

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4587832号  
(P4587832)

(45) 発行日 平成22年11月24日(2010.11.24)

(24) 登録日 平成22年9月17日(2010.9.17)

(51) Int.Cl.		F I			
<b>G06T</b>	<b>7/00</b>	<b>(2006.01)</b>	G06T	7/00	100C
<b>G06T</b>	<b>1/00</b>	<b>(2006.01)</b>	G06T	1/00	510
<b>G06T</b>	<b>1/20</b>	<b>(2006.01)</b>	G06T	1/20	C
<b>H04N</b>	<b>1/41</b>	<b>(2006.01)</b>	H04N	1/41	C

請求項の数 10 (全 19 頁)

(21) 出願番号	特願2005-36116 (P2005-36116)	(73) 特許権者	596170170
(22) 出願日	平成17年2月14日 (2005.2.14)		ゼロックス コーポレイション
(65) 公開番号	特開2005-228341 (P2005-228341A)		XEROX CORPORATION
(43) 公開日	平成17年8月25日 (2005.8.25)		アメリカ合衆国、コネチカット州 068
審査請求日	平成20年2月12日 (2008.2.12)		56、ノーウォーク、ピーオーボックス
(31) 優先権主張番号	10/776602		4505、グローバー・アヴェニュー 4
(32) 優先日	平成16年2月12日 (2004.2.12)		5
(33) 優先権主張国	米国 (US)	(74) 代理人	100079049
			弁理士 中島 淳
		(74) 代理人	100084995
			弁理士 加藤 和詳
		(72) 発明者	ドナルド ジェイ. カリー
			アメリカ合衆国 94025 カリフォル
			ニア州 メンロ パーク リーランド ア
			ベニュー 333

最終頁に続く

(54) 【発明の名称】 画像データ内で識別された領域を編成する方法及び装置

(57) 【特許請求の範囲】

【請求項1】

画像データ内で識別された領域を編成する方法であって、

前記画像データによって表される画像領域における類似のカラー値を有するピクセルを含むように閉曲線境界で定義されたプロブが下記(1)~(6)の何れかの条件を満足しているか否かを判定する判定ステップと、

(1) 大きさが処理対象プロブの大きさとして予め定められた最小の大きさ未満である、との条件

(2) ピクセル密度が予め定められた密度未満である、との条件

(3) 前記プロブを囲む最小の長方形からなる境界ボックスの縦横比が所定範囲内である、との条件

(4) 前記境界ボックスの縦の長さ又は横の長さが予め定められた長さを超えている、との条件

(5) 大きさが処理対象プロブの大きさとして予め定められた最大の大きさを超え、かつ平均輝度値が白色の輝度値に近似している、との条件

(6) 上記(1)~(5)に示す条件の何れも満足せず、かつカラー値の変動量が所定限度量を超えている、との条件

前記判定ステップで前記(1)~(6)の何れかの条件を満足したと判定された前記プロブにマークを割り当てるマーク割当ステップと、

互いのカラー値が類似した複数の前記プロブをグループ化したカラークラスタを作成す

10

20

るカラークラスタ作成ステップと、

前記カラークラスタ作成ステップで作成されたカラークラスタ毎に、隣接している前記プロブ同士を結合した空間クラスタを作成する空間クラスタ作成ステップと、

前記空間クラスタ作成ステップで作成された前記空間クラスタであって前記マーク割当ステップで前記マークが割り当てられた前記プロブが含まれる前記空間クラスタを除去するクラスタ除去ステップと、

を有する方法。

【請求項 2】

前記プロブ毎に、前記境界ボックスを拡張して得られた拡張境界ボックスを設け、

前記空間クラスタ作成ステップは、前記拡張境界ボックスがオーバーラップする前記プロブ同士を結合することにより前記空間クラスタを作成する請求項 1 記載の方法。

10

【請求項 3】

前記画像領域を複数の矩形領域に分割する分割ステップと、

前記クラスタ除去ステップで前記空間クラスタが除去された後、残存している前記空間クラスタを個別に囲む最小の長方形からなるクラスタ用境界ボックスを拡張して得られたクラスタ用拡張境界ボックスが該空間クラスタ間で前記分割ステップで分割して得られた矩形領域を跨いでオーバーラップする前記空間クラスタ同士を結合するクラスタ結合ステップと、を更に有する請求項 1 または請求項 2 記載の方法。

【請求項 4】

前記画像データにおける前記画像領域を構成する複数のピクセルの各々に対して、前記プロブ内でのみ一意となる ID を割り当てる ID 割当ステップと、

前記クラスタ結合ステップで前記空間クラスタ同士が結合された後、前記 ID 割当ステップで割り当てられた前記 ID に基づいて前記プロブに内包されている前記プロブを検出する検出ステップと、

前記検出ステップで検出されたプロブを除去するプロブ除去ステップと、を更に有する請求項 3 記載の方法。

20

【請求項 5】

前記 (6) の条件を、上記 (1) ~ (5) に示す条件の何れも満足せず、かつ前記プロブを横断するライン上に位置するピクセルのカラー値の変動量が所定限度量を超えているとの条件とした請求項 1 ~ 請求項 5 の何れか 1 項に記載の方法。

30

【請求項 6】

画像データ内で識別された領域を編成する装置であって、

前記画像データによって表される画像領域における類似のカラー値を有するピクセルを含むように閉曲線境界で定義されたプロブが下記 (1) ~ (6) の何れかの条件を満足しているか否かを判定する判定手段と、

(1) 大きさが処理対象プロブの大きさとして予め定められた最小の大きさ未満である、との条件

(2) ピクセル密度が予め定められた密度未満である、との条件

(3) 前記プロブを囲む最小の長方形からなる境界ボックスの縦横比が所定範囲内である、との条件

40

(4) 前記境界ボックスの縦の長さ又は横の長さが予め定められた長さを超えている、との条件

(5) 大きさが処理対象プロブの大きさとして予め定められた最大の大きさを超え、かつ平均輝度値が白色の輝度値に近似している、との条件

(6) 上記 (1) ~ (5) に示す条件の何れも満足せず、かつカラー値の変動量が所定限度量を超えている、との条件

前記判定手段で前記 (1) ~ (6) の何れかの条件を満足したと判定された前記プロブにマークを割り当てるマーク割当手段と、

互いのカラー値が類似した複数の前記プロブをグループ化したカラークラスタを作成するカラークラスタ作成手段と、

50

前記カラークラスタ作成手段で作成されたカラークラスタ毎に、隣接している前記プロブ同士を結合した空間クラスタを作成する空間クラスタ作成手段と、

前記空間クラスタ作成手段で作成された前記空間クラスタであって前記マーク割当ステップで前記マークが割り当てられた前記プロブが含まれる前記空間クラスタを除去するクラスタ除去手段と、

を有する装置。

【請求項 7】

前記プロブ毎に、前記境界ボックスを拡張して得られた拡張境界ボックスを設け、前記空間クラスタ作成手段は、前記拡張境界ボックスがオーバーラップする前記プロブ同士を結合することにより前記空間クラスタを作成する請求項 6 記載の装置。

10

【請求項 8】

前記画像領域を複数の矩形領域に分割する分割手段と、前記クラスタ除去手段で前記空間クラスタが除去された後、残存している前記空間クラスタを個別に囲む最小の長方形からなるクラスタ用境界ボックスを拡張して得られたクラスタ用拡張境界ボックスが該空間クラスタ間で前記分割ステップで分割して得られた矩形領域を跨いでオーバーラップする前記空間クラスタ同士を結合するクラスタ結合手段と、を更に有する請求項 6 または請求項 7 記載の装置。

【請求項 9】

前記画像データにおける前記画像領域を構成する複数のピクセルの各々に対して、前記プロブ内でのみ一意となる ID を割り当てる ID 割当手段と、

20

前記クラスタ結合手段で前記空間クラスタ同士が結合された後、前記 ID 割当ステップで割り当てられた前記 ID に基づいて前記プロブに内包されている前記プロブを検出する検出手段と、

前記検出手段で検出されたプロブを除去するプロブ除去手段と、を更に有する請求項 8 記載の装置。

【請求項 10】

前記 (6) の条件を、上記 (1) ~ (5) に示す条件の何れも満足せず、かつ前記プロブを横断するライン上に位置するピクセルのカラー値の変動量が所定限度量を超えているとの条件とした請求項 6 ~ 請求項 9 の何れか 1 項に記載の装置。

【発明の詳細な説明】

30

【技術分野】

【0001】

本発明は、類似のロケーションおよびカラー（色）特性を有する領域へ画像データを編成することに関する。

【背景技術】

【0002】

高解像度で走査（スキャン）される文書は、一般的には、非常に大量の記憶空間を必要とする。更に、サイズの大きい画像データは、たとえば、ローカル通信網（LAN）、広域通信網（WAN）、イントラネット、エクストラネット、インターネット、または他の分散ネットワークを介して転送する等の処理するのに相当多くの時間および帯域幅を必要とする。そのため、画像データはデータ量を削減するために、一般的に何らかの方法で圧縮処理が施される。

40

【0003】

しかしながら、従来の、例えば JPEG 形式のような圧縮方法は、例えばバイナリテキストや線画像等の、鋭い輪郭を有したりカラー遷移の度合いが大きい、高空間周波数を有する画像の場合、一般にうまく動作しないことが多い。

【発明の開示】

【発明が解決しようとする課題】

【0004】

カラーおよび/またはグレースケール情報が混在するテキストを有する画像を高品質で

50

レンダリングし、出力画像のカラー変動を許容範囲におさめるために画像を編成するシステムおよび方法を提供する。

【課題を解決するための手段】

【0005】

画像データの圧縮の必要性を満足させる新しい方法は、データを記述するのにMRC (Mixed Raster Content) 形式を使用するエンコーダパイプラインを使用することである。画像データ、たとえば、カラーおよび/またはグレースケール情報が混在するテキストを有する合成画像を定義する画像データは、2つ以上の平面へ分割される。これらの平面は、一般的に、背景(バックグラウンド)平面および前景(フォアグラウンド)平面と呼ばれる。合成画像の中の各々のピクセルについて、どの画像平面が、最終出力画像を再構成するために使用されるべき実際の画像データを含むかを示すため、セクタ平面が生成される。

10

【0006】

複数のバイナリ前景平面を有するMRC形式へ文書を変換するシステムおよび方法が提供される。文書を走査した後、境界連続性および/またはカラー連続性を使用して、領域またはプロブ(blob)を識別するため画像データが解析される。各々のピクセルは、プロブの境界長方形の内部でのみ一意であることを要求されるIDを割り当てられ、境界長方形およびピクセルIDによってプロブを識別すると共に他の属性をリストするテーブルが構成される。プロブを並べ替えおよびフィルタリングし、類似の位置およびカラー特性を有するプロブをクラスタにまとめるため、テーブルが解析される。次にクラスタは、複数のバイナリ前景層(レイヤ)の1つへ配置される。バイナリ前景層の各々は、その層へ割り当てられた全てのプロブの平均カラーに近似するカラー値を割り当てられる。このようにして、高品質の画像をレンダリングし、出力画像のカラー変動を許容できるものにすることができる。

20

【発明の効果】

【0007】

本発明により、カラーおよび/またはグレースケール情報が混在するテキストを有する画像を高品質でレンダリングし、出力画像のカラー変動を許容範囲におさめるために画像を編成するシステムを実現することができる。

【発明を実施するための最良の形態】

30

【0008】

本発明は、図面を参照して詳細に説明される。

【0009】

走査および/または印刷されたカラー文書を自動的に処理して、元の文書内容を正確にとらえる小さく高度に圧縮された画像データファイルを生成するシステムおよび方法の様々な例示的实施形態が開示される。文書の中を走査し、データを解析し、どのピクセルが類似のカラーを有し閉曲線境界により定義され、そのためグループにまとめることができるかを判断することによって、データファイルが生成される。次に、これらの区域は、空間的に相互に近接した類似カラーの区域を結合することによって、クラスタにまとめられる。次に、結合された区域(クラスタ化された領域)は、複数のバイナリ前景層の1つに配置され、たとえばPDFファイルとして出力するため圧縮されてよい。

40

【0010】

図1は、背景の連続色調またはグレースケール層(背景層)210およびN個のバイナリ前景層へ分解される文書の画像データを示す。この例において、6つの(N=6)バイナリ前景層220~270が存在する。画像データは、6つの区別できるカラー値へ分離される低空間周波数カラーデータを含む。6つのカラー値の各々は、多数のバイナリ前景層220~270の特定の1つに関連づけられ、層220~270は、それぞれ6つのカラーの1つを含む空間的領域範囲220'~270'を含む。領域220'~270'のいずれにも含まれないデータは、背景層210の中に留まる。以下、バイナリ前景層へ割り当てられる空間領域220'~270'を生成するシステムおよび方法を説明する。

50

## 【 0 0 1 1 】

画像データの中のピクセルは、「プロブ (blob)」に属するものとして識別される。「プロブ」とは、閉曲線境界によって定義され、類似のカラー値を有するピクセルを含む領域である。非一意の小さな ID を全てのピクセルに割り当て、ID を割り当てられるプロブの境界長方形の内部でのみ ID が一意となるように、画像データのマップが作られる。プロブ ID 画像の例は図 2 に示される。「1」のプロブ ID を有するプロブが、文字「A」の概略的形状を有するように示される。プロブを記述するために有用なパラメータは、プロブを完全に囲む最小の長方形であるプロブ境界ボックスである。x 次元の最小値である MIN\_X 1、x 次元の最大値である MAX\_X 1、y 次元の最小値である MIN\_Y 1、および y 次元の最大値である MAX\_Y 1 は、図 2 のプロブ # 1 についてプロブ境界ボックスの範囲を定義する。

10

## 【 0 0 1 2 】

プロブ # 2 は、MIN\_X 2、MAX\_X 2 (図示省略)、MIN\_Y 2、および MAX\_Y 2 によって境界を定められるもう 1 つのプロブである。これらの 2 つのプロブは、プロブクラスタ処理モジュールによってクラスタにまとめられる。次に、クラスタは、以下で詳細に説明するように、N 個のバイナリ前景層の 1 つへ割り当てられる。

## 【 0 0 1 3 】

図 3 は、走査された文書画像から N 個のバイナリ前景層を生成する例示的システムを示す。図 3 は、たとえば RGB 色空間で画像データを生成するためスキャナ 400 によって走査される文書 300 を示し、画像データは、N 層画像データ生成システム 1000 へ入力される。次に、N 層画像データ生成システム 1000 によって生成された N 個の層は、圧縮モジュール 800 によって圧縮され、たとえば PDF ファイルとして出力される。

20

## 【 0 0 1 4 】

N 層画像データ生成システム 1000 の最初のモジュールは、カラー変換モジュール 500 である。カラー変換モジュール 500 は、たとえば、RGB データを YCC 色空間へ変換する。

## 【 0 0 1 5 】

次に、YCC 色空間の画像は、画像に現れる領域またはプロブのリストを生成するプロブ識別モジュール 600 へ入力される。

## 【 0 0 1 6 】

プロブ識別モジュール 600 は、更に、文書画像 280 の中にある全てのプロブのリストであるグローバルプロブテーブル (GTB) を生成する。GTB の 1 つのエントリ (登録データ) が、図 4 に示される。GTB は、プロブと同じ数のプロブエントリを含む。各々の GTB エントリの最初の 4 つのフィールド 310 ~ 340 は、プロブに関連づけられた境界ボックスの座標である。

30

## 【 0 0 1 7 】

SUM\_Y (輝度) 350 および彩度値である SUM\_Cb 360 および SUM\_Cr 370 は、プロブへ割り当てられたカラー合計 (sum) であり、トータルピクセルカウント 380 は、所与のプロブへ割り当てられたプロブ ID 画像のピクセル総数である。任意設定項目である VAR (分散値) 390 は、プロブの彩度値の分散であり、ID フィールド 410 は、マップ画像で使用された (非一意の) プロブ ID である。インデックスフィールド 420 は、任意設定項目の一意のプロブ識別子である。各々の GTB エントリに含まれるマークフィールド 430 は、以下で更に説明するように、プロブが一組のテスト規準をパスしたかどうかに従って、プロブクラスタ処理モジュールによってプロブへ適用されたマークを記憶する。プロブネクスト (NEXT) フィールド 440 および内容 (CONTENT) フィールド 450 は、現在のプロブがリンクされるプロブを指し示すポインタを記憶する。

40

## 【 0 0 1 8 】

更に、走査された YCC 変換画像である向上カラーデータ (ENH: Enhanced Color Data) が、プロブ識別モジュールによって生成される。

50

## 【 0 0 1 9 】

プロブクラスタ処理モジュール700は、プロブIDマップ画像、GTB、およびENHを入力させ、グローバルプロブテーブルを解析し、或るカラーおよび空間特性を共有するプロブをN個のクラスタ領域へ結合する。プロブクラスタ処理モジュール700は、クラスタ領域のカラー値に依存して、クラスタ領域を(N+1)個のバイナリ前景層の1つへ割り当てる。次に、層は、たとえばPDF形式で出力される前に、圧縮モジュール800によって圧縮される。

## 【 0 0 2 0 】

図5は、プロブクラスタ処理(BCP)モジュール700の例示的实施形態を示す。モジュール700は、CPU710、入/出力ポート780、メモリ770、クラスタ構築モジュール720、クラスタ並べ替えモジュール730、プロブマークモジュール740、層(平面)生成モジュール750、クラスタ結合モジュール705、および内部プロブモジュール760を含む。

10

## 【 0 0 2 1 】

プロブクラスタ処理モジュール700は、入/出力ポート780を介して、プロブ識別モジュール600からENH、プロブIDマップデータ(BID)、およびGTBを入力させ、N+1個のバイナリ前景層を生成する。

## 【 0 0 2 2 】

プロブ識別モジュール600からGTBを受け取ると、クラスタ並べ替えモジュール730は、プロブを並べ替え、後続のプロブクラスタ処理段階で便利な順序にプロブを配置する。

20

## 【 0 0 2 3 】

クラスタ並べ替えモジュール730は、更に、カラー合計をピクセルカウントで割り、平均のプロブカラーを得るために、GTBのカラー合計を正規化する。

## 【 0 0 2 4 】

次に、並べ替えモジュール730は、GTBをプロブマークモジュール740へ引き渡す。プロブマークモジュール740は、GTBの各々のプロブを解析し、多数のテスト条件の結果に従って、マークフィールド430へマークを適用する。プロブをマークする目的は、プロブの数を減らし、結果的に処理負荷を減らすため、下流のプロセス(処理)によってどのプロブを除去できるかを示すことである。こうして、マークフィールド430は、プロセスの負荷を減らすためマークプロセスによって使用される。

30

## 【 0 0 2 5 】

プロブマークモジュール740は、5つの異なる条件に従ってプロブをテストする。もし条件が満たされるならば、マークフィールドのビットは、プロブをマークするために変更される。各々の条件は、マークフィールドで異なるビットパターンを生じる。プロセスを早めるため、プロブ形状自体の代わりに境界ボックスに基づく簡単なテストが使用される。プロブマークモジュール740の条件は、次の順序で適用される。

## 【 0 0 2 6 】

1) プロブ領域が小さすぎるかどうかをチェックする。

## 【 0 0 2 7 】

2) ピクセルカウント密度が、所定の最小密度よりも小さいかどうかをチェックする。

40

## 【 0 0 2 8 】

3) 境界ボックスの縦横比(幅/高さ)が大きすぎるかどうか、または小さすぎるかどうかをチェックする。この条件は、プロブが線であることを示しうる。

## 【 0 0 2 9 】

4) 境界ボックスの幅または高さが大きすぎるかどうかをチェックする。

## 【 0 0 3 0 】

5) プロブ領域が十分に大きいかどうか、および平均輝度が白に近いかどうかをチェックする。

完了したとき、マークフィールド430は、満たされた最後のテストで生成されたコード

50

を含む。プロブは、5つのテストの全てが失敗したときにのみマークされないままとなる（コード0）。

【0031】

プロブマークモジュール740は、プロブを除去または変更せず、1つ以上のテスト条件が満たされたとき、マークフィールドを変更するだけである。

【0032】

プロブのピクセルの全部または大部分が、カラー変動テストに使用されてよいが、カラー変動テストのスピードを増加するため、図6で示されるように、1つの線、たとえば中央線の上に位置するピクセルのみがテストされてよい。プロブマークモジュール740は、境界ボックスによって判断される左側の最初のプロブピクセルから始まり右側の最後の  
10  
プロブピクセルで終わる中央線上のプロブピクセルを評価する。次に、プロブマークモジュール740は、中央線上のプロブピクセルのカラー変動を検査する。

【0033】

プロブマークモジュール740は、現在の「X」ピクセルカラーと、Cb、Crカラー合計によって与えられる平均プロブ彩度との間の彩度差分を測定することによって、カラー変動を計算する。

【0034】

次に、プロブマークモジュール740は、計算された彩度変動を所定の限度と比較する。もし彩度変動が、この限度を超過すれば、それに従ってプロブがマークされる。

【0035】

プロブがマークされた後、それらのプロブは、類似のカラー特性を有し近接したロケーション（位置）にあるプロブが結合されてクラスタにされる。クラスタ構築モジュール720は、並べ替えおよびマークされたプロブのカラーおよび空間上での関係に基づいて、  
20  
プロブをクラスタへ編成する。

【0036】

クラスタ構築モジュール720は、画像区域を隣接する互いに重なり合わない（オーバーラップしない）正方形タイルへ分割することから開始する。タイルの目的は、評価されている画像区域を、より管理可能なサイズへ縮小することである。画像区域とタイルの関係は図7で示される。

【0037】

クラスタ構築モジュール720は、カラーおよび空間ロケーションによってプロブをグループ化し、プロブのカラークラスタおよび空間クラスタを形成する。各々のカラークラスタは、類似したカラーを有する1つ以上のプロブを含む。即ち、その各々の領域（プロブ）のカラーの違いは、カラー閾値よりも小さい。各々のカラークラスタは、少なくとも1つの空間クラスタを含み、各々の空間クラスタは、少なくとも1つのプロブを含む。もしカラークラスタが複数の空間クラスタを含むならば、それらの空間クラスタは、全てカラーが近接していなければならない。もし空間クラスタが多数のプロブを含むならば、  
30  
プロブは相互に接近していなければならない。

【0038】

プロブが空間クラスタへ合体されるべきかどうかを決定するため、拡張境界ボックスが  
40  
使用される。拡張境界ボックスの目的は、同じ空間クラスタの中にあると考えられるプロブの間の許容される距離閾値を定義することである。重なり合う（オーバーラップする）拡張境界ボックスを有するプロブは、相互に十分接近しており、同じ空間クラスタに属するものと考えられる。

【0039】

拡張境界ボックスの例は、図8に示される。拡張境界ボックス754は、元の境界ボックス752の幅および高さ按比例して所定の限度まで拡張される。

【0040】

たとえば、図2へ戻って、プロブ#1（文字「A」）として識別されるプロブは、x方向で2列だけ拡張される拡張境界ボックスを有し、EXP\_MAX\_X1の拡張境界ボ  
50

クスの境界を与える。同様に、プロブ# 2も2列だけ拡張され、拡張境界ボックスの境界EXP\_MIN\_X2を与える。したがって、もしプロブ# 1および# 2のカラー値が十分に近接していれば、プロブ# 1は、プロブ# 2として識別されるプロブを有する同じ空間クラスタの中に含まれる。なぜなら、プロブ# 1の拡張境界ボックス(EXP\_MAX\_X1)は、プロブ「2」の拡張境界ボックス(EXP\_MIN\_X2)とオーバーラップするからである。

【0041】

境界ボックスの拡張サイズは、結果としてのバイナリ前景層の数と、バイナリ層の圧縮効率とのトレードオフ(取捨選択)に基づいて選ばれる。

【0042】

クラスタがプロブから組立られるとき、クラスタの内容を追跡するため、クラスタ構築モジュール720は、各々のタイルに定着(root)される2レベルのクラスタデータを維持する。空間クラスタの内容フィールドは、常にプロブリストの冒頭を指し、カラークラスタの内容フィールドは、常に空間クラスタリストの冒頭を指す。

【0043】

ネクスト(NEXT)フィールドは、リストの次のレコードを指す。プロブのネクストフィールドは次のプロブを指す。空間クラスタのネクストフィールドは次の空間クラスタを指す。カラークラスタのネクストフィールドは次のカラークラスタを指す。リストの最後では、ネクストフィールドは空値(NULL)である。クラスタ構築モジュール720によって使用される例示的データ構造は、図9で示される。

【0044】

図9は、プロブがクラスタへ組み込まれるとき、プロブの各々を追跡するために使用される3レベルデータ構造を示す。データ構造のトップレベル735は、たとえば、カラークラスタ(CCLUSTER)746、747、および748を含む。カラークラスタは、所定のカラー閾値と比較して同一または類似のカラーを有する空間クラスタをグループにまとめたものである。データ構造の第2の(中間)レベル736は、オーバーラップする拡張境界ボックスを有するプロブを含む空間クラスタ(SCLUSTER)731、732、733、および734を含む。

【0045】

ボトムレベル737は元のプロブ(BLOB)を含み、「内容(CONT)」ポインタによって空間クラスタへ、また、「ネクスト(NEXT)」ポインタによって同じクラスタのプロブへリンクされる。こうして、各々のクラスタまたはプロブは、ネクストポインタを使用して同じレベルの他のクラスタまたはプロブへリンクされ、クラスタは内容ポインタによって、より低いレベルの他のクラスタへリンクされる。

【0046】

各々のタイルにある最初のプロブは、新しいクラスタ連鎖を作成し、この新しいクラスタ連鎖へ、後続のプロブがリンクされる。

【0047】

タイルの中にある次のプロブは、既存の空間クラスタへ割り当てるか、そのプロブを含ませるため新しいクラスタを作成することができる。たとえば、もし現在のプロブ872が、既に作成されたカラークラスタのいずれとも適合しないと決定されるならば、新しいカラークラスタ846(図10参照)がクラスタリストの中に作成され、カラークラスタ連鎖の始めに挿入される。

【0048】

図10は、新しいカラークラスタ846を、既にクラスタ845および844を含むクラスタ連鎖へ挿入する手順を示す。新しいカラークラスタを挿入するため、新しいカラークラスタの挿入に先立って、対応するタイルエントリのクラスタインデックスが、新しいカラークラスタのネクストフィールドへコピーされ、タイルエントリは、新しいカラークラスタのインデックスを含むように修正される。

【0049】

10

20

30

40

50

図 1 1 は、クラスタデータ構造からのリンクされたクラスタエントリの例示的模式図である。クラスタエントリは、図 1 0 の下部に示されるクラスタに対応するエントリである。

【 0 0 5 0 】

もし 2 つのカラークラスタが、カラー閾値の中に入るカラー値を有する、または 2 つの空間クラスタが、オーバーラップする拡張境界ボックスを有するならば、それらのクラスタは結合されてよい。結合は、平均カラーおよびピクセルカウントを更新することを含む。2 つの拡張境界ボックスは結合されて、双方の境界を定める 1 つの境界ボックスが生成される。

【 0 0 5 1 】

上記のクラスタ結合方法は最適化されているので、それはプロブをタイルクラスタ構造へ最初に挿入するときにも使用される。候補としての一時的な空間クラスタが作成され、その空間クラスタは、現在のプロブについて計算された拡張境界ボックスを有し、その内容フィールドは現在のプロブを指す。カラーが適合（一致）するかどうか、カラークラスタ連鎖が探査される。もし適合するカラークラスタが発見されなければ、現在のクラスタのカラーおよび拡張境界ボックスフィールドを使用して、新しいカラークラスタが作成（初期化）される。次に、適合する（または新しい）カラークラスタの空間クラスタリストが、現在の空間クラスタと交差する拡張境界ボックスを有する空間クラスタについて探査される。もし空間の適合が発見されるならば、現在のクラスタが、適合（一致）する空間クラスタへ結合される。もし適合が発見されなければ、現在のクラスタは、カラークラスタの既存の内容の始めへ付加される。空間適合のテストは、図 1 2 に示される。空間適合のテストは、すでにカラーが近接して一致しているクラスタ間でのみ実行される。空間適合の要件は、例えば図 1 2 のボックス 8 8 1、8 8 2、8 8 3、8 8 4 と 8 8 5 のように、少なくとも 2 つのクラスタの拡張境界ボックス間に少なくとも 1 ピクセルのオーバーラップがあることである。

【 0 0 5 2 】

もしオーバーラップするクラスタが発見されると、現在のクラスタデータは、その空間クラスタへ結合される。新しく集成された境界ボックスが確立され、この境界ボックスは、結合された境界ボックスの広がりを含む。

【 0 0 5 3 】

前述したように、空間的にオーバーラップしたクラスタが発見されないと、現在の空間クラスタは、カラークラスタ（新しいカラークラスタである可能性もある）の内容リストの始めに付加される。図 1 3 は、新しい空間クラスタを既存のクラスタ連鎖へ挿入するプロセスを示す。新しいクラスタ 8 5 9 は、空間クラスタ連鎖の始めに挿入される。カラークラスタ 8 4 6 の前の内容フィールド（PREV CONT）は、クラスタ 8 5 9 のネクストフィールドへコピーされ、前にクラスタ 8 5 8 を指したカラークラスタ 8 4 6 の内容フィールドは、新しいクラスタ 8 5 9 を示すように修正される。最後に、新しい空間プロブ 8 5 9 の内容フィールドは、現在のプロブ 8 7 3 となるように更新される。このケースについては、これにより 3 レベル構造 7 3 5 ~ 7 3 7 が完成される。

【 0 0 5 4 】

次に、クラスタ構築モジュール 7 2 0 は、G T B の次のプロブを検査するように進行する。このプロセスは、全てのプロブが、それらのカラー適合および空間近接性に基づいて、3 レベル構造 7 3 5 ~ 7 3 7 へ挿入されるまで反復される。

【 0 0 5 5 】

一度、3 レベル構造 7 3 5 ~ 7 3 7 の全体が完成されると、クラスタ構築モジュール 7 2 0 は、クラスタの結合を開始し、残りのクラスタの数を減らす。

【 0 0 5 6 】

クラスタが結合された後、図 1 4 で示されるように、タイル境界を横切ってオーバーラップするクラスタを結合することによって、各々の 2 x 2 の隣接タイル集合（セット）間の境界が除去される。タイル境界の間の境界が除去される前では、図 1 4 の上部左側のタ

10

20

30

40

50

イルは3つのカラークラスタ810、814、および818を有し、上部右側は2つのカラークラスタ812および816を有し、他のタイルは、各クラスタ820および822をそれぞれ有する(全部で7つのクラスタ)。平均のカラーが密接に適合するものと仮定すれば、図14の下部は、1つの結合ステップの後に、4つのカラークラスタ824、826、828、および830だけが残ることを示している。

**【0057】**

この時点で、新しいクラスタアレイ(配列)は、各々の方向にタイルの半数を割り振られ、エントリは、結合されたデータで満たされる。1つの2×2のタイル集合についての結合動作は、図14で示される。結合プロセスは、タイルサイズが画像データの全体をカバーするまで反復される。

10

**【0058】**

最終的なクラスタ構造は、クラスタ結合モジュール705へ渡される。クラスタ結合モジュール705は、最初に、クラスタ構造からカラークラスタのトップレベル735を除去する。クラスタ結合モジュール705の動作は、図15で示される。

**【0059】**

クラスタ結合モジュール705は、カラー適合および空間オーバーラップに基づいてクラスタを結合できるかどうかについて、再び空間クラスタの連鎖を探索する。適合が発見されると、前述したように、データは最初のクラスタへ結合され、第2のクラスタが削除される。リストが削除されたクラスタを有しないように、リンクが再結合される。クラスタ結合モジュール705の動作は、図16で示される。

20

**【0060】**

空間クラスタは、クラスタ結合モジュール705によって、関連するプロブデータの内容(CONT)から直接再計算される。クラスタ結合モジュール705は、空間クラスタの連鎖を評価することによって再計算する。各々の空間クラスタについて、ピクセルカウント、カラー平均、および集成(集約)された境界ボックス(拡張なし)が、クラスタに含まれるプロブに基づいて再計算される。

**【0061】**

クラスタ結合モジュール705は、不良と見なされるクラスタを除去し、プロブマークフィールドを再調整する。即ち、マークされた空間クラスタは除去され、空間クラスタのリストが除去されたクラスタなしで保持されるようにネクストリンクが更新される。

30

**【0062】**

プロブ識別プロセスの正常な結果は、全てのプロブが、全ての他のプロブの完全に内側にあるか、完全に外側にあるかである。クラスタ結合モジュール705が結合プロセスを完了した後、内部プロブモジュール760は、各々のプロブが他のマークされないプロブに完全に含まれるかどうかをチェックする。即ち、含まれるならば、それは「内部」プロブである。内部プロブであると決定されたプロブは、クラスタから除去される。

**【0063】**

内部プロブモジュール760の動作は、内部プロブがあるかどうかを全てのプロブペアの間でチェックすることが必要である。このような動作が与える計算負荷を理由として、多くの不良または小さなプロブが除去(マーク)されるクラスタ結合モジュール705の後で、内部プロブモジュールが適用される。

40

**【0064】**

内部プロブモジュール760は、プロブペアの全ての可能な組み合わせを検査する。境界ボックスのMIN\_Y値によって先に並べ替えられたGTBから、最初のプロブであるプロブAが順次に取り出される。次に、第2のプロブであるAの直後で始まるプロブBがGTBから取り出される。各々のプロブについて、グローバルプロブテーブルから境界ボックスおよびプロブIDが検索される。第2のプロブであるプロブBは、プロブAに関して内部プロブであるかどうかをテストされる。

**【0065】**

プロブBがプロブAの内部プロブであるかどうかを決定するプロセスが、図17に示さ

50

れている。図 17 は、プロブ A を含む境界ボックス 884 の中央線 881 を示す。中央線 881 の最初の 3 つのピクセルの各々は、プロブ A の内側にあることを判断され、プロブ B への遷移がピクセル 883 について起こる。これにより、プロブ B は A の内部プロブであると判断される。

【0066】

図 18 に示される状態マシン 890 は、内部プロブを決定する例示的プロセスを示す。

【0067】

一度、最初のプロブ ID A の状態であることが発見されると、状態は 1 へ変えられる。ここで、状態マシン 890 はプロブ ID B の状態の発生を探し始める。プロブ ID A 状態が観察され続けられる限り、状態は 1 に留まる。しかし、A または B 以外の ID が  
10  
、プロブ ID A の後およびプロブ ID B の前に発見されると、状態はゼロへリセットされて再開始される。プロブ ID B がプロブ ID A の直後に発見されるときにのみ、状態は 2 へ変えられる。これは B が内部プロブであることを示す。この状態に達すると、テストは直ちに中止されてよい。

【0068】

状態 2 が発見されると、B プロブのマークフィールド 430 の内部プロブビットをオンにすることによって、B プロブが内部プロブとしてマークされる。その他の全ての場合、B プロブは内部プロブとしてマークされない。

【0069】

前述したように、B が A の内部プロブであるかどうかをチェックした後、プロセスは、  
20  
G T B の次の B プロブをチェックするように進行する。これは、G T B の最後のプロブが評価されるか、B の MIN\_Y が A の MAX\_Y よりも大きくなるまで続く。

【0070】

次に、内部プロブモジュール 760 は、各々のプロブのピクセル数に基づいて、境界ボックスの和集合（内部プロブを除く）および新しい加重平均カラーを再計算する。

【0071】

もし結果のクラスタサイズが十分に大きければ、更新された Y C C カラー合計はピクセルカウントによって正規化され、内部プロブを除く新しい平均カラーがクラスタデータへ  
30  
ロードされ、先の合計と置き換わる。同様に、新しいピクセルカウントおよび更新された境界ボックス情報が、内部プロブを除いて、クラスタデータへロードされる。

【0072】

クラスタ並べ替えモジュール 730 は、空間クラスタを、最大ピクセルカウントから最小ピクセルカウントまでサイズによって並べ替える。並べ替え方法は、この後で図 19 に関して説明するように、バブルソート（基本交換法）アルゴリズムに基づくものであってよい。

【0073】

空間クラスタ 991、992、993 の連鎖は、一時に 1 つのクラスタがたどられる。各々のステップで、現在のクラスタ 991 のサイズは、後続のクラスタ 992 のサイズと比較される。後続のクラスタ 992 は、現在のクラスタ 991 からのネクストリンクをた  
40  
どることによって識別される。クラスタサイズはカウントフィールドによって与えられ、このカウントフィールドはクラスタの集成されたピクセル数を含む。もし 991 のピクセル数が 992 のカウントフィールドのピクセル数よりも小さければ、連鎖の中のクラスタ位置は交換され、それらのネクストリンクは、新しい位置を反映するように更新される。

【0074】

層（平面）生成モジュール 750 がクラスタ並べ替えモジュール 730 に続く。層生成モジュール 750 は、空間クラスタをバイナリマスク平面へ変換する。各々のクラスタは、1 から始まる層（レイヤ）番号を順次に割り当てられる。前述したように、空間クラスタ、そのためバイナリ平面は、ピクセルカウント値の降順で並べ替えられる。層の数は、生き残った空間クラスタ + 1 の数と同じである。

【0075】

層生成モジュール750はN+1個のバイナリ平面を作成する。平面1からNは、対応する層のバイナリデータを含む。平面0は、層1からNまでのOR結合されたデータと等しいバイナリセクタ層を含む。平面0のサイズは、背景平面と同じである。

#### 【0076】

層生成モジュール750は、一時に1つの平面を作成する。前述したように、クラスタ並べ替えモジュール730は、ピクセルカウント値の降順で平面を並べ替える。各々の平面について、平面サイズを計算し、バイナリ画像に正確なメモリ記憶量を割り振るため、境界ボックスが使用される。平面は、最初にゼロに設定される。次に、平面のクラスタ内容リストの各々のプロブが処理される。各々のプロブについて、プロブの境界ボックスに対応するIDマップ画像の全てのロケーションが検査される。プロブIDに等しいIDを有するこの境界ボックスの全てのマップピクセルについて、層平面の対応するビットが1へ設定される。

10

#### 【0077】

上記の説明は、ハードウェアの実施形態を参照して与えられたが、プロブクラスタの処理は、マイクロプロセッサで実行されるソフトウェアとしても実現されることができる。この場合、方法の各々のステップは、適切にプログラムされたマイクロプロセッサ、またはハードウェアとソフトウェアとの組み合わせによって実行される。

#### 【0078】

図20は、グローバルプロブテーブルにリストされたプロブをn個のバイナリ前景層へ変換する方法の1つの例示的实施形態を示すフローチャートである。図20で示されるように、方法の動作は、ステップS200で始まる。ステップS200では、グローバルプロブテーブルの各々のプロブが正規化される。様々な例示的实施形態において、プロブはピクセルカウントに従って正規化される。次に、プロセスはステップS300へ進む。ステップS300では、もしオプションが選択されるならば、プロブのYCCが、連続グレートーンへ変換される。次に、プロセスはステップS400へ進行する。ステップS400では、プロブは、走査された画像におけるそれらの位置に従って並べ替えられる。次にプロセスは、ステップS500へ進行する。ステップS500では、各々のプロブの並べ替えられた順序を記憶するため、インデックスが各々のプロブへ割り当てられる。次に、プロセスはステップS600へ進行する。ステップS600では、各々のプロブが解析され、多数のテスト条件の1つに従ってマークされる。次にプロセスはステップS700へ進行する。ステップS700では、或る閾値を超過したカラー変動を有するプロブがマークされる。次にプロセスはステップS800へ進行する。

20

30

#### 【0079】

ステップS800では、クラスタがGTBから構築される。その場合、カラーが類似したプロブであって相互に近接したプロブから、クラスタが作られる。ステップS900では、ステップS800で作成されたクラスタテーブルのトップレベル(カラーレベル)が除去される。ステップS1000では、生き残りのクラスタの数を減らすため、十分に類似したカラーを有し空間的にオーバーラップしたクラスタが結合される。ステップS1100では、ステップS1000の結合動作の後で残ったクラスタのカラーが再計算され、クラスタカウントが更新される。ステップS1200では、最小閾値数のピクセルを含まないクラスタが、クラスタリストから除去される。ステップS1400では、他のプロブの中に完全に含まれるプロブが「内部プロブ」としてマークされる。ステップS1500では、内部プロブとして前のステップでマークされたプロブが除去される。ステップS1600では、ステップS1500で内部プロブを除去した後の残りのクラスタが再計算される。ステップS1700では、プロブがサイズによって並べ替えられる。

40

#### 【0080】

ステップS1800では、残りのクラスタがバイナリ前景平面へ順次に割り当てられる。ステップS1900では、異なった前景平面が、偶数バイト境界(even byte boundary)へバイトを揃えられる。次に、ステップS2000では、N+1個のバイナリ平面が生成される。セクタ平面である平面0は、グレースケール背景平面の中で画像データをマ

50

スクするために使用される。ステップS 2 1 0 0でグレースケール背景平面は、圧縮能力を向上させるために調整される。次に、ステップS 2 2 0 0では、ステップS 1 9 0 0で決定されたバイナリ前景平面の各々が、そのような平面に適した圧縮手法を使用して圧縮される。次に、動作はステップS 2 2 0 0へ続く。

【0081】

ステップS 2 2 0 0では、背景平面が、そのようなデータに適した圧縮手法を使用して圧縮される。次に、ステップS 2 3 0 0では、圧縮されたバイナリ前景平面および圧縮されたグレースケール背景平面から、ポータブルドキュメント形式(PDF)文書ファイルが生成される。次に、ステップS 2 4 0 0では、生成されたPDF形式文書ファイルが下流のプロセッサへ出力され、および/またはメモリに記憶される。そして動作は終了する。

【0082】

更に、様々な例示的实施形態において、PDF形式の代わりに、任意の既知または後に開発される文書形式が、ステップS 2 3 0 0およびS 2 4 0 0で使用されてよいことを理解されたい。

【0083】

本発明は、様々な例示的实施形態に結びつけて説明されたが、これらの実施形態は、例証として考えるべきであり、限定として考えるべきではない。様々な修正、置換などが、本発明の精神および範囲の中で可能である。

【図面の簡単な説明】

【0084】

【図1】複数のバイナリ前景平面および背景平面へ分解される画像の例示的实施形態を示す図。

【図2】プロブID画像データの1つの例示的实施形態を示す図。

【図3】プロブクラスタ処理モジュールが動作することのできる1つの例示的システムを示す図。

【図4】グローバルプロブテーブルにおけるプロブまたはクラスタのエントリの1つの例示的实施形態を示す図。

【図5】図3のプロブクラスタ処理モジュールの1つの例示的实施形態を示す図。

【図6】境界ボックスの例示的实施形態を示す図。

【図7】1ページの画像データがどのように複数のタイルへ分割されるかを示す図。

【図8】拡張された境界ボックスの1つの例示的实施形態を示す図。

【図9】空間およびカラー関係に従ってプロブのクラスタを記述するデータ構造の1つの例示的实施形態を示す図。

【図10】図5のプロブクラスタ処理モジュールのクラスタ構築モジュールによって、どのように新しいクラスタがクラスタ連鎖の始めに挿入されるかを示す図。

【図11】ネクストフィールドおよび内容リンクフィールドを有するクラスタテーブルのエントリを示す図。

【図12】1つのクラスタが他のクラスタと空間的にオーバーラップすると判定されるための要件を示す図。

【図13】図5のプロブクラスタ処理モジュールのクラスタ構築モジュールによって、空間的にオーバーラップする拡張境界ボックスを有するプロブが、どのようにクラスタ連鎖へ付加されるかを示す図。

【図14】画像ページのタイルを結合することによって、どのようにクラスタが結合されるかを示す図。

【図15】プロブクラスタ処理モジュールのクラスタ結合モジュールによって、どのようにカラークラスタのトップレベルが除去されるかを示す図。

【図16】プロブクラスタ処理モジュールのクラスタ結合モジュールによって、どのように空間クラスタが結合されるかを示す図。

【図17】プロブクラスタ処理モジュールの内部プロブモジュールによる内部プロブの検出を示す図。

10

20

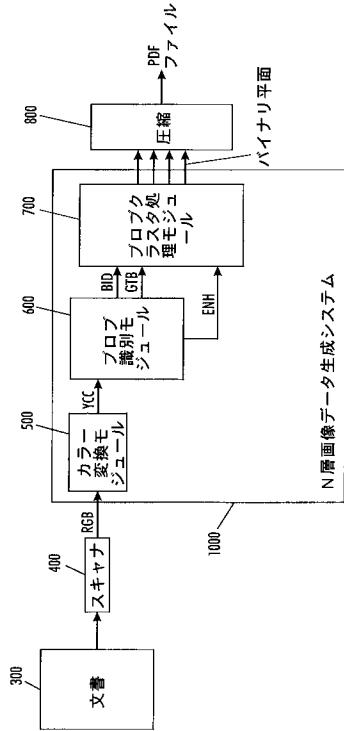
30

40

50



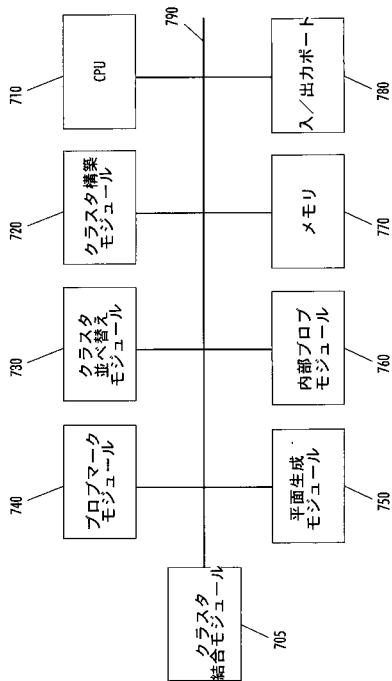
【図3】



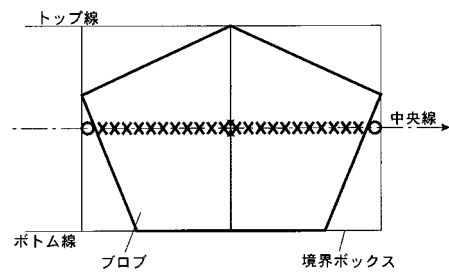
【図4】

310	
320	MIN_X
330	MAX_X
340	MIN_Y
350	MAX_Y
360	SUM_Y (合計Y)
370	SUM_Cb (合計Cb)
380	SUM_Cr (合計Cr)
390	COUNT (カウント)
410	VAR (分散)
420	ID
430	INDEX (インデックス)
440	MARK (マーク)
450	NEXT (ネクスト)
	CONT (内容)

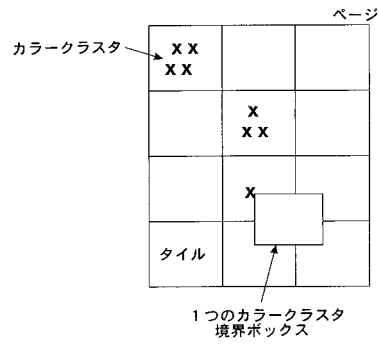
【図5】



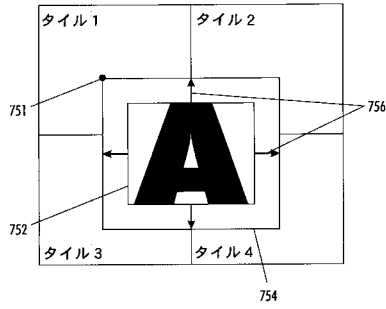
【図6】



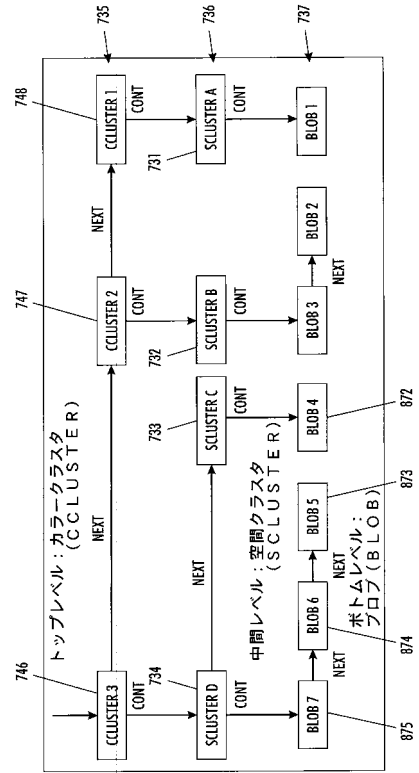
【図7】



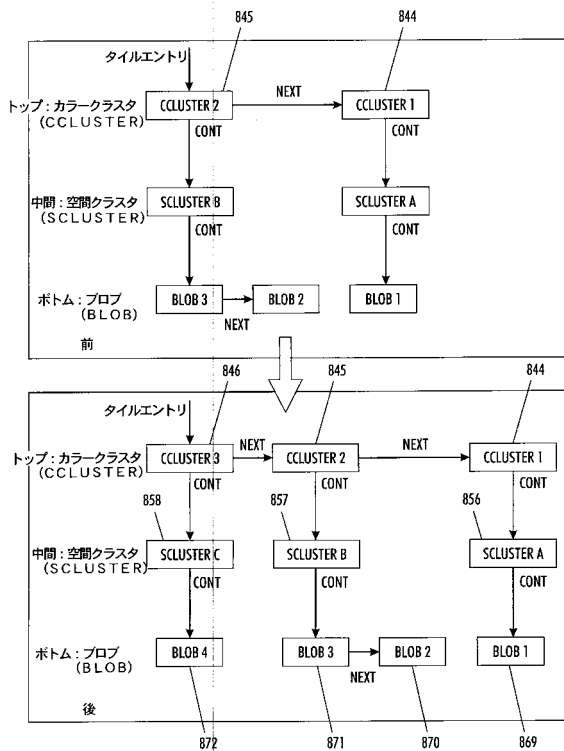
【図8】



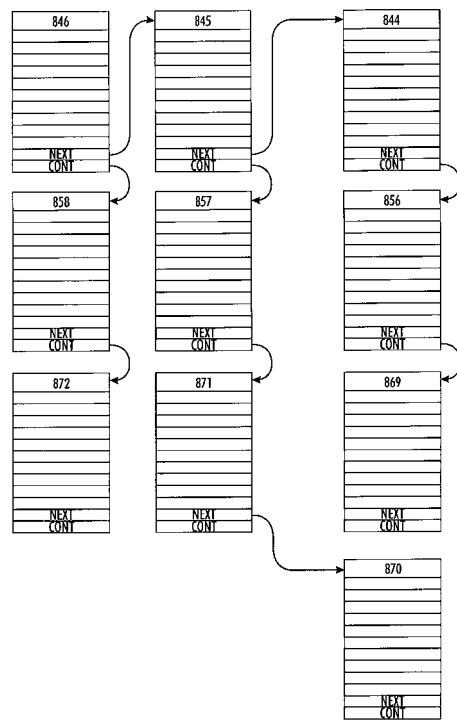
【図9】



【図10】

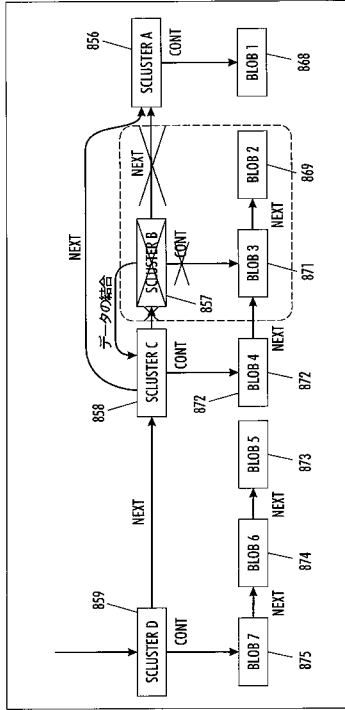


【図11】

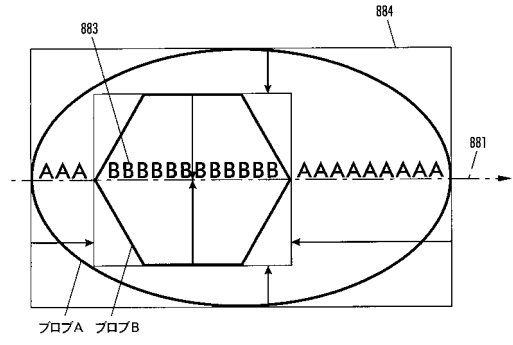




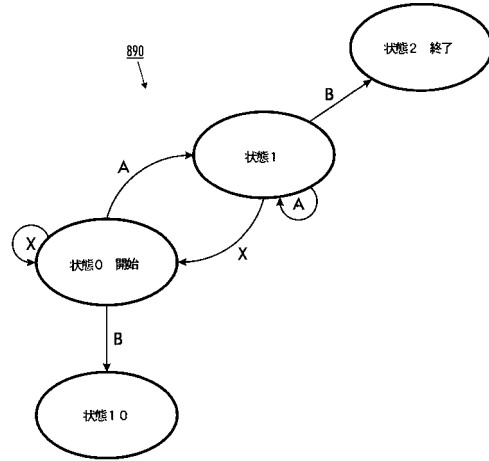
【図16】



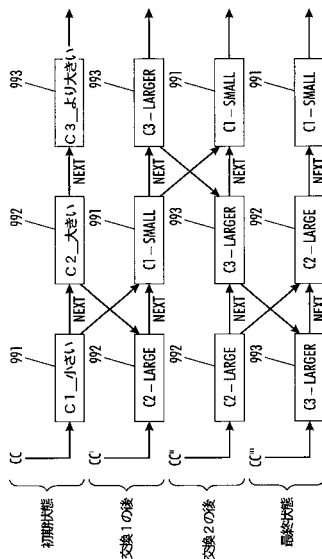
【図17】



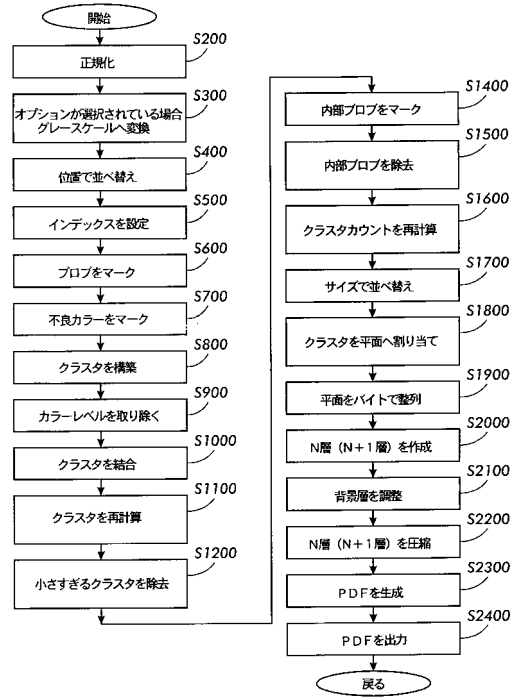
【図18】



【図19】



【図20】



---

フロントページの続き

(72)発明者 アスガー ナファリエ

アメリカ合衆国 94025 カリフォルニア州 メンロ パーク モリー ドライブ 525

(72)発明者 ドロン クレター

アメリカ合衆国 94402 カリフォルニア州 サン マテオ タートル ベイ プレイス 20

審査官 松永 稔

(56)参考文献 国際公開第2002/056255(WO, A1)

特開2005-228331(JP, A)

特開2005-229620(JP, A)

特開2005-228342(JP, A)

特開2005-228340(JP, A)

特開2003-228712(JP, A)

(58)調査した分野(Int.Cl., DB名)

G06T 7/00

G06T 1/00

H04N 1/41