイラスト画像をベクトルデータに変換した場合は、データサイズの削減効果も有し、コンピュータでの編集処理等が容易となる利点もある。

【0004】

また、文字や線画像といった種類の画像は2値画像、あるいは単色画像（文字や線の部分が単色で構成され且つそれ以外の部分は背景の白地である画像）として扱えるものが多い。このような画像に対しては、色連結領域毎に抽出した輪郭（文字や線画像の輪郭）を関数近似すれば好ましいベクトル化結果を得ることができる。なお、色連結領域（色領域）とは、同色と判断される連結画素により構成される領域（同色画素が繋がっている領域）である。一方、イラストは2色より多い複数色で構成されるものが多く、複数の色領域間の境界線の扱い方が問題となる。例えば2値画像での処理と同様に、多色画像から抽出した色領域それぞれの輪郭を関数近似すると、隣接する色領域それぞれの輪郭が個別に近似されることになる。この場合、近似誤差によって、隣接する色領域間に共通の境界線部分に対して、2つの異なる近似曲線が得られてしまい、隙間や重なりが生じる問題があった。この様子を図3に示す。図3（a）は入力画像から色領域それぞれの輪郭形状を抽出した例を示している。背景の白地部分の領域を数えないものとする、他の色領域に隣接する（他領域との境界線を持つ）色領域が3つ存在している。図3（b）は、図3（a）の輪郭情報を色領域毎に個別に関数近似した場合の処理結果を示しており、色領域間に隙間や重なりが生じている。

【0005】

上記の問題を踏まえ、色領域間に隙間を生じさせないベクトル化法が提案されている。特許文献5では色差が一定以上ある画素同士の境界を輪郭として追跡し、色領域の交点（以降、『色交点』と称する）に突き当たる度に分岐して探索を繰り返し行う。色領域の交点間で区切られた部分輪郭それぞれを関数近似の処理単位とし、関数近似後のデータを再度つなぎ合わせて各色領域のベクトルデータを生成するため、境界線に共通の関数近似結果が適用され隙間や重なりが原理的に生じない。

【先行技術文献】

【特許文献】

【0006】

【特許文献1】特登録3026592

【特許文献2】特登録4378208

【特許文献3】特開2007-293829

【特許文献4】特開2006-344069

【特許文献5】特開2006-031245

【非特許文献】

【0007】

【非特許文献1】鳥脇著“画像理解のためのデジタル画像処理（2）”初版3刷，ISBN4-7856-2004-8，昭晃堂，1993年4月20日発行，P.66-P.79

【発明の概要】

【発明が解決しようとする課題】

## 【 0 0 0 8 】

非特許文献 1 は、色領域の境界線を追跡することにより、色領域の境界線を抽出する。境界線は異なる色領域各々を図形と考えた場合に、当該図形の内部の点と外部の点との境界の集合を指す。境界線の追跡方法としては、非特許文献 1 の 6 8 ページに記載されているように、画素追跡型、辺追跡型、頂点追跡型が知られている。ここでは、画像中で隣接する色領域間に隙間のない共通の輪郭線が定義されるという点で頂点追跡型に絞って話を進める。頂点追跡による境界線追跡法としては、特許文献 5 による境界線追跡が知られている。これによれば、抽出対象である色領域の探索開始点を探した後、抽出対象を右側または左側に決めて色境界をたどっていく。色交点にぶつかったらこれまで抽出した境界線を記録し、新たな抽出を開始する。こうして全ての境界線が抽出し終わるまで追跡を繰り返し行うというものである。

10

## 【 0 0 0 9 】

この方法にはいくつかの問題点がある。まず色交点に遭遇した時の探索順（複数ある時にどれを優先するか）が何通りか存在する。抽出順の選び方によっては境界線を並べて輪郭を再構成することが難しくなったり、抽出漏れが起きやすくなったりする。また追跡の方向が領域形状によって変わるため、メモリ上の画素情報を参照する際にキャッシュが効かず、処理速度が低下することも起こりうる。

## 【 0 0 1 0 】

なお、本出願人は、上述したような境界線を順に追跡していく方法を用いずに、境界線を抽出する方法として、以下の手法を考えている。本出願人が提案している特願 2 0 1 0 - 2 3 4 0 1 2 では、まず、ラスター走査で各色領域毎に境界線を構成する最小単位となる線素のテーブルを作成後、当該テーブルを走査して輪郭を抽出する。そして、抽出した色領域の輪郭をたどることで、隣接する他の色領域とで構成される色交点を検知し、輪郭を色交点ごとに分割する。この方法は、境界線の抽出漏れや境界線の再構成順序の問題を解決することができ、ラスター走査とテーブル走査で高速に輪郭を抽出する方法である。しかしながら、色交点も考慮した境界線を求めるには、抽出した輪郭を再度辿りなおすために、追跡と同じように輪郭を 2 度探索することになり、さらなる改良の余地が残されていた。

20

## 【課題を解決するための手段】

## 【 0 0 1 1 】

そこで、本発明の画像処理装置は、所定サイズの画素マトリクスで多値画像をラスター走査し、当該ラスター走査したときの画素マトリクス内の複数画素に対応する前記多値画像における値に基づいて、異なる値をもつ画素間の境界を示す輪郭を区分すべき点である色交点の判定と、当該異なる値をもつ画素間の境界を示す前記輪郭を定義する輪郭点の抽出とを行う色交点判定・輪郭点抽出手段と、

30

前記色交点判定・輪郭点抽出手段で判定された色交点と抽出された輪郭点とを用いて、当該色交点ごとに区分した輪郭線で構成される輪郭情報を生成する輪郭情報再構成手段とを有することを特徴とする。

## 【発明の効果】

## 【 0 0 1 2 】

1 回のラスター走査で輪郭点と色交点とを取得するようにしたので、各領域の輪郭を示す輪郭情報（境界線）の生成を高速に行うことができる。

40

## 【図面の簡単な説明】

## 【 0 0 1 3 】

【図 1】本実施形態における情報処理装置のメイン処理ブロック図。

【図 2】本実施形態における情報処理装置のブロック図。

【図 3】関数近似によって色領域に重なりや隙間が生じる例を説明する図。

【図 4】本実施形態におけるフローチャート。

【図 5】3 × 3 の画素マトリクスでの画像からラスター走査を説明する図。

【図 6】注目ベクトルと、その流入ベクトル、流出ベクトルとの関係を示す図。

50

【図 7】一点のみで構成されるラベルのベクトル情報テーブルによる記録を示す図。

【図 8】 $3 \times 3$  の画素マトリクスの中心画素とその上下左右の 4 近傍画素を表す図。

【図 9】ケース 1 から 16 の各場合の色交点判定・輪郭点抽出処理を説明する図。

【図 10】色交点判定を伴わない輪郭点の抽出を説明する図。

【図 11】色交点判定を伴う輪郭点の抽出を説明する図。

【図 12】色交点判定を伴う輪郭点抽出時の色交点判定を説明する図。

【図 13】色交点判定し、色交点の場合のみ輪郭点を抽出する処理を説明する図。

【図 14】ステップ S 1200 での処理の流れを、具体例を元にして説明する図。

【図 15】具体例の場合での流入ベクトル未定水平ベクトルテーブル、流出ベクトル未定水平ベクトルテーブル、流入ベクトル未定垂直ベクトルテーブル、流出ベクトル未定垂直ベクトルテーブルの 4 種のテーブルの状態を表す図。

【図 16】 $2 \times 2$  のウィンドウ走査で起こりえる 4 画素のパターンを示す図。

【発明を実施するための形態】

【0014】

[ 実施例 1 ]

本実施形態の画像処理装置の構成例について、図 2 のブロック図を参照して説明する。画像処理装置は、汎用のコンピュータを用いて実現することが可能である。図 2 において、CPU 7 は装置全体を制御するプロセッサである。ROM 6 は変更を必要としないプログラムやパラメータを格納するメモリである。RAM 5 は外部装置等から供給されるプログラムやデータを一時記憶するメモリである。スキャナ 1 は、文書等を光電走査して電子的な画像データを得る。入出力 I/O 3 はスキャナ 1 と画像処理装置とを接続するインタフェースである。画像メモリ 2 はスキャナ 1 で読み取られた画像データ等を保持する。外部記憶装置 12 は固定して設置されたハードディスクやメモ리카ード、光ディスク等の記憶媒体である。また、I/O 13 は外部記憶装置 12 と画像処理装置とを接続するインタフェースである。I/O 15 は、マウス等のポインティングデバイス 10 やキーボード 9 等の入力デバイスと画像処理装置を接続するインタフェースである。映像 I/O 14 は、画像処理装置の保持するデータや供給されたデータを表示するためのディスプレイ 8 と画像処理装置とを接続するインタフェースである。通信 I/F 4 は、インターネット等のネットワーク回線（不図示）に接続するためのインタフェースである。システムバス 11 は各ユニットを通信可能に接続するバスである。

【0015】

以下、CPU 7 上で実行されるプログラムにより本発明を実現する処理手順を、図 1 のブロック図と図 4 のフローチャートを用いて説明する。

【0016】

ステップ S 1000 で処理を開始すると、処理対象となる画像領域を含む画像データを入力する。画像入力に関しては、スキャナ 1 にて読み取られた画像データを画像入出力 I/O 3 を介して画像メモリ 2 に入力する。また、通信 I/F 4 を介して、装置外部より前記処理対象となる画像領域を含む画像を入力してもよく、あるいは、外部記憶装置 12 に予め記憶される画像データを I/O 13 を介して読み込むものであってもよい。得られた入力画像は、画像メモリ 2 上に保持される。

【0017】

スキャナなどの入力手段を用いた場合、入力画像にノイズが重畳され、代表色の特定が難しいことが起こりうる。この場合、色領域分割処理を行うことで、近い画素値を持つ画素が同じ色情報となるようにまとめるようにするのが望ましい。そこでステップ S 1000 では CPU 7 が入力画像に対して色領域分割処理を施すことで、上記問題を解消することができる。ここでの色領域分割処理については、本発明が適用可能であれば、いずれの方法を用いてもよい。例えば特許文献 4 で開示されている手法では、入力画像中の画素から色情報を元にクラスタを構成し、類似するクラスタやノイズと思われるクラスタを統合することでスキャンノイズを除去する。本実施形態においては、このような方法を適用することで、スキャン画像入力等で生じるノイズ除去している。以上の処理は図 1 における

画像入力手段 101 を実現している。

【0018】

次にステップ S1000 で読み取った画像データに対して、ステップ S1100 でラベリング処理（領域番号（ラベル）付与処理）を行うことにより、画像データ中の同色と判断される領域に同一の領域番号を付与して、色領域を抽出する。ラベリング処理は、8 連結接続で考える場合、上下左右斜めの 8 方向のいずれかで隣接（連結）する同じ画素値を有する画素集合に同一の番号を付与する処理である。ここでは各色領域を後処理で識別できるようにするための番号（以下、領域番号）を、各色領域を構成する各画素に付与する。ラベリング処理（領域番号（ラベル）付与処理）された画像を、ラベル画像と呼ぶこととする。ラベリング処理では後で行う輪郭抽出処理と同じ連結状態となるラベリング処理を行う必要がある。本実施形態では、8 連結接続で輪郭を抽出する例を用いて説明するので、ラベリング処理も 8 連結接続として処理し、上下左右斜め方向のいずれかで隣接する同じ画素値を有する画素集合に同一の番号を付与することになる。なお、ステップ S1100 の処理は図 1 における領域番号付与手段 102 を実現している。

10

【0019】

ステップ S1200 では、ステップ 1100 で得られたラベル画像を入力画像として、図 5 (a) に示すような  $3 \times 3$  の画素マトリクス状ウィンドウの中心（同図中「x」で表示）を着目画素として、図 5 (b) に示すように入力画像の左上からラスタ走査を行なう。そして、各ラベル領域毎にそのラベル領域の輪郭を構成する水平方向の線素と垂直方向の線素の切り替わる位置を、それぞれの線素の始点（以降、単に輪郭点とも称する）として抽出する処理を行う。加えて、それら輪郭点が色交点であるか否かの判定する色交点判定処理を行う。それら線素上に存在する可能性のある色交点の存在判定の結果、色交点が存在する場合には、その位置に色交点である輪郭点の抽出処理を行う。尚、抽出される垂直・水平方向の線素を、それぞれ垂直ベクトル、水平ベクトルと呼ぶ。また、線素上に存在が判定された色交点も、当該線素が水平の場合には新たな水平ベクトルの始点となり、同様に当該線素が垂直な場合には垂直ベクトルの始点となる。輪郭点抽出処理、輪郭点の色交点判定、線素上での色交点判定と色交点である位置での輪郭点抽出処理を総称して、色交点判定・輪郭点抽出処理と呼ぶ。

20

【0020】

本実施形態では、水平ベクトルと垂直ベクトルは進行方向に対して、着目画素の属するラベル領域が右側に来るよう抽出される。この結果、抽出される外輪郭（輪郭抽出対象のラベル領域を外側から取り囲む輪郭）は時計回りとなる。また、内輪郭（輪郭抽出対象のラベル領域領域に周囲を取り囲まれる孔の境界に沿って、当該ラベル領域が右側に来るよう抽出される内部の輪郭）は反時計回りとなる。座標系については、特許文献 1 での説明と同様に、本実施例では、入力画像の各画素の座標値を、主走査方向を x 軸（右向きが正方向）、副走査方向を y 軸（下向きが正方向）とし、それぞれ整数値で表現するものとして話を進める。また、抽出される線素の始点や色交点の座標値は、画素と画素の間ので位置に抽出されるので、これらの座標値を整数値として扱うため、当該座標値を 2 倍にした値を対象座標として抽出する。即ち、 $m$  画素  $\times$   $n$  画素の画像は、 $2m \times 2n$  の正の偶数（整数）の座標表現で表し、抽出される線素の始点や色交点の座標値が、整数値（奇数）となるようにして説明する。以降、第  $j$  番目のラスタの第  $i$  番目の画素位置を  $(2i, 2j)$  ( $i, j$  は正の整数で、 $i \leq m, j \leq n$ ) で表現する。

30

40

【0021】

抽出した垂直ベクトルの始点や水平ベクトルの始点の情報には、それぞれに流入元となるベクトルの始点と、流出先となるベクトルの始点の情報への接続情報を付して記録する。以降、流入元となるベクトルの始点を単に流入元、流出先となるベクトル始点を流出先と称し、流入元となるベクトルを流出ベクトル、流出先となるベクトルを流出先と称する。着目しているベクトルの始点は、流入ベクトルの終点でもあり、かつ、流出ベクトルの始点は、着目しているベクトルの終点でもある。この様子を図 6 に一例として示した。同図は、注目ベクトルが水平ベクトルである場合を表記していて、61 が注目ベクトル、6

50

2 が注目ベクトル 6 1 の始点であり、かつ、流入ベクトル 6 3 の終点でもある。6 4 が流入ベクトル 6 3 の始点（流入元）、6 5 が流出ベクトル、6 6 が流出ベクトル 6 5 の始点（流出先）であり、かつ、注目ベクトル 6 1 の終点でもある。ここで、垂直ベクトルの始点を白抜き三角印、水平ベクトルの始点を白抜き丸印で表記した。

#### 【 0 0 2 2 】

これらのベクトルはラベルごとに 1 つのベクトル情報テーブルに記録して管理する。例として、図 7 に、注目するラベルが一点（1 画素）のみで構成されるラベルである場合の前記ベクトル情報テーブルによる記録を示した。図 7（a）は、一点のみで構成されるラベルを中心にその周囲 8 画素を含めて  $3 \times 3$  のラベル画像を表現しており、周囲 8 画素は中心画素のラベルとは異なるラベルをもつ画素である。同ラベルは、水平ベクトル（ $r$ ）、垂直ベクトル（ $r + 2$ ）、水平ベクトル（ $r + 3$ ）、垂直ベクトル（ $r + 1$ ）の 4 本の線素で囲まれている。同図（b）は、この様子を記録したベクトル情報テーブルである。図 7（b）のテーブルにおいて、各ベクトルは横一行で 1 本の線素を表現している。ベクトルカウンタ欄は、抽出された順にカウントアップされながら定まるベクトル番号（ベクトルインデックスやテーブルインデックスとも称することがある）である。ベクトル番号は、基本的に、輪郭点、もしくは、色交点として抽出された順に昇順に定められる番号である。図 7（a）の状態の説明における  $r$ 、 $r + 1$ 、 $r + 2$ 、 $r + 3$  も、抽出された順に定められるベクトル番号ではあるが、このベクトル番号をもって、各ベクトルを特定するベクトルの名称としても用いるものとする。なお、説明の便宜上、ベクトル番号の初期値として  $r$  を用いたが、初期値が 0 から始まるものとする、これらベクトル番号は、0、1、2、3 となる。始点欄は、X 座標欄と Y 座標欄から構成され、それぞれ、各ベクトルの始点の X 座標値と Y 座標値を格納する。図 7（b）では、中心の画素の座標値が（ $2i$ 、 $2j$ ）であったとして、各ベクトルの始点の座標値を記載してある。流入ベクトル欄は、当該ベクトル番号のベクトルの流入元となるベクトルのベクトル番号が格納される。同様に、流出ベクトル欄は、流出先となるベクトルのベクトル番号が格納される。色交点フラグ欄は、同ベクトル番号のベクトルが色交点である場合には“TRUE”が、色交点ではない場合には“FALSE”がはいるフラグ領域である。未使用フラグ欄は、ステップ S1300 の輪郭情報再構成処理で参照するフラグ領域であり、これに関しては、追ってさらに説明を加える。

#### 【 0 0 2 3 】

$3 \times 3$  のウィンドウでは、図 8 に示したように、 $3 \times 3$  のウィンドウの中心である着目画素と着目画素の上下左右の 4 近傍画素に着目し、着目画素の周辺での輪郭ベクトルの始点や色交点の抽出、及び、それらの局所的な接続関係を求める。まず、着目画素  $P_0$  が輪郭抽出対象のラベル領域の画素である場合には  $R_0 = 1$  とした上で、着目画素  $P_0$  と、図 7 に示す  $P_1$ 、 $P_2$ 、 $P_3$ 、 $P_4$  の 4 近傍の各画素とを比較し、着目画素の所属ラベルと所属ラベルが一致した画素の位置  $P_k$ （ $k = 1, 2, 3, 4$ ）の値  $R_k$  を 1 とし、一致しない画素位置では  $R_k$  を 0 とする。そして、これら  $R_0$ 、 $R_1$ 、 $R_2$ 、 $R_3$  に係数をかけて定義する以下の式により  $R$  を求める。

#### 【 0 0 2 4 】

$$R = R_0 + 1 \times R_1 + 2 \times R_2 + 4 \times R_3 + 8 \times R_4$$

尚、着目画素  $P_0$  が輪郭抽出対象のラベル領域の画素である場合には、4 近傍画素の状態にかかわらず、 $R = 0$  とする。

#### 【 0 0 2 5 】

上記  $R$  は、着目画素と着目画素の上下左右の 4 近傍画素の所属ラベルの状態によって、0 ~ 16 の値を持つのに対応させて、それぞれの場合をケース 0 からケース 16 で区別する。各ケースに応じて所定の色交点判定・輪郭点抽出処理を行う。ケース 0 では、色交点判定・輪郭点抽出は全くなされない。

#### 【 0 0 2 6 】

ケース 1 ~ ケース 16 の場合の色交点判定・輪郭点抽出処理を図 9 に示した。図 9 においては、中心画素のラベルに対して、その左右上下の 4 画素がそれぞれ同じラベルか否か

で表現してあり、同 4 画素のうち、中心画素のラベルと同じラベルの画素は中心画素と同じ塗りつぶしパターンで表記し、そうでないものは塗りつぶしていない。尚、図 9 における中心画素の左上、左下、右上、右下の位置にある画素は、ケース 1 ~ 16 のいずれに属するかを分類する上では関わりをもたない非注目画素であることから、図 9 ではどんなラベルを持つ画素かによらず塗りつぶしのない状態で表記してある。図 9 ( a ) から図 9 ( p ) までの 16 種類の  $3 \times 3$  のウィンドウの状態にあわせて、それぞれ、a から p までのアルファベット順に、ケース 1 からケース 16 の処理内容をケース番号の昇順に対応させて表記してある。図 9 において、白抜き丸印は、色交点であるか否かの判断をすることなしに、色交点ではない水平ベクトルの始点の抽出処理をする輪郭点候補位置である。白抜き三角印は、色交点であるか否かの判断をすることなしに、色交点ではない垂直ベクトルの始点の抽出処理をする輪郭点候補位置である。横向き矢印を伴った黒塗り丸印は、色交点であるか否かの判断を伴う水平ベクトルの始点の抽出処理をする輪郭点候補位置である。縦向き矢印を伴った黒塗り三角印は、色交点であるか否かの判断を伴う垂直ベクトルの始点の抽出処理をする輪郭点候補位置である。矢印の伴わない黒塗り丸印は、色交点か否かを判定し、色交点である場合には、色交点である水平ベクトルの始点を抽出する候補位置である。矢印の伴わない黒塗り三角印は、色交点であるか否かを判定し、色交点である場合には、色交点である垂直ベクトルの始点を抽出する候補位置である。実線矢印は、この矢印の位置にベクトルが抽出される場合には、流出先となる抽出済みのベクトルが存在するものを意味している。破線矢印は、この矢印の位置にベクトルが抽出された場合、流出先となるべきベクトルが、まだ抽出されていない状態、即ち、流出先が未定の状態で抽出されることになるベクトルであることを意味している。尚、以降、水平、垂直共に、ベクトルの始点の抽出のことを、単に、ベクトルの抽出と表現することがある。

#### 【 0 0 2 7 】

図 10 を用いて、色交点であるか否かの判断を伴わない輪郭点の抽出を説明する。図 10 は、ケース 16 での処理内容を示す図であり、中心画素のラベルと、その上下左右の 4 画素のラベルは全て同じ状態にある。図 10 ( a ) は、図 10 内で改めてケース 16 のパターンを示すために、図 9 ( p ) を再掲したものである。

#### 【 0 0 2 8 】

まず、図 10 ( b ) に示すように、中心画素 ( 座標値が  $(2i, 2j)$  であるとする ) の左上に垂直ベクトルの始点が存在するか否かを判定する。即ち、中心画素の左上の画素 ( 座標値は  $(2i - 2, 2j - 2)$  ) が、中心画素のラベルと同じラベルであるか否かを判定し、異なる場合には、図 10 ( c ) に示すように中心画素の左上の位置 ( 座標値は  $(2i - 1, 2j - 1)$  ) に垂直ベクトルを抽出する。同じ場合には、図 10 ( d ) に示すように、同位置にベクトルは抽出されない。

#### 【 0 0 2 9 】

次に、図 10 ( e ) に示すように、中心画素の左下に水平ベクトルの始点が存在するか否かを判定する。即ち、中心画素の左下の画素 ( 座標値は  $(2i - 2, 2j + 2)$  ) が、中心画素のラベルと同じラベルであるか否かを判定し、異なる場合には、図 10 ( f ) に示すように中心画素の左下の位置 ( 座標値は  $(2i - 1, 2j + 1)$  ) に水平ベクトルを抽出する。同じ場合には、同図 ( g ) に示すように、同位置にベクトルは抽出しない。

#### 【 0 0 3 0 】

次に、図 10 ( h ) に示すように、中心画素の右上に水平ベクトルの始点が存在するか否かを判定する。即ち、中心画素の右上の画素 ( 座標値は  $(2i + 2, 2j - 2)$  ) が、中心画素のラベルと同じラベルであるか否かを判定し、異なる場合には、図 10 ( i ) に示すように中心画素の右上の位置 ( 座標値は  $(2i + 1, 2j - 1)$  ) に水平ベクトルを抽出する。同じ場合には、図 10 ( j ) に示すように、同位置にベクトルは抽出しない。

#### 【 0 0 3 1 】

さらに、図 10 ( k ) に示すように、中心画素の右下に垂直ベクトルの始点が存在するか否かを判定する。即ち、中心画素の右下の画素 ( 座標値は  $(2i + 2, 2j + 2)$  ) が、中心画素のラベルと同じラベルであるか否かを判定し、異なる場合には、図 10 ( l )



に示すように中心画素の右下の位置（座標値は（ $2i + 1$ ， $2j + 1$ ））に垂直ベクトルを抽出する。同じ場合には、図10（m）に示すように、同位置にベクトルは抽出しない。

【0032】

以上、図10を用いて、ケース16での処理内容と、色交点であるか否かの判断を伴わない輪郭点の抽出について説明した。

【0033】

ここで、ベクトルを抽出した際のベクトル情報テーブル上でのデータの扱いについて、さらに説明を加える。注目するラベル領域（中心画素の属するラベル領域）に対する（図7（b）を用いて、上述した形式の）ベクトル情報テーブルにおいて、それまで既に見つけられていたベクトル情報の次の行に相当するテーブル領域にベクトル情報を追記する。

10

【0034】

まず、該当行のベクトルカウンタ欄に、抽出された順に応じたベクトル番号を書き込む。

【0035】

次に、抽出されたベクトルの座標値（X座標値、Y座標値）を、始点のX座標値とY座標値の欄に書き込む。

【0036】

流入ベクトル欄と流出ベクトル欄については、図10（1）のケースのように、抽出したベクトルの流出先や流入元が、まだ走査されていない領域（以降、未走査領域とも称する）方向にある場合と、図10（c）のように、既に走査し終えた領域（以降、既走査領域とも称する）方向にある場合とで扱いが異なる。未走査領域方向に流入元や流出先がある場合には、前述のベクトル情報テーブルとは別に、流入ベクトルが未定のベクトルの情報を管理するテーブルと流出ベクトルが未定のベクトルの情報を管理するテーブルとをそれぞれ別途用いて処理する。垂直ベクトル用の流入ベクトル未定のベクトルを管理するテーブルを、流入ベクトル未定垂直ベクトルテーブルと称する。また、垂直ベクトル用の流出ベクトル未定のベクトルを管理するテーブルを、流出ベクトル未定垂直ベクトルテーブルと称する。水平ベクトル用にも同様に、流入ベクトル未定水平ベクトルテーブルと流出ベクトル未定水平ベクトルと称するテーブルをそれぞれ用いる。それぞれのテーブルには、該当するベクトル番号を検出した順に格納していくものとする。図10（1）のケースの場合には、ベクトル情報テーブルの流入ベクトル欄には、その時点では何も記録せず、流入ベクトル未定垂直ベクトルテーブルに、ベクトル情報ベクトル上でのベクトル番号を追記して同テーブルを更新する。同様に、ベクトル情報テーブルの流出ベクトル欄には、その時点では何も記録せず、流出ベクトル未定垂直ベクトルテーブルに、ベクトル情報ベクトル上でのベクトル番号を追記して同テーブルを更新する。尚、図10（i）で示される水平ベクトルを抽出する場合の流出ベクトル欄も同様に、その時点では何も記入せず、流出ベクトル未定水平ベクトルにそのベクトル番号を追記して同テーブルを更新する。図10（f）で示される水平ベクトルが抽出される際は、その時点では流入ベクトル欄に何も記入せず、流入ベクトル未定水平ベクトルテーブルにそのベクトル番号を記入して同テーブルを更新する。

20

30

40

【0037】

一方、上述の図10（c）の垂直ベクトルの様に、流入元や流出先が既走査領域方向にある場合は、上記の流入ベクトルが未定のベクトルの情報を管理するテーブルや流出ベクトルが未定のベクトル情報を管理するテーブル参照して、その流入元や流出先を定める。

【0038】

流入元が既走査領域方向にある垂直ベクトルの流入元は、流出先が未定の水平ベクトルである。よって、このような垂直ベクトルの流入元を定める場合には、流出ベクトル未定水平ベクトルテーブルを参照する。このテーブルに格納されているベクトル番号をもつベクトルの中で、当該垂直ベクトルの始点と同じY座標値を持つベクトルを該当するラベルのベクトル情報テーブルから探し出す。ベクトル情報テーブルの当該垂直ベクトルの流入

50

ベクトル欄に、得られた水平ベクトルのベクトル番号を記入し、それまで未定であったその水平ベクトルの流出ベクトル欄に、当該垂直ベクトルのベクトル番号を記載する。また、流出ベクトル未定水平ベクトルテーブルからその水平ベクトルのベクトル番号を削除して同テーブルを更新する。

#### 【 0 0 3 9 】

次に、当該垂直ベクトルの様に、流出先が既走査方向にある垂直ベクトルの流出先を定める場合を説明する。流出先が既走査領域方向にある垂直ベクトルの流出先は、流入元が未定の水平ベクトルか、または、後述する色交点位置に抽出された垂直ベクトルの中で流入元が未定のものである。よって、このような垂直ベクトルの流出先を定める場合には、流入ベクトル未定水平ベクトルテーブルと流入ベクトル未定垂直ベクトルテーブルの両方を参照する。これらテーブルに格納されているベクトル番号をもつベクトルの中で、当該垂直ベクトルの始点と同じX座標値を持つベクトルを該当するラベルのベクトル情報テーブルから探し出す。複数見つかった場合には、そのY座標値が当該垂直ベクトルのY座標値よりも小さいものの中で最も大きなY座標値をもつもの（即ち、既走査領域の中で最も近くにある流入元未定の輪郭点）を選ぶ。ベクトル情報テーブルの当該垂直ベクトルの流出ベクトル欄に、得られた水平、もしくは、垂直ベクトルのベクトル番号を記入し、それまで未定であったそのベクトルの流入ベクトル欄に、当該垂直ベクトルのベクトル番号を記載する。また、得られたベクトルの番号が記載されていた、流入ベクトル未定水平ベクトルテーブル、もしくは、流入ベクトル未定垂直ベクトルテーブルからそのベクトルのベクトル番号を削除して同テーブルを更新する。

#### 【 0 0 4 0 】

尚、図 1 0 ( i ) で示される水平ベクトルの様に、流入元が既走査領域方向にある水平ベクトルの流入元は、流出先が未定の垂直ベクトルである。よって、このような水平ベクトルの流入元を定める場合には、流出ベクトル未定垂直ベクトルテーブルを参照する。このテーブルに格納されているベクトル番号をもつベクトルの中で、当該水平ベクトルの始点と同じX座標値を持つベクトルを該当するラベルのベクトル情報テーブルから探し出す。ベクトル情報テーブルの当該水平ベクトルの流入ベクトル欄に、得られた垂直ベクトルのベクトル番号を記入し、それまで未定であったその垂直ベクトルの流出ベクトル欄に、当該水平ベクトルのベクトル番号を記載する。また、流出ベクトル未定垂直ベクトルテーブルからその垂直ベクトルのベクトル番号を削除して同テーブルを更新する。

#### 【 0 0 4 1 】

次に、図 1 0 ( f ) で示される水平ベクトルの様に、流出先が既走査方向にある水平ベクトルの流出先を定める場合を説明する。流出先が既走査領域方向にある水平ベクトルの流出先は、流入元が未定の垂直ベクトルか、または、後述する色交点位置に抽出された水平ベクトルの中で流入元が未定のものである。よって、このような水平ベクトルの流出先を定める場合には、流入ベクトル未定水平ベクトルテーブルと流入ベクトル未定垂直ベクトルテーブルの両方を参照する。これらテーブルに格納されているベクトル番号をもつベクトルの中で、当該水平ベクトルの始点と同じY座標値を持つベクトルを該当するラベルのベクトル情報テーブルから探し出す。万が一、複数見つかる場合には、そのX座標値が当該水平ベクトルのX座標値よりも小さいものの中で最も大きなX座標値をもつもの（即ち、既走査領域の中で最も近くにある流入元未定の輪郭点）を選ぶ。ベクトル情報テーブルの当該水平ベクトルの流出ベクトル欄に、得られた水平、もしくは、垂直ベクトルのベクトル番号を記入し、それまで未定であったそのベクトルの流入ベクトル欄に、当該水平ベクトルのベクトル番号を記載する。また、得られたベクトルの番号が記載されていた、流入ベクトル未定水平ベクトルテーブル、もしくは、流入ベクトル未定垂直ベクトルテーブルからそのベクトルのベクトル番号を削除して同テーブルを更新する。

#### 【 0 0 4 2 】

色交点フラグ欄には、追記中のベクトルが色交点である場合には、TRUEを記録し、色交点ではない場合にFALSEを記録する。ケース 1 6 では、色交点となる輪郭点はそもそも発生しないパターンであり、色交点であるか否かの判断は不要であった。このため

、上述の図10(c)、図10(f)、図10(i)、図10(l)で抽出される輪郭点は、色交点ではないため、いずれの場合も色交点フラグ欄はFALSEを記録する。

【0043】

未使用フラグ欄は、ステップS1200では、TRUEを記録する。同欄は、ステップS1300の輪郭情報再構成処理で参照するフラグ領域であり、追ってさらに説明を加える。

【0044】

次に、図11と図12を用いて、色交点であるかの判断(色交点判定)を伴う輪郭点の抽出とその色交点判定を説明する。

【0045】

図11は、ケース1での処理内容を示す図であり、中心画素のラベルと、その上下左右の4画素のラベルは全て異なる状態にある。図11(a)は、図11内で改めてケース1のパターンを示すために、図9(a)を再掲したものである。

【0046】

まず、図11(b)に示すように、中心画素(座標値が $(2i, 2j)$ であるとする)の左上の位置(座標値は $(2i-1, 2j-1)$ )に水平ベクトルの始点を抽出する。この水平ベクトルは、中心画素の上の画素が中心画素とはことなるラベルをもつので、流出先は未定のベクトルであるが、中心画素とその左上、上、左の4画素でなる $2 \times 2$ の領域の各画素のラベルの状態により、流入元が未定か確定するかは異なるものとなる。また、中心画素とその左上、上、左の4画素でなる $2 \times 2$ の領域の各画素のラベルの状態により、中心画素の左上の位置の水平ベクトルが、色交点となるか否かも異なる。図11(c)と図11(d)はともに、中心画素とその左上の画素のラベルが異なる場合をあらわしている。これらの場合には、当該水平ベクトルの流入ベクトルも未走査領域にあることになるので、流入ベクトルも流出ベクトルも未定の水平ベクトルとなる。一方、図11(e)と図11(f)はともに、中心画素とその左上の画素のラベルが同じ場合をあらわしている。これらの場合には、当該水平ベクトルの流入ベクトルは既走査領域に存在することになるので、流入ベクトルは確定していて、流出ベクトルは未走査の水平ベクトルとなる。尚、当該水平ベクトルが、流入ベクトルも流出ベクトルも未定の水平ベクトルであっても、色交点である場合とない場合とがあり得る。この様子を図11(c)と図11(d)に示した。図11(c)は、当該水平ベクトルの始点が色交点ではない場合であり、白抜き丸印で表わした。図11(d)は、当該水平ベクトルの始点が色交点の場合であり、右上がり斜線で塗りつぶした丸印で表わした。また、該水平ベクトルが、流入ベクトルは確定していて、流出ベクトルは未走査の水平ベクトルの場合も同様に、色交点である場合とない場合とがあり得る。この様子を図11(e)と図11(f)に示した。図11(e)は、当該水平ベクトルの始点が色交点ではない場合であり、白抜き丸印で表わした。図11(f)は、当該水平ベクトルの始点が色交点の場合であり、右上がり斜線で塗りつぶした丸印で表わした。

【0047】

この位置(中心画素の左上の位置)の水平ベクトルの始点の色交点判定について、図12(a)~(f)を用いて説明する。図12(a)自体は図9(e)、即ち、ケース5の状態をもとに表記したものであるが、中心画素の左上の位置の水平ベクトルの始点の色交点判定という意味では、図11(b)で表わされる、ケース1の中の同位置と同じ処理を行うものである。さて、図12(a)に示す様に、中心画素とその左上、上、左の4画素でなる $2 \times 2$ の領域の各画素のラベルの状態により、色交点判定が行われる。図12(b)~(d)は、いずれも中心画素とその左上の画素のラベルが異なる場合を示している。これらの場合には、 $2 \times 2$ の中の残り2画素(中心画素の上の画素と左の画素)のラベルも調べる。

【0048】

中心画素以外の3画素が全て同一のラベルである場合(図12(b))は、2色の領域でしか接していない状態なので、色交点ではない輪郭点と判断する。

## 【 0 0 4 9 】

また、中心画素の上と左が同一のラベルで、左上の画素はこれらとは異なるラベルの場合（不図示）には、上と左の画素が8連結状態にあり、中心画素と左上の画素は非連結の状態にある。この場合は、当該水平ベクトルの始点位置では2色の領域でしか接していない状態にあるものとみなし、色交点ではない輪郭点とする。

## 【 0 0 5 0 】

また、図12(c)に示す様に、中心画素の左上と左の画素が同ラベルで、残りの1画素（上の画素）が中心画素とも他の2画素とも異なるラベルの場合には、3色の領域で接しているので色交点である輪郭点とする。

## 【 0 0 5 1 】

また、中心画素の左上と上の画素が同ラベルで、残りの1画素（左の画素）が注目画素とも他の2画素とも異なるラベルの場合（不図示）にも、3色の領域で接しているので色交点である輪郭点とする。即ち、注目画素以外の3画素のうち、縦または横につながる2画素が同ラベルで、残りの1画素が注目画素とも他の2画素とも異なるラベルの場合には、色交点である輪郭点とする。

## 【 0 0 5 2 】

また、図12(d)に示す様に、中心画素以外の3画素がいずれも互いに異なるラベルである場合には、4色の領域が接する交点となっているので、色交点とする。

## 【 0 0 5 3 】

図12(e)と(f)は、中心画素と左上の画素のラベルが同じ場合、即ち、3×3の9画素領域中における2×2の4画素領域の中で、中心画素とその対角画素のラベルが同じ場合を示している。残りの2画素が、中心画素のラベルとは異なるラベルながら、同ラベルとなっている場合（図12(e)）は、どちらの対角画素の組も8連結状態にあり、この場合にはどちらの連結状態も同等に扱う意味で、色交点とする。一方、図12(f)に示す様に、残りの2画素が互いに異なるラベルである場合、先に述べた不図示の場合同様、中心画素と左上の画素が8連結状態にあり、上と左の画素は非連結の状態にある。当該水平ベクトルの始点位置では2色の領域でしか接していない状態にあるものとみなし、色交点ではない輪郭点とする。

## 【 0 0 5 4 】

さて、ここまで図11(b)～(f)と図12(a)～(f)を用いて、中心画素の左上の位置における、色交点であるかの判断（色交点判定）を伴う輪郭点の抽出とその色交点判定を説明した。一方、中心画素の左下の位置における色交点判定を伴う輪郭点の抽出と色交点判定も、流入元と流出先の扱いを含めて、抽出されるベクトルが垂直ベクトルの始点になる点とその位置が中心画素の左下であることを考慮する必要はあるものの、同様の処理内容となる。図11(g)～(k)と図12(g)～(l)が中心画素の左下の位置における色交点判定を伴う輪郭点の抽出と色交点判定を説明する図ではあるが、不図示の場合も含めて容易に適用可能であるため、これらを用いた詳述は省略する。以下、同様に、中心画素の右上の位置における色交点判定を伴う輪郭点の抽出と色交点判定は、図11(l)～(p)と図12(m)～(r)が中心画素の右上の位置における色交点判定を伴う輪郭点の抽出と色交点判定を説明する図である。抽出されるベクトルが垂直ベクトルの始点で、位置が中心画素の右上であることを考慮する必要はあるものの、不図示の場合も含めて容易に適用可能であるため、これらを用いた詳述は省略する。中心画素の右下の位置における色交点判定を伴う輪郭点の抽出と色交点判定は、図11(q)～(u)と図12(s)～(x)が中心画素の右下の位置における色交点判定を伴う輪郭点の抽出と色交点判定を説明する図である。抽出されるベクトルが水平ベクトルの始点で、位置が中心画素の右下であることを考慮する必要はあるものの、不図示の場合も含めて容易に適用可能であるため、これらを用いた詳述は省略する。

## 【 0 0 5 5 】

次に、図9の中の表記にあった、矢印を伴わない黒塗り丸印の位置での処理と、矢印を伴わない黒塗り三角印の位置での処理とを、図13を用いて説明する。矢印の伴わない黒

10

20

30

40

50

塗り丸印の位置での処理とは、色交点か否かを判定し、色交点である場合には、色交点である水平ベクトルの始点を抽出する処理である。矢印の伴わない黒塗り三角印の位置での処理とは、色交点であるか否かを判定し、色交点である場合には、色交点である垂直ベクトルの始点を抽出する処理である。

#### 【 0 0 5 6 】

図 1 3 ( a ) と図 1 3 ( e ) は、ケース 1 1 ( 図 9 ( k ) ) での処理に対応した図である。

#### 【 0 0 5 7 】

図 1 3 ( a ) は、中心画素とその左の画素が同ラベルで、中心画素の上の画素は中心画素と異なるラベルの場合に発生する処理を説明している。中心画素 ( 座標値が  $( 2 i , 2 j )$  であるとする ) の左上の位置 ( 座標値は  $( 2 i - 1 , 2 j - 1 )$  ) が色交点であるか否かを判定し、色交点であるときのみ、水平ベクトルの始点を抽出する。中心画素の左上を中心とした  $2 \times 2$  の 4 画素領域の画素パターンが、図 1 3 ( d ) に示す様に、中心画素とその左の画素以外の 2 画素が互いに異なるラベルで、かつ、中心画素の対角画素である左上の画素も中心画素とは異なるラベルである場合にのみ、左上の位置 ( 座標値は  $( 2 i - 1 , 2 j - 1 )$  ) に色交点である水平ベクトルの始点を抽出する。この水平ベクトルは、流入元は既走査領域にあるので確定され、流出先は未走査領域にあるので未定となる。なお、図 1 3 ( b ) は、中心画素の対角位置が中心画素のラベルとは異なるラベルではあるものの、中心画素の上の画素と同じラベルである場合を示している。また、図 1 3 ( c ) は、中心画素とその左の画素以外の 2 画素が互いに異なるラベルではあるものの、中心画素の対角位置の画素が中心画素と同じラベルである場合を示している。図 1 3 ( b ) や図 1 3 ( c ) の場合には、中心画素の左上の位置は色交点ではなく、輪郭点は抽出されない。

#### 【 0 0 5 8 】

図 1 3 ( e ) は、中心画素とその右の画素が同ラベルで、中心画素の下の画素は中心画素と異なるラベルの場合に発生する処理を説明している。中心画素 ( 座標値が  $( 2 i , 2 j )$  であるとする ) の右下の位置 ( 座標値は  $( 2 i + 1 , 2 j + 1 )$  ) が色交点であるか否かを判定し、色交点であるときのみ、水平ベクトルの始点を抽出する。中心画素の右下を中心とした  $2 \times 2$  の 4 画素領域の画素パターンが、図 1 3 ( h ) に示す様に、中心画素とその右の画素以外の 2 画素が互いに異なるラベルで、かつ、中心画素の対角画素である右下の画素も中心画素とは異なるラベルである場合にのみ、右下の位置 ( 座標値は  $( 2 i + 1 , 2 j + 1 )$  ) に色交点である水平ベクトルの始点を抽出する。この水平ベクトルは、流入元は未走査領域にあるので未定となり、流出先は既走査領域にあるので確定される。なお、図 1 3 ( f ) は、中心画素の対角位置が中心画素のラベルとは異なるラベルではあるものの、中心画素の下の画素と同じラベルである場合を示している。図 1 3 ( g ) は、中心画素とその右の画素以外の 2 画素が互いに異なるラベルではあるものの、中心画素の対角位置の画素が中心画素と同じラベルである場合を示している。図 1 3 ( f ) や図 1 3 ( g ) の場合には、中心画素の右下の位置は色交点ではなく、輪郭点は抽出されない。

#### 【 0 0 5 9 】

図 1 3 ( i ) と図 1 3 ( m ) は、ケース 6 ( 図 9 ( f ) ) での処理に対応した図である。

#### 【 0 0 6 0 】

図 1 3 ( i ) は、中心画素とその下の画素が同ラベルで、中心画素の左の画素は中心画素と異なるラベルの場合に発生する処理を説明している。中心画素 ( 座標値が  $( 2 i , 2 j )$  であるとする ) の左下の位置 ( 座標値は  $( 2 i - 1 , 2 j + 1 )$  ) が色交点であるか否かを判定し、色交点であるときのみ、垂直ベクトルの始点を抽出する。中心画素の左下を中心とした  $2 \times 2$  の 4 画素領域の画素パターンが、図 1 3 ( l ) に示す様に、中心画素とその下の画素以外の 2 画素が互いに異なるラベルで、かつ、中心画素の対角画素である左下の画素も中心画素とは異なるラベルである場合にのみ、左下の位置 ( 座標値は  $( 2 i - 1 , 2 j + 1 )$  ) に色交点である垂直ベクトルの始点を抽出する。この垂直ベクトルは

、流入元は未走査領域にあるので未定となり、流出先は既走査領域にあるので確定される。なお、図13(j)は、中心画素の対角位置が中心画素のラベルとは異なるラベルではあるものの、中心画素の左の画素と同じラベルである場合を示している。また、図13(k)は、中心画素とその下の画素以外の2画素が互いに異なるラベルではあるものの、中心画素の対角位置の画素が中心画素と同じラベルである場合を示している。図13(j)や図13(k)の場合には、中心画素の左下の位置は色交点ではなく、輪郭点は抽出されない。

#### 【0061】

図13(m)は、中心画素とその上の画素が同ラベルで、中心画素の右の画素は中心画素と異なるラベルの場合に発生する処理を説明している。中心画素(座標値が(2i, 2j)であるとする)の右上の位置(座標値は(2i+1, 2j-1))が色交点であるかを判定し、色交点であるときのみ、垂直ベクトルの始点を抽出する。中心画素の右上を中心とした2x2の4画素領域の画素パターンが、図13(p)に示す様に、中心画素とその上の画素以外の2画素が互いに異なるラベルで、かつ、中心画素の対角画素である右上の画素も中心画素とは異なるラベルである場合にのみ、右上の位置(座標値は(2i+1, 2j-1))に色交点である垂直ベクトルの始点を抽出する。この垂直ベクトルは、流入元は既走査領域にあるので確定され、流出先は未走査領域にあるので未定となる。尚、図13(n)は、中心画素の対角位置が中心画素のラベルとは異なるラベルではあるものの、中心画素の右の画素と同じラベルである場合を示している。図13(o)は、中心画素とその上の画素以外の2画素が互いに異なるラベルではあるものの、中心画素の対角位置の画素が中心画素と同じラベルである場合を示している。図13(n)や図13(o)の場合には、中心画素の右上の位置は色交点ではなく、輪郭点は抽出されない。

#### 【0062】

先にも述べたが、色交点と判断された輪郭点に対応するベクトル情報テーブル中の色交点フラグ欄には、TRUEを記録し、色交点ではない場合にFALSEを記録する。

#### 【0063】

尚、上述の図10は図9(p)に示したケース16の処理、また、上述の図11は図9(a)に示したケース1の処理、図13(a)と図13(e)は図9(k)に示したケース11の処理、図13(i)と図13(m)は図9(f)に示したケース6の処理に対応している。それ以外のケースは、これまで説明した、中心画素の左上、左下、右上、右下での、色交点であるか否かの判断を伴わない輪郭点の抽出処理、色交点判定を伴う輪郭点の抽出処理、色交点判定をして色交点の場合のみ輪郭点を抽出する処理の組合せで構成される。例えば、図9(b)に示したケース2の場合は、中心画素の左下での色交点判定を伴う輪郭点の抽出処理と、同右上での色交点判定をして色交点の場合のみ輪郭点を抽出する処理、同右下での色交点判定を伴う輪郭点の抽出処理で構成される。また、図9(d)に示したケース4では、中心画素の左下での色交点判定を伴う輪郭点の抽出処理と、同右上での色交点であるか否かの判断を伴わない輪郭点の抽出処理、同右下での色交点判定をして色交点の場合のみ輪郭点を抽出する処理で構成される。といった具合である。これ以外のケースに関しても、同様に処理を行えば構成可能であるため、これらに関する詳述は省略する。尚、各ケースにおいて構成に必要となる、中心画素の左上、左下、右上、右下での処理(ケースによって、存在しない位置もある)の順番は、ラスタ走査の順における未走査領域と抽出した輪郭点の流入元と流出先の扱いを一貫させる必要がある。このため、本実施例では、左上、左下、右上、右下(ケースにより、処理が存在しない位置は、左記の順を飛ばす)の順に行うものとする。

#### 【0064】

以上により、ステップS1200における色交点判定・輪郭点抽出処理の説明を終える。ステップS1200における、ベクトル情報テーブルや流入ベクトル未定水平ベクトルテーブル、流入ベクトル未定垂直ベクトルテーブル、流出ベクトル未定水平ベクトルテーブル、流出ベクトル未定垂直ベクトルテーブルは、RAM5上に不図示の領域として確保されるものである。なお、ステップS1200の処理は図1における色交点判定・輪郭点抽出手段103を

10

20

30

40

50

実現している。とりわけ、上記において、図 1 2 及び図 1 3 に基いて説明した色交点判定は、図 1 における色交点判定手段 1 0 3 1 を実現しており、図 1 0、図 1 1 及び図 1 3 に基いて説明した輪郭点抽出処理は、図 1 における輪郭点抽出手段 1 0 3 2 を実現している。

#### 【 0 0 6 5 】

ステップ S 1 2 0 0 での処理の流れを、図 1 4 と図 1 5 を用いて、具体例を元にして、さらに説明を加える。

#### 【 0 0 6 6 】

図 1 5 は上述した、流入ベクトル未定水平ベクトルテーブル、流出ベクトル未定水平ベクトルテーブル、流入ベクトル未定垂直ベクトルテーブル、流出ベクトル未定垂直ベクトルテーブルの 4 種のテーブルの状態を表している。それぞれのテーブルは、あらかじめ R A M 5 上に十分な領域が確保されているものとする。また、それぞれの領域は、それぞれ連続したメモリ領域として構成されていて、先頭領域から発生順にそれぞれのテーブルに該当する未定のベクトル番号を格納するものとする。それぞれの領域において、有効データの次である最終尾の位置には、最終尾であることを示すマーカーとして「 - 1 」を格納しておく。データを付け加えるには、このマーカーの位置に新たなベクトル番号を上書きし、その次の領域に改めてマーカーを書き込む。あるデータを削除する際には、削除するデータの次の有効データからマーカーまでを、削除するデータの位置から順に 1 つずつシフトするように書き写していくものとする。

#### 【 0 0 6 7 】

さて、図 1 4 ( a ) の 1 4 0 で示されるようなラベル画像にステップ S 1 2 0 0 の処理を行うと、色領域 1 ( ラベル番号が 1 の領域 ) に対しては、図 1 4 ( b ) で示されるベクトル情報テーブルが得られることになる。図 1 4 ( a ) における、A ( 0 )、B ( 1 )、C ( 2 )、D ( 3 )、E ( 4 )、F ( 5 ) の 6 点が、それぞれ、図 1 4 ( b ) のベクトル番号 ( ベクトルカウンタ欄の値 ) 0、1、2、3、4、5 の 6 本のベクトルに対応する。図 1 4 ( a ) のラベル画像 1 4 0 をラスタ走査すると、A ( 0 )、B ( 1 )、C ( 2 )、D ( 3 )、E ( 4 )、F ( 5 ) で示される位置の順に輪郭点が抽出されていくことになるからである。

#### 【 0 0 6 8 】

すなわち、ラベル画像の左上からラスタ走査を開始すると、3 × 3 の画素マトリクスのウィンドウが、図 1 4 ( a ) の 1 4 0 0 で示される位置に来た時、図 9 ( g ) に示したケース 7 の状態となる。このときの中心画素の左上の位置に交点判定を伴う輪郭点抽出処理が行われ、図 1 2 ( b ) で示した状態であることから、図 1 4 ( a ) における A ( 0 ) の位置に、ベクトル番号 0 として、色交点ではない輪郭点 ( 水平ベクトルの始点 ) が抽出される。このベクトルは、この時点では流入ベクトルも流出ベクトルも未定であるから、図 1 5 ( a ) に示す様に、流入ベクトル未定水平ベクトルテーブルと流出ベクトル未定水平ベクトルテーブルに、ベクトル番号 0 が追記されることになる。

#### 【 0 0 6 9 】

ラスタ走査を進めると、次に、図 1 4 ( a ) の 1 4 0 1 で示される位置に来た時、図 9 ( m ) に示したケース 1 3 の状態となる。このときの中心画素の右上の位置に交点判定を伴う輪郭点抽出処理が行われ、図 1 2 ( n ) で示した状態であることから、図 1 4 ( a ) における B ( 1 ) の位置に、ベクトル番号 1 として、色交点ではない輪郭点 ( 垂直ベクトルの始点 ) が抽出される。このベクトルは、流出ベクトルは未定ながら、流入ベクトルは既走査領域にあるので、流出ベクトル未定水平ベクトルテーブルを探索することにより、ベクトル番号 0 が流入ベクトルであることが判明する。よって、上述したように、ベクトル情報テーブルのベクトル番号 1 の流入ベクトル欄とベクトル番号 0 流出ベクトル欄の記載を行う。図 1 5 ( b ) に示す様に、流出ベクトル未定水平ベクトルテーブルからベクトル番号 0 を削除するとともに、流出ベクトル未定垂直ベクトルテーブルに、ベクトル番号 1 を追記する。

#### 【 0 0 7 0 】

次に、図 1 4 ( a ) の 1 4 0 2 で示される位置に来た時、図 9 ( f ) に示したケース 6 の状態となる。このときの中心画素の右上の位置に交点判定結果に基づく輪郭点抽出処理が行われ、図 1 3 ( p ) で示した状態であることから、図 1 4 ( a ) における C ( 2 ) の位置に、ベクトル番号 2 として、色交点である輪郭点 ( 垂直ベクトルの始点 ) が抽出される。このベクトルも、流出ベクトルは未定ながら、流入ベクトルは既走査領域にあるので、流出ベクトル未定垂直ベクトルテーブルを探索することにより、ベクトル番号 1 が流入ベクトルであることが判明する。よって、上述したように、ベクトル情報テーブルのベクトル番号 2 の流入ベクトル欄とベクトル番号 1 流出ベクトル欄の記載を行う。図 1 5 ( c ) に示す様に、流出ベクトル未定垂直ベクトルテーブルからベクトル番号 1 を削除した上で、ベクトル番号 2 を追記する。

10

**【 0 0 7 1 】**

次に、図 1 4 ( a ) の 1 4 0 3 で示される位置に来た時、図 9 ( d ) に示したケース 4 の状態となる。このときの中心画素の左下の位置に交点判定を伴う輪郭点抽出処理が行われ、図 1 2 ( h ) で示した状態であることから、図 1 4 ( a ) における D ( 3 ) の位置に、ベクトル番号 3 として、色交点ではない輪郭点 ( 垂直ベクトルの始点 ) が抽出される。このベクトルは、流入ベクトルは未定ながら、流出ベクトルは既走査領域にあるので、上述のように、流入ベクトル未定水平ベクトルテーブルと流入ベクトル未定垂直ベクトルテーブルの両方を探索することにより、ベクトル番号 0 が流出ベクトルであることが判明する。よって、上述したように、ベクトル情報テーブルのベクトル番号 3 の流出ベクトル欄とベクトル番号 0 流入ベクトル欄の記載を行う。図 1 5 ( d ) に示す様に、流入ベクトル未定水平ベクトルテーブルからベクトル番号 0 を削除するとともに、流入ベクトル未定垂直ベクトルテーブルに、ベクトル番号 3 を追記する。さらに、このケース 4 の状態では、中心画素の右下の位置に交点判定結果に基づく輪郭点抽出処理が行われ、図 1 3 ( h ) で示した状態であることから、図 1 4 ( a ) における E ( 4 ) の位置に、ベクトル番号 4 として、色交点である輪郭点 ( 水平ベクトルの始点 ) が抽出される。このベクトルは、流入ベクトルは未定ながら、流出ベクトルは既走査領域にあるので、上述のように、流入ベクトル未定水平ベクトルテーブルと流入ベクトル未定垂直ベクトルテーブルの両方を探索することにより、ベクトル番号 3 が流出ベクトルであることが判明する。よって、上述したように、ベクトル情報テーブルのベクトル番号 4 の流出ベクトル欄とベクトル番号 3 流入ベクトル欄の記載を行う。図 1 5 ( e ) に示す様に、流入ベクトル未定垂直ベクトルテーブルからベクトル番号 3 を削除するとともに、流入ベクトル未定水平ベクトルテーブルに、ベクトル番号 4 を追記する。

20

30

**【 0 0 7 2 】**

次に、図 1 4 ( a ) の 1 4 0 4 で示される位置に来た時、図 9 ( j ) に示したケース 1 0 の状態となる。このときの中心画素の右下の位置に交点判定を伴う輪郭点抽出処理が行われ、不図示ながら、図 1 2 ( u ) の様に、中心画素以外の 3 画素のうち、縦または横につながる 2 画素が同ラベルで、残りの 1 画素が中心画素とも他の 2 画素とも異なるラベルなので、色交点である輪郭点である。よって、図 1 4 ( a ) における F ( 5 ) の位置に、ベクトル番号 5 として、色交点である輪郭点 ( 水平ベクトルの始点 ) が抽出される。このベクトルは、流入ベクトルも流入ベクトルも既走査領域にある。まず、流出ベクトル未定垂直ベクトルテーブルを探索することにより、ベクトル番号 2 が流入ベクトルであることが判明する。よって、上述したように、ベクトル情報テーブルのベクトル番号 5 の流入ベクトル欄とベクトル番号 2 流出ベクトル欄の記載を行う。図 1 5 ( f ) に示す様に、流出ベクトル未定垂直ベクトルテーブルからベクトル番号 2 を削除する。そして、流入ベクトル未定水平ベクトルテーブルと流入ベクトル未定垂直ベクトルテーブルの両方を探索することにより、ベクトル番号 4 が流出ベクトルであることが判明する。図 1 5 ( f ) に示す様に、流入ベクトル未定水平ベクトルテーブルからベクトル番号 4 を削除する。

40

**【 0 0 7 3 】**

かくして、図 1 4 ( a ) の 1 4 0 で示されるようなラベル画像の色領域 1 ( ラベル番号が 1 の領域 ) に対して、図 1 4 ( b ) で示されるベクトル情報テーブルが得られることに

50



なる。

#### 【 0 0 7 4 】

さて、これまで述べてきた、ステップ S 1 2 0 0 の輪郭点・色交点抽出処理は、ラベル画像のラベル毎に順次にベクトル情報を抽出することが可能である。この方式では、ラベル画像中の全てのラベル領域の処理には、ラベル数に相当する回数分だけラベル画像を繰り返しラスタ走査し、上記の処理を繰り返すことになる。このとき、ラベル画像中で、各ラベル領域の存在する範囲をあらかじめ矩形領域として求めておき、ラベル毎に、それぞれの存在する矩形領域（ラベル画像の部分領域となる）のみにラスタ走査の範囲を限定すれば、処理時間を短縮することができる。

#### 【 0 0 7 5 】

一方で、ラベル画像を一回走査するだけで、ラベル画像中に出現する全てのラベルの輪郭ベクトル情報を抽出することも可能である。この場合は、ラベル画像中に出現する全てのラベル分だけ、各々のラベル領域用に別々にベクトル情報テーブルと流入ベクトルや流入ベクトルが未定のベクトルの情報を管理するテーブル類を用意する。その上で、ラスタ走査時に逐次、中心画素が属するラベルのテーブル類を使い分けて処理を進めるようにする。上述の走査を繰り返す場合に比し、ラベル数分の十分なメモリ容量を要すものの、一回のラスタ走査で、全てのラベルに対して、それぞれ別々の独立したベクトル情報テーブル中に、それぞれのラベル領域の輪郭ベクトルデータを抽出することができる。

#### 【 0 0 7 6 】

次にステップ S 1 3 0 0 では、ステップ S 1 2 0 0 で抽出したベクトル情報テーブル上のベクトル情報から、色領域（ラベル領域）を色交点間で区切られた部分輪郭の集まりで表現された輪郭情報に再構成する。

#### 【 0 0 7 7 】

ベクトル情報テーブルをベクトル番号の昇順に未使用フラグを参照しながら、未処理（未処理フラグが TRUE）かつ、色交点（色交点フラグが TRUE）であるベクトルを探し、該当するベクトルが見つかった場合には、このベクトルから処理を開始する。このベクトルを第 1 の区分輪郭線の始点とするともに第一の区分輪郭線の第 1 点目の輪郭点とし、当該ベクトルの未使用フラグを FALSE に変更する。次に、このベクトルの流出ベクトルを参照して、同区分輪郭線の次の輪郭点とする。もし、この輪郭点が色交点でなければ、当該ベクトルの未使用フラグを FALSE に変更し、このベクトルの流出ベクトルを参照して同様の処理を続ける。一方、同輪郭点が色交点（色交点フラグが TRUE）であった場合は、この輪郭点をもって同区分輪郭線の終了点とする。もし、この色交点の未使用フラグが既に FALSE となっていた場合には、抽出中の輪郭の開始座標に戻ったということであり、1 つの閉ループの抽出が完了となる。もし、この輪郭点の未使用フラグが未だ TRUE の場合には、この輪郭点を次の区分輪郭線の始点とし、この新しい区分曲線の第 1 点目の輪郭点とする。このベクトルの未使用フラグを FALSE に変更してから、前の区分輪郭線と同様に、流出ベクトルを参照して同区分輪郭線上の次の輪郭点を求める動作を続ける。1 つの閉ループの抽出が完了した場合には、再度、未処理（未処理フラグが TRUE）かつ、色交点（色交点フラグが TRUE）であるベクトルを探して残りの閉ループを探す。もし、テーブル内に色交点フラグが TRUE であるベクトルが見つからなかった場合は、テーブル内で最初に見つかる未処理のベクトルから処理を開始する。流出ベクトルのベクトル番号をたどり、出現する x y 座標を順に記録していけば輪郭座標を再構成することができる。そうして輪郭情報の再構成を続け、抽出中の輪郭の開始座標に戻れば 1 つの閉ループの抽出が完了となる。未処理のベクトルが無くなるまで処理を続けると外輪郭・内輪郭合わせた輪郭座標を全て抽出することができる。抽出された輪郭情報は、ラベル単位で情報を持つこととする。

#### 【 0 0 7 8 】

図 1 4 ( a ) から求められる図 1 4 ( b ) のベクトル情報テーブルを基に、ステップ S 1 3 0 0 の処理で輪郭情報を再構成すると、図 1 4 ( c ) のように、色領域（ラベル番号が 1 の領域）を色交点間で区切られた部分輪郭の集まりで表現した輪郭情報が得られる。

図 1 4 ( d ) は、図 1 4 ( c ) の輪郭情報が、ラベル番号 1 の領域の輪郭を構成することを示すデータ形式の例を示している。

【 0 0 7 9 】

このようにして抽出される色交点で区間分割された境界線（分割境界線）データを、区間ごとに関数近似することにより、それぞれの色領域の隣接箇所での重なり・隙間のないベクトルデータを生成することができる。

【 0 0 8 0 】

以上、ステップ S 1 3 0 0 の輪郭情報再構成処理を説明した。尚、このステップ S 1 3 0 0 の処理は、図 1 における輪郭情報再構成手段を実現している。

【 0 0 8 1 】

上述したように、本発明では、ラベリングされたラベル画像をラスター走査し、各ラベル領域の輪郭を構成する線素を、ウィンドウを用いて抽出する。抽出した線素は線素の始点となる座標情報とともに流出先・流出元を関連付けたうえでテーブルに登録しておく。また線素の始点が色領域間の色交点にあたる場合、色交点情報である旨も合わせて記録する。画像全体を走査し終えた後に、テーブルを参照して線素をつなぎ合わせる。この時、色交点情報を持つ箇所をつなぎ合わせるデータを分けることで境界線を抽出することができる。したがって、テーブルには線素が再構成されるべき順番で漏れなく登録されるため、抽出できる境界線も特定領域の輪郭再構成に適した順番で、漏れなく抽出することができる。1 回の画像全面に対するラスター走査とテーブル走査で境界線が抽出できるため、処理も高速である。また、線素の抽出と色交点判定はウィンドウをラスター走査順にずらしながら行うため、追跡による境界線抽出のように、領域形状によって追跡の方向が変わることがなく、メモリキャッシュが効きやすくなり処理速度の低下防止に効果がある。また、色領域の形状によらず、画像全体に対する 1 回のラスター走査とテーブル走査で境界線を抽出できる。

【 0 0 8 2 】

〔 実施例 2 〕

ウィンドウのサイズは  $3 \times 3$  でなくてもよく、 $2 \times 2$  でも可能である。 $2 \times 2$  のウィンドウ走査で起こりえる  $2 \times 2$  内の 4 画素のパターンを図 1 6 ( a )、図 1 6 ( d )、図 1 6 ( g )、図 1 6 ( k )、図 1 6 ( o ) に示した。図 1 6 ( a ) と図 1 6 ( d ) は、4 画素内に 2 種類のラベルの画素が存在する場合である。図 1 6 ( g )、図 1 6 ( k ) は、4 画素内に 3 種類のラベルの画素が存在する場合である。図 1 6 ( o ) は、4 画素内に 4 種類のラベルの画素が存在する場合である。尚、4 画素内に 1 種類のラベルの画素しか存在しない場合には、輪郭点は抽出されることがないので、表記していない。基本的には、これらのパターンの  $90^\circ$ 、 $180^\circ$ 、 $270^\circ$  の回転したパターンがあり得るが、図 1 6 ( a )、図 1 6 ( d )、図 1 6 ( g )、図 1 6 ( k )、図 1 6 ( o ) の場合を示せば、それらの回転パターンの場合は、容易に構成可能であるため、これらに関する詳述は省略する。以下、図 1 6 ( a )、図 1 6 ( d )、図 1 6 ( g )、図 1 6 ( k )、図 1 6 ( o ) の場合を説明する。

【 0 0 8 3 】

実施例 1 の  $3 \times 3$  のウィンドウ走査での説明の中で、ラベル画像を一回走査するだけで、ラベル画像中に出現する全てのラベルの輪郭ベクトル情報を抽出する場合を述べた。その中で、ラベル画像中に出現する全てのラベル分だけ、各々のラベル領域用に別々にベクトル情報テーブルと流入ベクトルや流入ベクトルが未定のベクトルの情報を管理するテーブル類を用意することを説明した。 $2 \times 2$  のウィンドウ走査においても、同様である。

【 0 0 8 4 】

さて、図 1 6 ( a ) のパターンの場合には、図 1 6 ( b ) と図 1 6 ( c ) に示したように、2 種類のラベルのそれぞれに、一点ずつ、色交点ではない輪郭点を抽出する。図 1 6 ( d ) のパターンの場合には、図 1 6 ( e ) と図 1 6 ( f ) で示したように、2 種類のラベルのそれぞれに、2 点ずつ、色交点である輪郭点を抽出する。図 1 6 ( g ) のパターンの場合には、図 1 6 ( h )、図 1 6 ( i )、図 1 6 ( j ) で示したように、3 種類のラベ

ルのそれぞれに対応させて、それぞれに色交点である輪郭点を1点ずつ抽出する。図16(k)のパターンの場合には、図16(l)、図16(m)、図16(n)で示したように、3種類のラベルのうち、4画素中に一画素しか存在しないラベルに対しては、色交点ではない輪郭点を1点ずつ抽出し(図16(l)と図16(m)に相当)、4画素中の斜め方向に隣接している2画素に同じラベルが存在しているラベル(図16(n)に相当)には、色交点ではない輪郭点を2点抽出する。また、図16(o)のパターンの場合には、図16(p)、図16(q)、図16(r)、図16(s)で示したように、のそれぞれに対応させて、それぞれに色交点である輪郭点を1点ずつ抽出する。

【0085】

このように、構成すれば、 $2 \times 2$ のウィンドウ走査でも、 $3 \times 3$ のウィンドウ走査で得られる輪郭情報と全く同等な輪郭情報を抽出することが可能である。

10

【0086】

一回のラスタ走査で、ラベル画像中の全てのラベル領域の輪郭情報を抽出するためには、 $3 \times 3$ の場合には、実施例1で述べた様に、中心画素のラベルについてのみを抽出すればよい。これに対し、 $2 \times 2$ の場合には、ウィンドウ内4画素内に出現するラベル全てについての上を抽出しながら行う必要がある点が大きく異なる。

【0087】

また、 $2 \times 2$ の $2 \times 2$ のウィンドウ走査の場合でも、一回の走査では、上記の輪郭点抽出のうち、ある一つのラベルに対応する輪郭点のみを該当する一つのラベルのテーブル領域のみに抽出する様に構成してすることも可能である。画像中に存在するラベルの数の回数、輪郭抽出するラベルを切り替えながら、同様なラスタ走査を繰り返せば、 $2 \times 2$ のウィンドウ走査によっても、ラベル毎の輪郭情報抽出が可能である。この場合には、ベクトル情報テーブルと流入ベクトルや流入ベクトルが未定のベクトルの情報を管理するテーブル類用のメモリ領域をあらかじめ全てのラベル分確保する必要はなくなる。

20

【0088】

(その他の実施例)

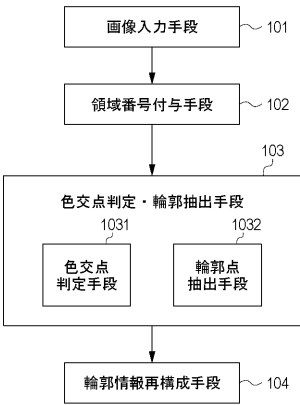
以上本発明にかかる実施形態を説明したが、実施形態の処理の多くは、情報処理装置上で実行されるコンピュータプログラムによるものであるので、当然、本発明はかかるコンピュータプログラムをもその範疇とする。通常、コンピュータプログラムは5のRAMや6のROM、またはCD-ROM等のコンピュータ可読記憶媒体に格納されていて、それをコンピュータにセットし、システムにコピーもしくはインストールすることで実行可能になる。従って、かかるコンピュータ可読記憶媒体も当然に本発明の範疇に入る。

30

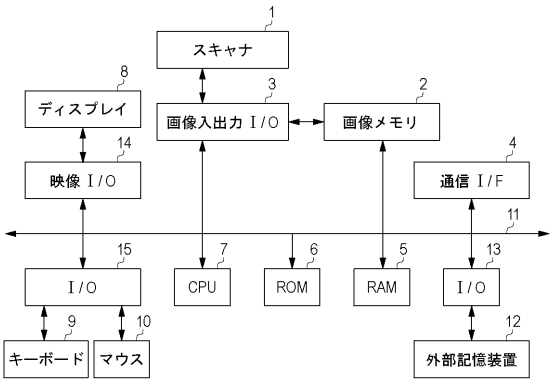
【0089】

また、本発明は、以下の処理を実行することによっても実現される。その処理は、上述した実施例の機能を実現させるソフトウェア(プログラム)を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ(またはCPUやMPU等)がプログラムを読み出して実行する処理である。

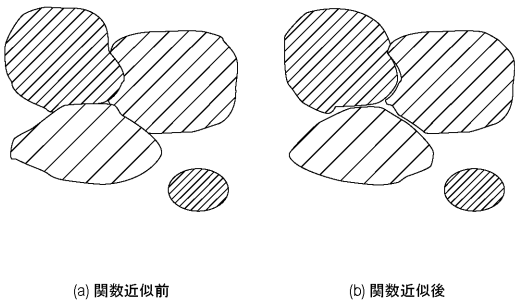
【図 1】



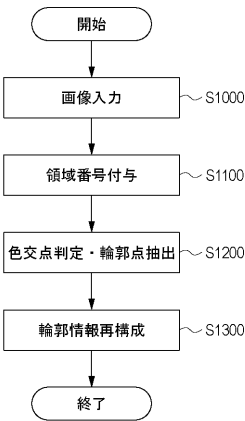
【図 2】



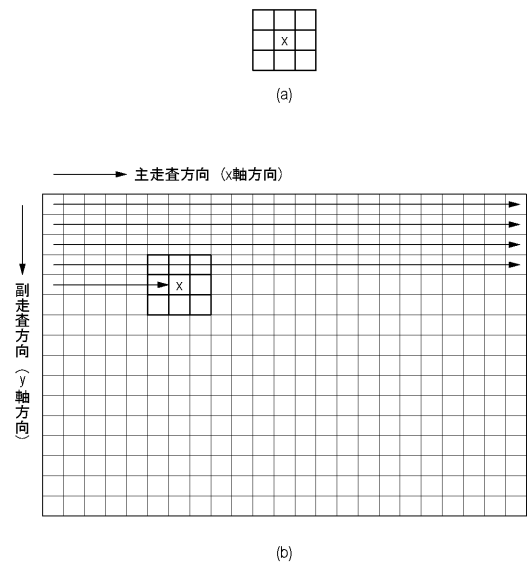
【図 3】



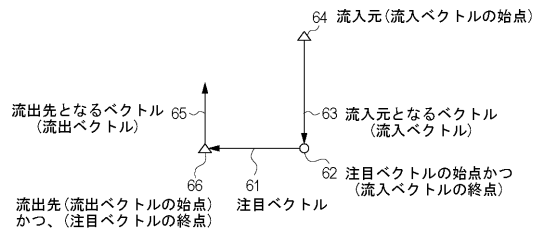
【図 4】



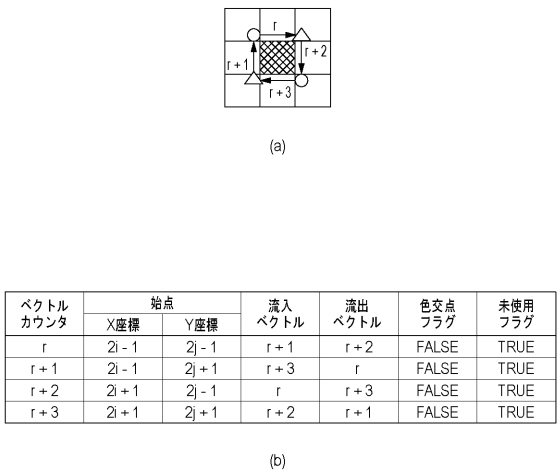
【 図 5 】



【 図 6 】



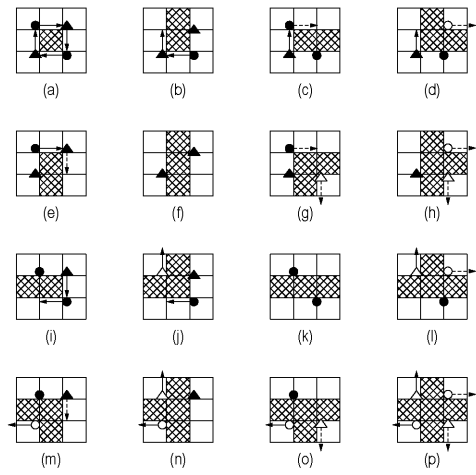
【 図 7 】



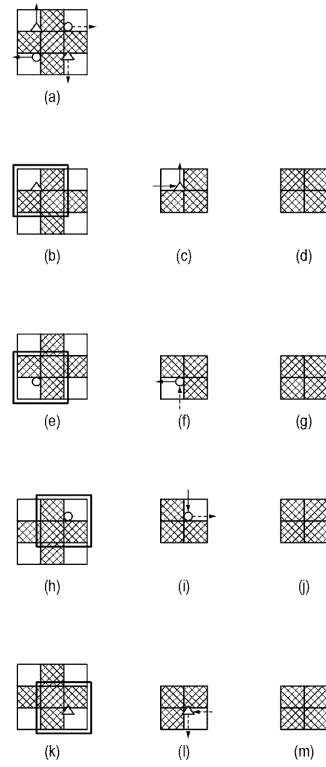
【 図 8 】

	P <sub>1</sub>	
P <sub>4</sub>	P <sub>0</sub>	P <sub>2</sub>
	P <sub>3</sub>	

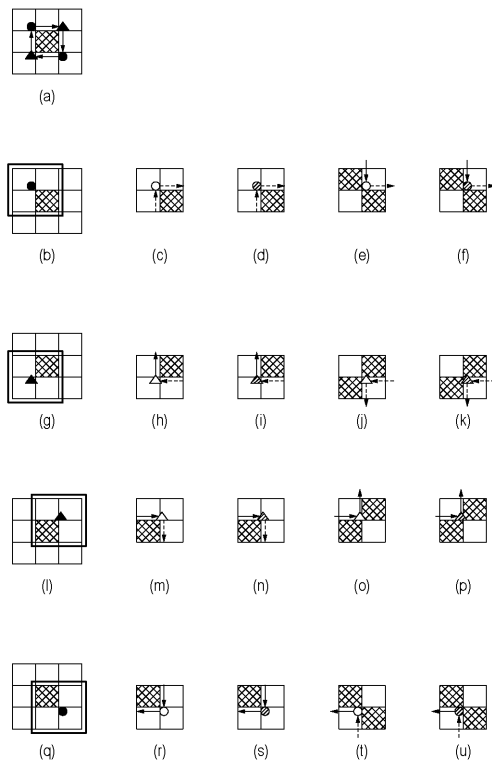
【図 9】



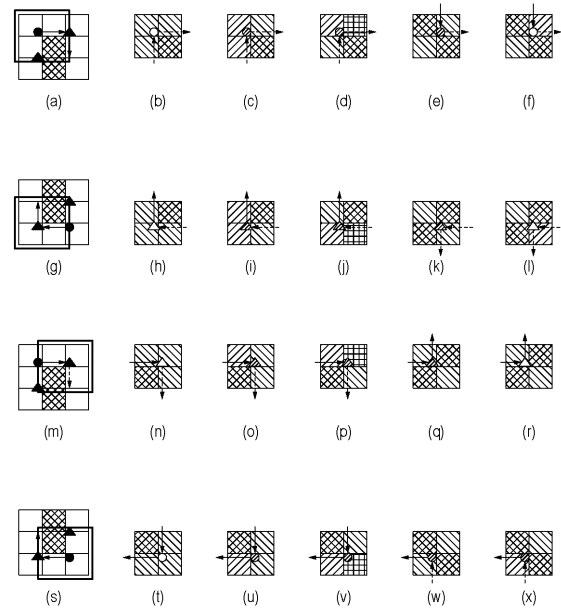
【図 10】



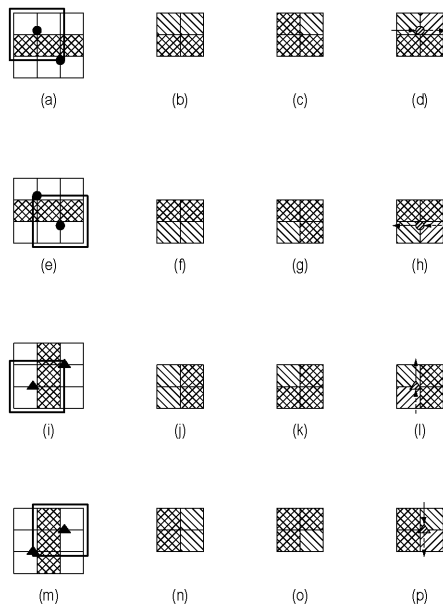
【図 11】



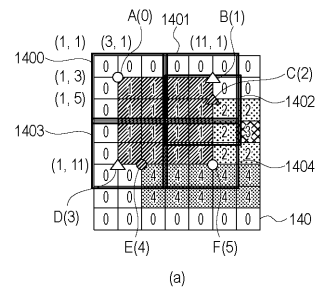
【図 12】



【図 13】



【図 14】



ベクトル カウンタ	始点		流入 ベクトル	流出 ベクトル	色交点 フラグ	未使用 フラグ
	X座標	Y座標				
0	3	3	3	1	FALSE	TRUE
1	11	3	0	2	FALSE	TRUE
2	11	5	1	5	TRUE	TRUE
3	3	11	4	0	FALSE	TRUE
4	5	11	5	3	TRUE	TRUE
5	11	11	2	4	TRUE	TRUE

(b)

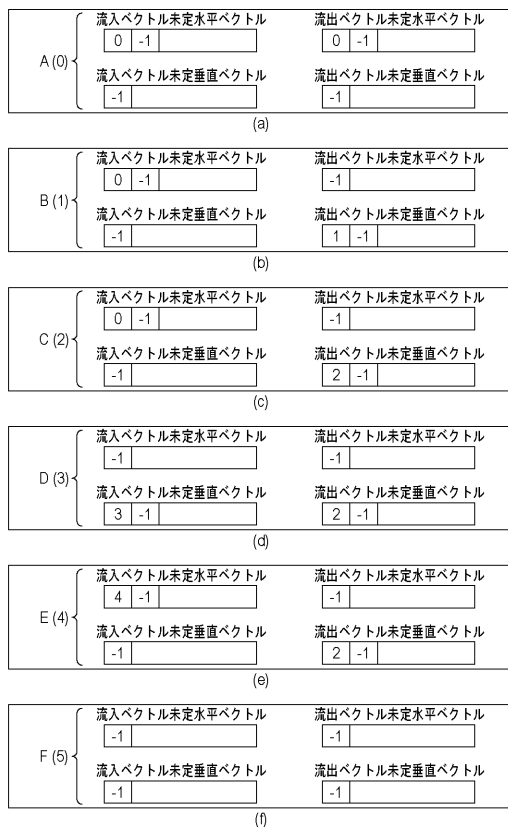
線番号	端点座標		座標点列
	開始点	終了点	
1	(11,5)	(11,11)	(11,5), (11,11)
2	(11,11)	(5,11)	(11,11), (11,5)
3	(5,11)	(11,5)	(5,11), (3,11), (3,3), (11,3), (11,5)

(c)

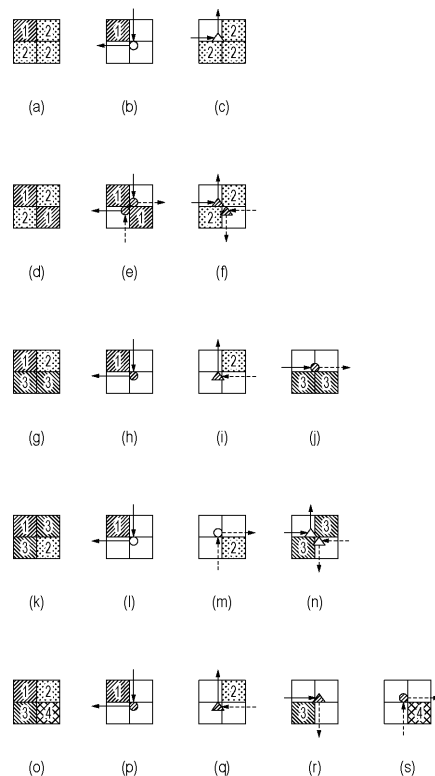
領域番号	線番号
1	線1, 線2, 線3,

(d)

【図 15】



【図 16】



---

フロントページの続き

(56)参考文献 特開2008-146496(JP,A)  
特開平04-157578(JP,A)  
特開平08-007114(JP,A)

(58)調査した分野(Int.Cl., DB名)  
G06T 7/60