

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6863675号
(P6863675)

(45) 発行日 令和3年4月21日 (2021.4.21)

(24) 登録日 令和3年4月5日 (2021.4.5)

(51) Int. Cl.	F I
G06T 7/10 (2017.01)	G06T 7/10
G06T 11/80 (2006.01)	G06T 11/80 A

請求項の数 13 外国語出願 (全 20 頁)

(21) 出願番号	特願2015-225816 (P2015-225816)	(73) 特許権者	500102435
(22) 出願日	平成27年11月18日 (2015.11.18)		ダッソー システムズ
(65) 公開番号	特開2016-100016 (P2016-100016A)		DASSAULT SYSTEMES
(43) 公開日	平成28年5月30日 (2016.5.30)		フランス国 78140 ペリジー ビラ
審査請求日	平成30年10月29日 (2018.10.29)		クブレー リュ マルセル ダッソー 1
(31) 優先権主張番号	14306826.0		O
(32) 優先日	平成26年11月18日 (2014.11.18)	(74) 代理人	110001243
(33) 優先権主張国・地域又は機関	欧州特許庁 (EP)		特許業務法人 谷・阿部特許事務所
		(72) 発明者	ローラ ペティユー
			フランス 92100 ブーローニュ ビ
			ヤンクール リュ ド シリー 140
		(72) 発明者	アメリ ルナール
			フランス 92130 イシー レ ムリ
			ノー リュ ドゥ シャン シャルドン
			24 アパートメント 301
			最終頁に続く

(54) 【発明の名称】 画像分割

(57) 【特許請求の範囲】

【請求項 1】

画像を分割するためのコンピュータ実行方法であって、
 縁を含む画像を表示するステップ (S10) と、
 前記画像に重ねて曲線を描くステップ (S20) と、
 前記曲線と前記画像の前記縁とで囲まれた、全ての閉じられたエリアを計算するステップであって、各々の閉じられたエリアは、他の計算された閉じられたエリアから独立しており、個々に選択可能である、ステップと

を備え、

前記画像に重ねて前記曲線を描くステップの後に、

前記画像に重ねて描いた前記曲線から連続したポリラインを構築するステップ (S40) と、

前記連続したポリラインを形成する各線に関して、前記表示された画像の前記各線を辺として有するピクセルを識別するステップ (S50) と

をさらに備え、

前記連続したポリラインを構築するステップ (S40) は、

前記画像に重ねて描かれた前記曲線の2つの異なる終点をリンクするピクセルの集合を計算するステップであって、前記集合の隣接するピクセルは共通の辺を有する、ステップと、

前記ポリラインの各線が、前記計算されたピクセルの集合の前記ピクセルの1つの辺で

10

20

ある、連続したポリラインを構築するステップと
を含むことを特徴とするコンピュータ実行方法。

【請求項 2】

前記表示された画像の前記ピクセルを識別するステップ (S 5 0) の後に、
各々識別されたピクセルをマーク付けするステップであって、前記マーク付けは前記各々識別されたピクセルに対するポリラインの位置を示す、ステップをさらに備えることを特徴とする請求項 1 に記載のコンピュータ実行方法。

【請求項 3】

前記各々識別されたピクセルをマーク付けするステップは、 R G B A 色空間で符号化された色で、各々識別されたピクセルを色付けするステップ (S 6 0) により実行され、
第 1 の R G B A 値は、ポリラインが前記ピクセルの右側に位置することを示し、
第 2 の R G B A 値は、ポリラインが前記ピクセルの左側に位置することを示し、
第 3 の R G B A 値は、ポリラインが前記ピクセルの上側に位置することを示し、
第 4 の R G B A 値は、ポリラインが前記ピクセルの下側に位置することを示すことを特徴とする請求項 2 に記載のコンピュータ実行方法。

【請求項 4】

前記各々識別されたピクセルを色付けするステップは、累積的であることを特徴とする請求項 3 に記載のコンピュータ実行方法。

【請求項 5】

前記曲線と前記画像の前記縁とで囲まれた全ての閉じられたエリアを計算するステップは領域拡張アルゴリズムを用いることによって実行され、ピクセルの領域加入基準は前記連続したポリラインを超えないことを特徴とする請求項 2 から請求項 4 のいずれか 1 項に記載のコンピュータ実行方法。

【請求項 6】

前記領域拡張アルゴリズムは、
(i) 第 1 の閉じられたエリアに属する、前記画像の第 1 のピクセルを選択するステップと、
(i i) 前記選択された第 1 のピクセルに隣接する、前記領域加入基準を満たすピクセルを、前記選択されたピクセルへの前記マーク付けを用いて、識別するステップと、
(i i i) ステップ (i i) で識別された各々隣接するピクセルを、前記選択された第 1 のピクセルとして、選択、処理されるピクセルのリストに追加するステップと、
(i v) 前記選択された第 1 のピクセルをチェック済みとしてマーク付けし、前記選択された第 1 のピクセルを前記第 1 の閉じられたエリアに属するピクセルのリストに追加するステップと、
(v) 選択すべきピクセルの前記リストから第 2 のピクセルを選択し、ステップ (i i) から (i v) を前記第 2 のピクセルに対して実行するステップと、
(v i) 選択すべきピクセルの前記リストの全ての前記ピクセルが選択されるまで、前記ステップ (v) を繰り返すステップと
を備えることを特徴とする請求項 5 に記載のコンピュータ実行方法。

【請求項 7】

(v i i) 前記画像の第 2 の閉じられたエリアに属する第 3 のピクセルを識別するステップであって、前記第 3 のピクセルはステップ (i v) でマーク付けされていない、ステップと
(v i i i) 前記ステップ (i i) から (v i) を実行するステップと、
(i x) 前記画像の全ての前記ピクセルがチェック済みとしてマーク付けされるまで前記ステップ (v i i) から (v i i i) を繰り返すステップと
をさらに備えることを特徴とする請求項 6 に記載のコンピュータ実行方法。

【請求項 8】

前記連続したポリラインを構築するステップから開始され、実行されるステップはマスクに対して実行され、前記マスクは前記表示された画像と同じサイズおよび同じ数のピク

10

20

30

40

50

セルを有する画像であることを特徴とする請求項 1 から 7 のいずれか 1 項に記載のコンピュータ実行方法。

【請求項 9】

前記マスクは空白画像として初期設定され、チェック済みとしてマーク付けされた前記ピクセルは閉じられたエリアに属するピクセルのリストに追加される場合に、黒で色付けされることを特徴とする、請求項 7 と組み合わせられた、請求項 8 に記載のコンピュータ実行方法。

【請求項 10】

前記計算された閉じられたエリアは、四分木データ構造に記憶されることを特徴とする請求項 1 から 9 のいずれか 1 項に記載のコンピュータ実行方法。

10

【請求項 11】

前記全ての閉じられたエリアを計算するステップの後に、
少なくとも 1 つの閉じられたエリアを個々に選択するステップと、
選択された各々閉じられたエリアに関して独立した画像を生成するステップと
をさらに備えることを特徴とする請求項 1 から 10 の 1 項に記載のコンピュータ実行方法。

【請求項 12】

請求項 1 から 11 のいずれか 1 項の方法を実行するための命令を備えたコンピュータプログラム。

【請求項 13】

20

メモリとグラフィカルユーザインタフェースとに結合されたプロセッサを備え、前記メモリは請求項 12 の前記コンピュータプログラムを記録することを特徴とするシステム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、コンピュータプログラム及びシステムの分野に関し、より詳細には、画像を分割する方法、システム、及びプログラムに関する。

【背景技術】

【0002】

画像の処理に用いられる描画ソフトウェアのほとんどは、画像のサブパート (subpart) を抽出する機能をユーザに提供する。典型的には、ユーザは、画像を分割する時、以下の操作を行う必要がある。最初に、ユーザは、自分が探しているサブパートの境界を定める。ユーザは、自由な形を描いてもよいし、プリミティブな形 (長方形、円形・・・) を描いてもよい。そうすると、ユーザの描画に基づいて、ソフトウェアは、2 つの異なる部分、すなわち、描画の内側の第 1 の部分と、描画の外側の第 2 の部分に分けようとする。

30

【0003】

ユーザが自由な形を描いた場合、ソフトウェアは、最初の点を最後の点にリンクさせて、描画を閉じる。これは、描画が画像の「内側」と「外側」とを画定することを確実にするために必要である。これは、図 1、図 2 に示してある。図 1 においては、ユーザが描いた自由な形 10 は、その開始点と終点の間が連結されていないので、閉じられていない。リンク 20 が、図 2 に示すように、(境界である) 閉じられた自由な形を取得するためにソフトウェアによって作成される。このようにして、画像を分割する既知の方法によって、画像を 2 つの異なったエリアに分けることができる。1 番目のエリアは境界の内側であり、2 番目のエリアは境界の外側である。そして、ユーザが分離を有効に (自動的に行われる場合もある) した後、内側の部分のみが保持される。この選択は、逆に、外側部分を保持する場合もあるが、常にそうとは限らない。ただ画像のサブパートが選択される。この選択から別の画像を作成するために、または、別の画像または何か他のものを取り除くために、ユーザは、さらにアクションを行う必要がある。例えば、ユーザの選択を用いて別の画像を作成するためには、ユーザは、新しい画像を作成し、コピーペーストを行うか、自分の選択を逆にして、不要な部分を取り除く必要がある。

40

50

【0004】

画像を分割するための既知の解決法は、幾つかの制約を伴う。第1の制約は、ユーザは、自分が欲するサブパートを常に正確に画定できるとは限らないことである。これは、ソフトウェアは、ユーザがプリミティブな形（例えば、長方形）を用いて境界を特定するのを可能にするだけか、（例えば、その描画が既に閉じた曲線を画定している場合）ソフトウェアは描画の形を考慮しないで、最初の点と最後の点をリンクして閉じた曲線を作成するからである。これは、図3、図4に示すように、良くない結果につながる。図3において、ユーザは画像に自由な形を描いた。これは閉じられた自由な形である。しかしながら、閉じられた自由な形の開始点と終点の間が連結されていないので、ソフトウェアは、これら2つの点の間にリンクを生成して、図4に示す第2の閉じられたエリアを生成する。よって、ユーザの意図は考慮されない。

10

【0005】

別の短所は、ソフトウェアが保持するサブパートは常に内側であるということである。一部のプログラムにおいては、ユーザは、それを逆にして、内側のパートではなく外側のパートを保持する操作を行うことができるが、常に可能とは限らない。

【0006】

さらなる短所は、幾つかのサブパートを選択することは、時間がかかり、面倒なことである。これは、ユーザが画像に取り組み、2つの異なるパートを画像から抽出したいという以下のシナリオで例証される。既存のソフトウェアを用いると、ユーザは、次のステップに従う必要がある。すなわち、元画像をインポートする、閉じた形を描くことによって第1のサブパートを画定する、新しい画像にこの部分をコピー/ペーストする（または、選択を逆にして、選択していない部分を除く）、元画像を再びインポートする、閉じた形を描くことによって第2のサブパートを画定する、新しい画像にこの部分をコピー/ペーストする（または、選択を逆にして、選択していない部分を除く）。

20

【0007】

さらなる短所の1つは、ユーザが画像を単に半分に切りたい場合、図5、図6に示すように、ユーザが欲しい画像の部分の周りに円形を描く必要がある。これは、直観的ではなく、時間がかかり、かつ、人間工学的に効率が良くない。

【0008】

さらなる短所としては、既知の解決法は、ユーザが描いた閉じられたエリアまたはプリミティブな形の「内側」と「外側」の検出に依存しているので、サブパートの選択は、予測不可能な場合がある。しかしながら、自由な形の「内側」と「外側」の識別は、現在のアルゴリズムにとっては複雑なタスクとなり得る。特に、自由な形が幾つかの交点を含む時、アルゴリズムは、自由な形のどの部分が内側でどの部分が外側か識別できない場合がある。さらに、閉じられたエリアの「内側」と「外側」の検出は、現在のアルゴリズムにとって簡単ではないので、コンピュータシステムにとっては時間がかかる。よって、かなりの量のコンピューティング資源が、予測不可能であり得る結果のために用いられる。

30

【0009】

この文脈で、画像を1つまたは複数の選択可能なサブパートに分割する改良された方法が必要とされている。

40

【発明の概要】

【0010】

従って、画像を分割するコンピュータ実行方法を提供する。コンピュータ実行方法は以下のステップを含む。

- ・縁を含む画像を表示するステップ。
- ・画像に重ねて曲線を描くステップ。
- ・曲線と画像の縁とで囲まれた、全ての閉じられたエリアを計算するステップであって、各々の閉じられたエリアは、他の計算された閉じられたエリアから独立しており、個々に選択可能である、ステップ。

【0011】

50

コンピュータ実行方法は以下のステップをさらに含む。

・画像に重ねて曲線を描くステップの後、画像に重ねて描いた曲線から連続したポリラインを構築するステップ（S40）と、連続したポリラインを形成する各線に関して、表示された画像の当該各線を辺として有するピクセルを識別するステップ。

・連続したポリラインを構築するステップは、画像に重ねて描かれた曲線の2つの異なる終点をリンクするピクセルの集合を計算することを含み、その集合の隣接するピクセルは共通の辺を有し、連続したポリラインを構築するステップにおいて、ポリラインの各線は、計算されたピクセルの集合のピクセルの1つの辺である。

・表示された画像のピクセルを識別するステップの後、各々識別されたピクセルをマーク付けするステップであって、マーク付けは当該各識別されたピクセルに対するポリラインの位置を示す、ステップ。

10

・各識別されたピクセルをマーク付けするステップは、RGB A色空間で符号化された色で、各識別されたピクセルを色付けすることによって行われる。第1のRGB A値は、ポリラインがピクセルの右側に位置することを示し、第2のRGB A値は、ポリラインがピクセルの左側に位置することを示し、第3のRGB A値は、ポリラインがピクセルの上側に位置することを示し、第4のRGB A値は、ポリラインがピクセルの下側に位置することを示す。

・各ピクセルに色をつけることは、累積的である。

・曲線と画像の縁とで囲まれた全ての閉じられたエリアを計算するステップは、領域拡張アルゴリズムを用いることによって実行され、ピクセルの領域加入基準は、連続したポリ

20

・領域拡張アルゴリズムは、以下のステップを含む。（i）画像の第1の閉じられたエリアに属する第1のピクセルを選択するステップ、（ii）基準を満たす選択された第1のピクセルに隣接するピクセルを、選択されたピクセルへのマーク付けを用いて、識別するステップ、（iii）ステップ（ii）で識別された各隣のピクセルを、第1の選択されたピクセルとして、選択、処理されるピクセルのリストに追加するステップ、（iv）選択された第1のピクセルをチェック済みとしてマーク付けし、選択された第1のピクセルを第1の閉じられたエリアに属するピクセルのリストに追加するステップ、（v）選択すべきピクセルのリストから第2のピクセルを選択し、ステップ（ii）～（iv）を第2のピクセルに対して実行するステップ、（vi）選択すべきピクセルのリストの全てのピクセルが選択されるまで、ステップ（v）を繰り返すステップ。

30

・（vii）画像の第2の閉じられたエリアに属する第3のピクセルを識別するステップであって、第3のピクセルはステップ（iv）でマーク付けされていない、ステップ、（viii）ステップ（ii）～（vi）を実行するステップ、（ix）画像の全てのピクセルがチェック済みとしてマーク付けされるまでステップ（vii）～（viii）を繰り返すステップ。

・連続したポリラインを構築するステップから開始され、実行されるステップは、マスクに対して実行される。マスクは、表示された画像と同じサイズ及び同じ数のピクセルを有する画像である。

・マスクは、空白画像として初期設定され、チェック済みとしてマーク付けされたピクセルは、閉じられたエリアに属するピクセルリストに追加される時、黒で色付けされる。

40

・計算された閉じられたエリアは、四分木データ構造に記憶される。

・全ての閉じられたエリアを計算するステップの後、少なくとも1つの閉じられたエリアを個々に選択するステップと、選択された各々閉じられたエリアに関して独立した画像を生成するステップ。

【0012】

上記方法を行う命令を含むコンピュータプログラムをさらに提供する。

【0013】

コンピュータプログラムを記録したコンピュータ可読記憶媒体をさらに提供する。

【0014】

50

コンピュータプログラムを記録したメモリとグラフィックユーザインタフェースとに接続されたプロセッサを含むシステムをさらに提供する。

【図面の簡単な説明】

【0015】

発明の実施形態を、添付の図面を参照して、例を挙げて記載するが、発明はそれらの例に限定されない。

【図1】画像を分割する既知の方法の例を示す図である。

【図2】画像を分割する既知の方法の例を示す図である。

【図3】画像を分割する既知の方法の例を示す図である。

【図4】画像を分割する既知の方法の例を示す図である。

10

【図5】画像を分割する既知の方法の例を示す図である。

【図6】画像を分割する既知の方法の例を示す図である。

【図7】本発明の例を示すフローチャートである。

【図8】本発明で用いられる伝播アルゴリズムの例を示すフローチャートである。

【図9】本発明に係る、分割された画像から新しい画像を生成する操作を示すフローチャートである。

【図10】画像に重ねてユーザが描いた曲線を示す図である。

【図11】本発明に係る、曲線からポリラインを計算する例を示す図である。

【図12】本発明に係る、曲線からポリラインを計算する例を示す図である。

【図13】本発明に係る、曲線からポリラインを計算する例を示す図である。

20

【図14】本発明に係る、閉じられたエリアの計算を示す図である。

【図15】本発明に係る、閉じられたエリアの計算を示す図である。

【図16】本発明に係る、閉じられたエリアの計算を示す図である。

【図17】本発明に係る、閉じられたエリアの計算を示す図である。

【図18】本発明に係る、閉じられたエリアの計算を示す図である。

【図19】本発明に係る、閉じられたエリアの計算を示す図である。

【図20】本発明に係る、閉じられたエリアの計算を示す図である。

【図21】本発明に係る、閉じられたエリアの計算を示す図である。

【図22】本発明を行うシステムの例を示す図である。

【発明を実施するための形態】

30

【0016】

図7のフローチャートを参照して、画像を分割するコンピュータ実行方法を提案する。画像の分割とは、分割可能な少なくとも2つの部分に画像を分けることを意味する。方法は、縁を含む画像を表示するステップを含む。次に、曲線を、表示された画像に重なるように描く。次に、曲線と画像の縁とによって囲まれた全ての閉じられたエリアを計算する。各計算された閉じられたエリアは、他の計算された閉じられたエリアから独立しており、個々に選択可能である。

【0017】

本発明の方法によって、画像はより効率良く分割される。実際、閉じられたエリアの数（すなわち、サブパートの数）は、ユーザが描いた曲線と画像の縁からのみ計算され、曲線の開始点と終点のリンク付けは必要とされない。興味深いことに、閉じられたエリアの数を計算する時、画像の縁が使用される。こうすると、ユーザは、画像を半分に切りたい場合、画像の最上部から最下部に線を引くだけでよい。閉じられたエリアの数を計算する時、画像の縁を使用するので、少なくとも1つの閉じられたエリア（ユーザが描いた曲線が交点を含まない時の、完全な画像）から、描画の閉じた形の数に応じて、幾つかの閉じられたエリアまでが存在する。従って、ユーザの意図が変更されることなく、同時に、「内側」「外側」の判定がないので閉じられたエリアの計算に必要なコンピューティング資源が低減される。さらなる利点を以下に記載する。

40

【0018】

方法は、コンピュータで実行される。これは、方法のステップ（または、ほぼ全てのス

50

テップ)は、少なくとも1つのコンピュータまたは任意の同様のシステムによって実行されることを意味する。従って、方法のステップは、おそらく、全自動または半自動で、コンピュータによって実行される。例においては、方法のステップの少なくとも一部のトリガは、ユーザとコンピュータのインタラクションを通して行われてよい。必要とされるユーザとコンピュータとのインタラクションのレベルは、想定される自動化のレベルによって決められてよく、ユーザの要望を実行する必要とバランスをとってよい。例においては、このレベルは、ユーザが決定してよく、及び/または、予め決定されてよい。

【0019】

例えば、画像に重ねて曲線を描くステップ(S20)は、ユーザアクションによってトリガされる。少なくとも1つの閉じられたエリアを個々に選択するステップ(S110)も、ユーザアクションで行うことができる。

10

【0020】

方法をコンピュータで実行する典型的な例は、この目的のために適合されたシステムを用いて方法を行うことである。システムは、メモリとグラフィカルユーザインタフェース(GUI)に接続されたプロセッサを含んでよく、メモリは、方法を実行する命令を含むコンピュータプログラムを記録している。メモリは、データベースも記憶してよい。メモリは、このような記憶に適合した任意のハードウェアで、幾つかの物理的に異なる部分(例えば、プログラム用の部分、及び、おそらく、データベース用の部分)を含み得る。

【0021】

「データベース」とは、検索及び読み取りのために体系化された任意のデータ(すなわち、情報)のコレクションを意味する。データベースは、メモリに記憶されて、コンピュータによる迅速な検索と読み取りを可能にする。データベースは、様々なデータ処理操作と共に、データの記憶、読み取り、修正、削除を容易にするように構造化されている。データベースは、レコードに分けることができるファイルまたはファイルのセットからなり、各レコードは、1つまたは複数のフィールドからなるものとすることができる。フィールドは、データ記憶装置の基本単位である。ユーザは、主にクエリを通してデータを読み取ってよい。キーワード及びソートコマンドを用いて、ユーザは、使用中のデータベース管理システムの規則に従って、多くのレコードのフィールドを迅速に検索、並び替え、グループ化、選択して、特定のデータ集合体を読み取ることができ、または、特定のデータ集合体に関するレポートを生成することができる。

20

30

【0022】

方法は、一般的に、画像を操作する。画像は、二次元であってよい。画像は、三次元、例えば、コンピュータ支援設計(CAD)システムにおいて三次元シーンで示された三次元モデルオブジェクトの表現であってよい。本発明においては、画像という語は、画像の数値表現であるデジタル画像を指す。数値表現は、当分野で既知のように、画像データを2進数のセットで表現することを含む。画像は、ラスタ画像またはベクタ画像であってよいが、それらに限定されない。ラスタ画像は、画素(ピクセル)の有限集合として記憶され、各ピクセルはデジタル値を表す。ベクタ画像は、画像を表現する数式に基づいた幾何プリミティブの集合として記憶される。

【0023】

表示される時、ラスタ画像とベクタ画像とは、ピクセルの集合として表現される。この目的のために、ベクタ画像は、表示前に、ラスタ画像に変換される。これは、ラスタ化と呼ばれる。ピクセルは、画像の最小の個々の要素である。ピクセルは、画像のある特定の点で、表色系の所与の色の明度を表す量子化された値を保持する。実際には、画像は、コンピュータシステムが実行する画像ビューアソフトウェアによって表示される。画像ビューアソフトウェアは、本発明の方法を実行することができる、または、方法は、独立したコンピュータプログラムで実行することができる。

40

【0024】

ラスタ画像もしくはベクタ画像、または、ピクセルで表現された任意の他の種類の画像のピクセルの集合は、一般的に、ピクセルからなる長方形格子を形成し、各ピクセルも長

50

方形である。各ピクセルは、隣のピクセルと連結される。ピクセルの連結性とは、当分野で既知のように、二次元または三次元画像において、ピクセルと隣接するピクセルとの関連の仕方である。例えば、画像のピクセルが長方形の場合、各ピクセルは、4つの辺を有し、1つまたは複数の辺を隣のピクセルと共有する。

【0025】

ピクセルの格子(grid)は、長方形格子に限られず、当分野で既知のように、1つのピクセルが6つまでの隣接するピクセルを有し得る六角形格子または長手積みの長方形格子であってもよい。以下では、長方形のピクセルからなる長方形格子を用いて表現される画像で最も多いケースを、説明目的のみで記載する。

【0026】

図22は、発明の方法を実行するシステムの例を示す。システムは、典型的には、コンピュータ、例えば、パーソナルコンピュータである。図22のコンピュータは、内部通信バス1000に接続された中央処理装置(CPU)1010と、同様にバスに接続されたランダムアクセスメモリ(RAM)1070とを備える。コンピュータは、バスに接続されたビデオランダムアクセスメモリ1100に関連付けられたグラフィック処理装置(GPU)1110をさらに備える。ビデオRAM1100は、フレームバッファとしても当分野では知られている。大容量記憶装置コントローラ1020は、ハードドライブ1030等の大容量メモリデバイスへのアクセスを管理する。コンピュータプログラム命令及びデータを有形に実現するのに適した大容量メモリデバイスは、例を挙げると、EPROM、EEPROM、及び、フラッシュメモリデバイス等の半導体メモリデバイス、内部ハードディスク、及び、リムーバブルディスク等の磁気ディスク、光磁気ディスク、並びに、CD-ROMディスク1040を含む、あらゆる形式の不揮発メモリを含む。前述のいずれも、特別に設計されたASIC(特定用途向け集積回路)によって補完されてよい、または、当該ASICに組み込まれてよい。ネットワークアダプタ1050は、ネットワーク1060へのアクセスを管理する。コンピュータは、カーソル制御装置、キーボード等の触覚装置1090も備えてよい。カーソル制御装置は、コンピュータで用いられて、ディスプレイ1080の任意の所望の位置にユーザがカーソルを選択的に置くのを可能にする。さらに、カーソル制御装置は、ユーザが、様々なコマンドを選択し、制御信号を入力するのを可能にする。カーソル制御装置は、システムへの入力制御信号のための多くの信号生成装置を含む。典型的には、カーソル制御装置は、マウスであってよく、マウスのボタンは、信号生成に用いられる。選択的に、または、追加的に、コンピュータシステムは、タッチパッド及び/またはタッチスクリーンを備えてよい。

【0027】

本発明は、コンピュータプログラムによって実行することができる。コンピュータプログラムは、コンピュータによって実行可能な命令を含み、命令は、上記システムに方法を実行させる手段を含む。プログラムは、システムのメモリを含む任意のデータ記憶媒体に記録可能であってよい。プログラムは、例えば、デジタル電子回路で、または、コンピュータのハードウェア、ファームウェア、ソフトウェア、もしくは、それらの組み合わせで、実行されてよい。プログラムは、装置、例えば、プログラム可能なプロセッサによって実行するために機械可読記憶装置で有形に具体化される製品として、実行されてよい。方法のステップは、命令のプログラムを実行するプログラム可能なプロセッサによって実行されて、入力データに関する操作を実行し、出力を生成することによって方法の機能を実行してもよい。プロセッサは、従って、プログラム可能であってよく、データ記憶システム、少なくとも1つの入力装置、及び、少なくとも1つの出力装置に結合されて、それらからデータ及び命令を受信し、それらにデータ及び命令を送信してよい。アプリケーションプログラムは、必要に応じて、高水準手続き型プログラミング言語もしくはオブジェクト指向プログラム言語で、または、アセンブリ言語もしくは機械語で実行されてよい。いずれの場合でも、言語は、コンパイラ型言語であってもよく、インタープリタ型言語であってもよい。プログラムは、完全インストールプログラムであってもよく、更新型プログラムであってもよい。プログラムのシステムへの適用によって、いずれの場合でも、方法

10

20

30

40

50

を行う命令となる。

【0028】

図7を参照すると、ステップS10において、画像が、例えば、システムによって表示される。画像を表示するとは、ピクセルの集合が、表示装置、例えば、図22のディスプレイ1080上に表現されることを意味する。画像を表示するプロセスは、当分野で既知のように行われる。画像は縁を含む。画像の縁は、ピクセルの連結性が部分的に満たされている画像のピクセルの集合である。例えば、長方形のピクセルは、当該ピクセルが3つ以下の隣り合うピクセルを有する場合、画像の縁に属する。画像の表示は、ユーザアクションで、例えば、ユーザが画像を選択すると、トリガされてよい。

【0029】

次に、ステップS20において、ステップS10で表示された画像に、例えば、ユーザによって曲線が描かれる。曲線という語は、線分に類似しているが直線である必要はないオブジェクトを意味する。線分は、2つの異なる終点によって区切られ、その終点間の線上のあらゆる点を含む線の一部である。本発明において、曲線という語は、自由な形という表現と同義である。あるいは、所定の形を画像に描いてよい。任意の所定の形を用いてよい。

【0030】

曲線の描画は、ユーザによって行われる。ユーザは、当分野で既知のように進める。例えば、ユーザは、マウス、スタイラス等の触覚装置をタッチディスプレイ上で用いることができる。

【0031】

画像に曲線を描くという表現は、ユーザが画像表現（例えば、画像が表現されるコンピュータ画面）とインタラクトし、かつ、画像は修正されない（例えば、画像のピクセルは色付けされない）ことを意味する。ユーザに示される曲線の表現は、一時的なものであり、画像表現に重ね合わせられる。よって、曲線を画像に重ねて描くとも言うことができる。

【0032】

ここで、図10を参照すると、ユーザが描いた曲線が示されている。ユーザが描いた曲線の表現100は、画像上に重ねられ、画像は修正されない。画像の縁は、黒の実線で表される。興味深いことに、曲線の一部は画像の外側にあることがわかる。これは、ユーザがコンピュータ画面上に曲線を描き、画像に直接、描いていないので、可能である。あるいは、画像は、画像より大きい画像キャンバス（図示せず）に配置されてよい。ユーザは、（全体的または部分的に）画像の外に曲線を描くことができる。曲線の描画は画像の縁に制限されない。

【0033】

次に、ステップS30において、マスクを、例えば、システムによって計算する。マスクは、典型的には、ユーザが曲線の描画をやめると、計算される。マスクは、表示された画像と同じサイズ及び同じ数のピクセルを有する画像である。例えば、ステップS10で表示された画像が、 $(x \times y)$ ピクセルのサイズを有し、 $n = x \times y$ 個のピクセルを含む場合、マスクは、同じサイズの $(x \times y)$ ピクセルを有し、同じ数の n 個のピクセルを有するように、計算される。言い換えると、マスクは、ステップS10に表示された画像と同じサイズと解像度を有する。

【0034】

マスクの1つのピクセルと画像の1つのピクセルは1対1で対応し、その結果、マスクの1つのピクセルに対して行った操作（単数または複数）の結果は、画像の1つの対応するピクセルに行われる。

【0035】

実際には、マスクは、空白画像として初期設定され、そのピクセルの不透明度値は、例えば、RGBA（赤、緑、青、アルファ）色空間において、各ピクセルに関して同じである。これによって、マスクの各ピクセルのカラーチャネルを均一に設定することができる

10

20

30

40

50

。マスクは、透明として初期設定されるのが好ましい。マスクは、実行されると、ユーザに表示されず、本方法で使用される内部構造とみなされるべきであることは、理解されたい。表示されないのは、ユーザは、マスクと直接インタラクトしない。

【0036】

次に、図7のステップS40において、連続したポリラインが、画像に描かれた曲線から、例えば、システムによって構築される。連続したポリラインは、1つまたは複数の線分（線とも言う）で構成され、各線分の終点は特定される。ポリラインは、1つの点に連結される2つの異なる終点を除いては、ポリラインの各点が2つの線分に連結されるので、連続している。連続したポリラインを構築することは、描かれた曲線の各点のピクセル位置を計算することを意味する。

10

【0037】

曲線の部分（単数または複数）が図10に示すように画像の外側に描かれた場合、これらの部分（単数または複数）は考慮されない。これは、図14に示されており、3つの連続したポリライン140、142、144が図10の曲線から構築されている。画像の外側の（すなわち、画像に重ならない）曲線の部分は無視され、システムは3本の独立した曲線が画像に重ねて描かれていると認識する。図14では、1つの連続したポリライン140、142、144が、各独立した曲線に対して計算されている。実際、曲線の無視された部分は、（i）画像の外側に位置し、（ii）曲線と画像の縁との第1の交点から曲線と画像の縁との第2の交点に延びる部分である。言い換えると、（画像の内側と外側に描かれる）曲線の保持される部分は、（i）画像の内側で、（ii）（a）曲線と画像の縁との第1の交点から曲線と画像の縁との第2の交点に、または、（b）曲線と画像の縁との第1の交点から画像の内側の曲線の終点に、延びる部分である。例えば、ステップS40で構築された連続したポリライン142は、画像の縁に2つの終点があり、連続したポリライン140、144は、縁に1つだけ終点があり、第2の終点は、画像の内側にある。曲線が全体的に画像に重ねて描かれる場合、完全な曲線が保持されることは理解されよう。

20

【0038】

画像に描かれた曲線のピクセル位置は、当分野で既知のように計算されてよい。図8、図9に示された実施形態においては、当分野で広く知られているプレゼンハムのアルゴリズムの派生形（derivation）が、これら全てのピクセル位置をリンクさせ、ピクセルの連続した線を得るために用いられる。

30

【0039】

プレゼンハムのアルゴリズムは、2つの離れたピクセルをリンクする最短の経路を計算する。これは、図11に示されており、灰色のピクセルが、プレゼンハムのアルゴリズムを用いて、画像に描かれた曲線113の2つの異なる終点の位置を表す2つのピクセル110と112をリンクして得られたピクセルである。オリジナルのプレゼンハムのアルゴリズムにおいては、頂点のみを共有する隣り合うピクセル114、116があり、他の隣り合うピクセル116、118は、他のピクセルと少なくとも1つの辺を共有することが分かる。

【0040】

40

図12を参照すると、プレゼンハムのアルゴリズムの派生形が示されている。異なる終点を表す2つのピクセル110、112をリンクするピクセル集合において、各ピクセルは、オリジナルのプレゼンハムのアルゴリズムで識別されたピクセルの中のピクセルと、少なくとも1つの共通の辺を共有する。言い換えると、隣り合うピクセル間に許された唯一のピクセルの連結性は、辺の連結性である。実際には、図12に示す結果は、オリジナルのプレゼンハムのアルゴリズムで選択されなかった（i）1つの頂点のみを共有する2つのピクセルと共通の辺を共有し、（ii）描かれた曲線上に位置する、ピクセル120、122、124、126を識別することによって得られる。

【0041】

ここで、図13を参照すると、（点線で表された）連続したポリライン130は、ピク

50

セルの連続した辺から計算することができ、各辺がポリラインの線である。連続した辺は、(i) 曲線の 2 つの異なる終点をリンクし、オリジナルのプレゼンハムのアルゴリズムから得られたピクセル集合、及び / または、(i i) 2 つの隣り合うピクセル間に辺の連結性のみがあることを確実にするために追加された新しいピクセルから得られる。図 1 3 において、連続したポリラインを形成するピクセルの辺は、ポリラインが全てのピクセルの基準点をリンクするように選択される。基準点は、ピクセルの 4 つの頂点の 1 つである。ピクセルの基準点の選択は任意である。この例においては、ピクセルの左下の頂点が基準点として選択されている。任意の頂点を基準点に選ぶことができ、全てのピクセルが、この選択に従う。図 1 3 においては、各ピクセルは、その左下の頂点を基準点とし、その結果、ポリライン 1 3 0 は、ピクセル 1 1 0 の頂点 1 1 1 から開始し、ピクセル 1 1 2 の頂点 1 1 9 で終了する。

10

【 0 0 4 2 】

ポリラインの計算は、一般的には、ステップ S 3 0 で計算されたマスク上で行われる。これは、画像の修正を避け、従って、コンピューティング資源の消費を低減する。図 1 6 は、図 1 1 の曲線 1 1 3 から得られたポリライン 1 6 0 を示す。

【 0 0 4 3 】

連続したポリラインを構築するステップは、画像の縁に関するポリラインを構築することをさらに備える。実際には、画像は、ピクセルからなる長方形格子を形成し、従って、画像の縁は長方形である。このような場合、画像の 4 つの辺の各辺はポリラインである。画像の辺に関するポリラインの構築は、画像に関する情報を用いる。画像の各頂点の座標を計算し、その頂点を線分にリンクさせる。これは図 1 5 に示される。図 1 5 は、それぞれ、頂点 1 5 0 - 1 5 2、頂点 1 5 2 - 1 5 4、頂点 1 5 4 - 1 5 6、頂点 1 5 6 - 1 5 0 を接続する画像の縁の 4 つのポリライン 1 5 1、1 5 3、1 5 5、1 5 7 が、点線で表されている以外は、図 1 4 と同じである。画像の縁のポリラインを構築することによって、画像の縁が境界とみなされるので、有利なことに、ユーザによる閉じられたエリアの作成が容易になる。例えば、ユーザは、画像を半分に切りたい場合、単に、画像の最上部から最下部に線を引いて、画像の左もしくは右、または、両方の部分を保持したいかを選択する。よって、ユーザは、画像を半分に切りたい場合、図 5、図 6 を参照して記載したように画像の周りに円形を描く必要はない。

20

【 0 0 4 4 】

次に、図 7 のステップ S 5 0 で、少なくとも 1 つの辺が計算された連続したポリラインの線の 1 つである画像の全てのピクセルを識別する。連続したポリラインが、描かれた曲線からマスク上に構築された場合、マスクのピクセルを識別することを理解されよう。図 1 7 は、図 1 6 のポリラインと、識別されたピクセルとを示す。このステップは、ポリラインの両側にある 2 つのピクセルからなるポリラインの縁を作成し、この 2 つのピクセルの縁が、閉じられたエリアに属するピクセルの識別を容易にし、閉じられたエリアの計算は容易になり、ポリラインに接触するピクセルが、正確な閉じられたエリアに確実に割り当てられるようにする。

30

【 0 0 4 5 】

次に、ステップ S 6 0 において、ステップ S 5 0 で識別された各ピクセルは、当該ピクセルに対するステップ S 4 0 で計算したポリラインの位置を識別するために、マーク付けされる。逆の操作、すなわち、各ピクセルに関して、ポリラインに対するピクセルの位置を決定できることは理解されよう。ピクセルをマーク付けすることは、情報をピクセルに追加することを意味し、追加の情報をピクセルデータと共に記憶してよく、追加の情報は、新しいデータ構造で記憶されてよい。マーク付けは、ピクセルデータと共に記憶されるのが好ましく、これによって、本方法を実行するシステムに関する最適な計算結果が得られる。

40

【 0 0 4 6 】

ピクセルに対するポリラインの位置は、ピクセルの基準点を用いて推定する。ピクセルの左下の頂点が、基準点として選択され、(典型的には、画像基準の原点は、画像の左下

50

にあるピクセルの左下の頂点である)画像基準における座標 (a, b) を有する場合、他の頂点は、座標 $(a + 1, b)$ 、 $(a, b + 1)$ 、 $(a + 1, b + 1)$ を有する。このようにして、各辺、位置に関連付けられた2つの座標、すなわち、ピクセルの右の辺または左の辺または上の辺または下の辺によって定義される。ピクセルに対するポリラインの位置の決定は、簡単である。 (i) ポリラインの線がピクセルの右の辺である場合、ポリラインはピクセルの右側にあり、 (ii) ポリラインの線がピクセルの左の辺である場合、ポリラインはピクセルの左側にあり、 (iii) ポリラインの線がピクセルの上の辺である場合、ポリラインはピクセルの上側にあり、 (iv) ポリラインの線がピクセルの下の辺である場合、ポリラインはピクセルの下側にある。

【0047】

10

図7、図8に示す実施形態においては、マーク付けは、情報を符号化するためにマスクのRGB Aチャンネルを用いて行われる。言い換えると、マーク付けは、ポリラインに対する位置に従って、ステップS50で識別された各ピクセルに色付けをすることである。ここで、頭字語RGB Aは、Red(赤)、Green(緑)、Blue(青)、Alpha(アルファ)色空間を指し、各色(赤、緑、青)と透明度(A)は、nビットで定義される、例えば、1つのピクセルのRGB A情報は、位置を符号化するために、チャンネル毎に $n = 8$ ビットの、1つの32ビット符号なし整数を提供することができる。

【0048】

よって、第1のRGB A値(例えば、透明な緑、 $R = 0$ 、 $G = 255$ 、 $B = 0$ 、 $A = 0$)は、ポリラインがピクセルの右側に位置することを示し、第2のRGB A値(例えば、透明な白、 $R = 0$ 、 $G = 0$ 、 $B = 0$ 、 $A = 255$)は、ポリラインがピクセルの左側に位置することを示し、第3のRGB A値(例えば、透明な赤、 $R = 255$ 、 $G = 0$ 、 $B = 0$ 、 $A = 0$)は、ポリラインがピクセルの上側に位置することを示し、第4のRGB A値(例えば、透明な青、 $R = 0$ 、 $G = 0$ 、 $B = 255$ 、 $A = 0$)は、ポリラインがピクセルの下側に位置することを示す。RGB A値の使用は、ピクセルの色付けを意味することは理解されよう。よって、ステップS60を行う時、(空白画像として初期設定された)マスクは、これらのピクセルに対するポリラインの位置を示す色付けされたピクセル及び透明なピクセルを含む。

【0049】

興味深いことに、ピクセルにマーク付けするプロセスは、累積的である。これは、ピクセルは複数回マーク付けされてよいこと、及び、各連続したマーク付けの情報を読み取ることができることを意味する。例えば、ポリラインがあるピクセルの上側と右側にある場合、そのピクセルは、2回マーク付けされ、最後のマーク付け(すなわち、2回のマーク付けの結果)は、ポリラインがピクセルの上側と右側にあるという情報を提供する。この例においては、第3のRGB A値が、ピクセルの上側に位置するポリラインが $R = 0$ 、 $G = 255$ 、 $B = 0$ 、 $A = 0$ に設定されていることを示し、第1のRGB A値が、ピクセルの右側に位置するポリラインが $R = 255$ 、 $G = 0$ 、 $B = 0$ 、 $A = 0$ に設定されていることを示した場合、ピクセルのマーク付けは、 $R = 255$ 、 $G = 255$ 、 $B = 0$ 、 $A = 0$ に設定される。ピクセルは、透明な黄色で色付けされる。

【0050】

図17の識別されたピクセルは、色付けされる(ここでは、色は、各ピクセルに書かれた数字で表される)。(数字1で表される)青のピクセルは、ポリラインがピクセルの下側に位置することを示し、(数字4で表される)赤のピクセルは、ポリラインがピクセルの上側に位置することを示し、(数字3で表される)薄い青のピクセルは、ポリラインがピクセルの下側及び左側に位置することを示し、(数字2で表される)黄色のピクセルは、ポリラインがピクセルの上側及び右側に位置することを示す。数字2と3のピクセルは、従って、マーク付けが累積したピクセルである。

【0051】

次に、ステップS70~S100において、曲線と画像の縁とによって囲まれた全ての閉じられたエリアを計算する。全ての閉じられたエリアという表現は、1つまたは複数の

50

閉じられたエリアを計算することを含む。閉じられたエリアは、ユーザが選択したい画像上の範囲を定められた表面であり、当該表面は、ユーザが描いた曲線から構築された連続したポリライン、及び/または、画像の縁から構築された連続したポリラインによって定められる。各閉じられたエリアは、画像の他の閉じられたエリアから独立している。これは、2つの閉じられたエリア間に交点がないということ、すなわち、画像のピクセルは1つの閉じられたエリアにのみ属することができるということを意味する。1つまたは複数の閉じられたエリアは個々に選択可能である、すなわち、1つの閉じられたエリアの選択は、第2の閉じられたエリアの選択を伴わない。閉じられたエリアを計算することは、画像のピクセルの集合を識別し、その集合内の各ピクセルは、その集合内に少なくとも1つの隣接するピクセルを有することを意味する。

10

【0052】

曲線と画像の縁とによって囲まれた全ての閉じられたエリアを計算することは、領域拡張アルゴリズムの使用によって行うことができる。領域拡張は、シード点を選択し、この選択したシード点から領域を拡張する画像セグメンテーション法である。選択したシード点の隣の点が、少なくとも1つの加入基準を順守している場合、領域は拡張される、そして、隣の点が少なくとも1つの加入基準を順守している場合、その点自体がシード点として分類される。各新しいシード点に関して、チェックすべきシード点がなくなるまで、プロセスは繰り返される。

【0053】

図8は、本発明と共に用いてよい領域拡張アルゴリズムの例を示す。領域加入基準は、選択したピクセルの隣のピクセルは、ポリラインを超えないということである。よって、領域加入基準は、ポリライン（単数または複数）を超えない隣のピクセルを保存することを目的とする。

20

【0054】

ステップS700において、マスクのピクセルを選択する。選択されたピクセルは、方法のこのステップではまだ分かってない閉じられたエリアに属する。

【0055】

ここで、図18を参照すると、図17のポリラインとマーク付けされたピクセルが、ポリラインの新しい線と2つのピクセル184、185が追加されて少し修正されて示されている。ピクセルの1つ180が選択された、例えば、ピクセルのランダムな選択が本分割方法を実行するシステムによって行われる。あるいは、ユーザがピクセルの1つを選択してよい。

30

【0056】

図8に戻ると、ステップS710において、選択したピクセルに隣接するピクセルをチェックして、これらの隣接するピクセルが加入基準を満たしているか否かを識別する。ステップS700で選択されたピクセル180に隣接するピクセルのうち、既に構築済みのポリラインに入っている隣接するピクセルは拒絶される、すなわち、ステップ700で選択されたピクセルが属する現在の閉じられたエリアに属しないと見なされる。言い換えると、ポリラインを超えない1つまたは複数の隣接するピクセル、すなわち、選択したピクセルの閉じられたエリアに属するピクセルを識別する。隣接するピクセルがポリラインを超えるか否かの判定は、選択したピクセルに対してであることは理解されよう。基準を満たすピクセルの識別は、従って、拒絶されるピクセル（ポリラインを超えるピクセル）を識別すること、または、保持すべきピクセル（ポリラインを超えないピクセル）を識別することである。

40

【0057】

加入基準の検証は、選択したピクセルへのマーク付けを用いて行われるのが好ましい。マスク上の選択されたピクセルが空白の場合、その辺のいずれもポリラインの線にあたらないということを意味するので、その隣接するピクセルはどれも（選択したピクセルに対して）ポリラインを超えない。マスク上の選択されたピクセルがマーク付けされている場合、マーク付けは、この選択されたピクセルに対するポリラインの位置を示し、そのポリ

50

ラインの線として用いられた辺を共有する隣接するピクセルは、ポリラインを超えるとみなされて、破棄される。

【 0 0 5 8 】

次に、ステップ S 7 2 0 において、ステップ S 7 1 0 で識別された隣接するピクセルが、ピクセルのリストに追加され、現在選択されているピクセルと同じように選択、処理される。これは、領域拡張アルゴリズムの正確な伝播を確実にする。

【 0 0 5 9 】

図 1 8 に戻ると、この例におけるピクセルの連結性は、(画像の縁、または、マスクの縁のピクセルを除いた) 画像の各ピクセルの 4 つの隣接するピクセルによる。よって、選択されたピクセル (図 1 8 の網掛けされたピクセル 1 8 0) は、4 つの隣接するピクセルを有する。網掛けされたピクセルは色付けされておらず、それは、ピクセル 1 8 0 の辺のいずれも構築済みのポリラインの線を含まないことを意味する。保持すべきピクセルとして識別されたピクセルは、ピクセルリストに追加され、ステップ S 7 0 0 で選択されたピクセルと同じように選択、処理される。この例においては、選択したピクセル 1 8 0 の 4 つの隣接するピクセルは、基準を満たすので、保持され、リストに追加される。

【 0 0 6 0 】

次に、ステップ S 7 3 0 において、選択されたピクセルは、チェック済みとしてマーク付けされる。これは、R G B A チャンネルを用いて行うことができる。例えば、ピクセルは黒で色付けされる。そして、マーク付けされたピクセルは、現在計算されている閉じられたエリアに属するピクセルのリストに追加される。

【 0 0 6 1 】

次に、ステップ S 7 4 0 において、新しいピクセルが、ステップ S 7 2 0 で生成されたリストから選択される。このピクセルは、ステップ S 7 1 0 で選択されたピクセルと同じように処理される。よって、ステップ S 7 1 0 ~ S 7 4 0 が、このピクセルに対して実行される。

【 0 0 6 2 】

ここで、図 1 9 を参照すると、今度は、図 1 8 に示す選択済みのピクセル 1 8 0 に隣接するピクセルである新しく選択されたピクセル 1 8 2 に焦点を当てている。このピクセル 1 8 2 は、選択済みのピクセル 1 8 0 と同じように選択、処理されるピクセルのリストに追加された。選択されたピクセル 1 8 2 は、青でマーク付けされる、それは、(i) ポリラインが、このピクセルの下側に位置すること、(i i) (黄色で色付けされ、数字 2 で表される) 隣のピクセル 1 9 0 がポリラインを超えることを示す。よって、このピクセル 1 9 0 は、拒絶され、選択すべきピクセルのリストには追加されず、従って、ステップ S 7 0 0 で選択されたピクセルと同じ閉じられたエリアには属さないとして、マーク付けされない。対照的に、2 つの他の隣接するピクセルは、選択すべきピクセルのリストに追加される。ピクセル 1 8 0 は、チェック済みとして既にマーク付けされているので、選択、処理されるピクセルのリストに追加されないことは理解されよう。このようにして、ステップ S 7 1 0 ~ S 7 6 0 が、収束して解決に至るまで繰り返される。

【 0 0 6 3 】

図 8 に戻る。ステップ S 7 6 0 において、選択すべきピクセルリストの全てのピクセルが確かに処理されたと判定される。従って、リストに含まれた各ピクセルがチェック済みとしてマーク付け (例えば、黒に色付け) されていない間は、ステップ S 7 1 0 ~ S 7 4 0 が繰り返される (S 7 5 0) 。

【 0 0 6 4 】

プロセスのこのステップにおいて、マスクは、画像の第 1 の閉じられたエリアに属する黒で色付けされたピクセルの集合を含む。黒で色付けされたこれらのピクセルは、従って、第 1 の閉じられたエリアを形成するピクセルの集合に属する。これは、図 1 8 のポリラインと識別されたピクセルを示す図 2 0 に示され、ポリラインの上部に置かれたピクセルは全て黒で色付けされている。興味深いことに、ステップ S 5 0 で識別され、ポリラインの下に配置されたピクセルは、プロセスのこのステップではそのまま、ポリラインに接

10

20

30

40

50

触する第2の閉じられたエリアを同様に計算するために必要なピクセルへのマーク付けは変更されていない。

【0065】

実際には、閉じられたエリアが計算された後、この新しく計算された閉じられたエリアのピクセルは、データ構造に記憶される(S100)。このデータ構造は、画像の各ピクセルに対して、当該各ピクセルが属する閉じられたエリアを記憶する。用いられるデータ構造は、当分野で既知の四分木データ構造であってよい。四分木データ構造は、計算された閉じられたエリアの検索時に、パフォーマンスを有利に向上させる。

【0066】

次に、第2の閉じられたエリアに属する新しいピクセルが、マスク上で選択される(ステップS770)。この選択したピクセルは、従って、チェック済みとしてマーク付けされていない、例えば、黒で色付けされていないピクセルである。画像は、1つの閉じられたエリアのみを含むことができる、例えば、ユーザが描いた曲線は、少なくとも2回、その曲線自体と交差することはない、画像の縁と交差することはないので、画像の縁のみが閉じられたエリアを作成することは理解されよう。

【0067】

次に、ステップS710~S760が、第2の閉じられたエリアが計算されるまで、すなわち、第2の閉じられたエリアの全てのピクセルが黒で色付けされるまで、同様に繰り返される。そして、第2の閉じられたエリアのピクセルが、計算済みの第1の閉じられたエリアのピクセルが既に記憶されている同じ四分木データ構造に記憶される。

【0068】

マスクの全てのピクセルがステップS770でマーク付けされていない間、ステップS710~S760が繰り返される。

【0069】

次に、ステップS780で、マスクの全てのピクセルが黒で色付けされたことが判定される。よって、全ての閉じられたエリアが計算される。この後、マスクの各ピクセルに関して、そのピクセルと対応する画像のピクセルを識別し、その画像のピクセルは、計算済みの閉じられたエリアに従ってグループ分けされる。

【0070】

四分木データ構造が各々新しく計算された閉じられたエリアの記憶に用いられる時、このステップS780が暗に行われていることを理解されよう。実際、四分木は、新しく発見された閉じられたエリアに属するピクセルの座標を用いて書き込まれる。

【0071】

図21は、図15に表された連続したポリライン140、142、144、151、153、155、157から計算された3つの閉じられたエリア200、222、224を示す。

【0072】

図9を参照すると、本発明に係る、分割された幾つかの初期画像から新しい画像を生成する例が示されている。

【0073】

幾つかの画像が、ユーザに表示され、ユーザは、表示された画像の1つまたは複数に自由な形を描く。均等に、画像は、並べて表示されてもよく、部分的、全体的に、重なってもよい。ユーザが描いた曲線は、表示された画像に重ねて表され、画像は修正されない。

【0074】

次に、ステップS70を参照して記載したように計算された閉じた形の存在を検出する。画像の縁は、閉じられたエリアを生成し、それは、考慮してもよく、考慮しなくてもよい。例えば、ユーザが描いた曲線が、少なくとも2回、その曲線自体を交差せず、画像の縁を交差しない場合、画像の縁のみが1つまたは複数の閉じられたエリアを生成する。これらの閉じられたエリアをユーザに示さないか、または、逆に、ユーザに示すかを決定することができる。

【 0 0 7 5 】

次に、閉じられたエリアがユーザに示される。これは、例えば、各閉じられたエリアの特定のレンダリングを用いる、各閉じられたエリアの縁を形成するピクセルの特定のレンダリングを用いる等で、行うことができるが、それらに限定されない。このステップにおいて、画像はカットされる。これは、画像の各ピクセルに関して、当該各ピクセルが属する閉じられたエリアを記憶する（四分木データ構造を用いて保存可能な）最終的なマスクを用いて行われる。

【 0 0 7 6 】

その結果、ユーザは、画像の1つまたは複数の閉じられたエリアを選択した。選択は、当分野で既知のように行われる。例えば、ユーザは、選択しようとする閉じられたエリア（単数または複数）の上にマウスのカーソルを移動し、マウスをクリックして、画像のこのサブパートを選択することをシステムに知らせる。

10

【 0 0 7 7 】

新しい画像が、これらの閉じられたエリアのそれぞれに対して、最初の画像を修正することなしに作成される。これは、各選択されたサブパートのピクセルの新しい画像へのコピーを用いて行うことができる。

【 0 0 7 8 】

次に、ユーザは、プロセスを停止してもよく、初期画像は本分割方法によって修正されていないので、さらに選択を行うことができる。

【 0 0 7 9 】

20

本発明の好ましい実施形態を記載した。発明の精神と範囲を逸脱することなく、様々な変更を行ってよいことは理解されよう。従って、他の実施も請求項の範囲内である。

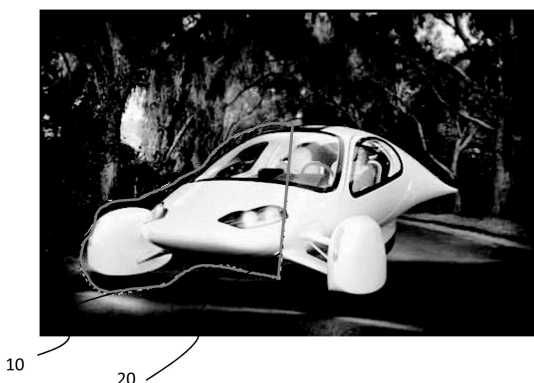
【 図 1 】



【 図 3 】



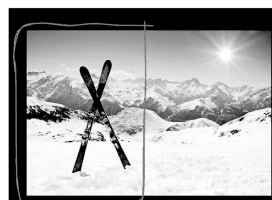
【 図 2 】



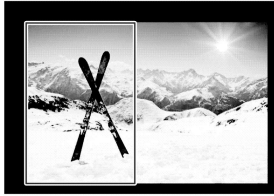
【 図 4 】



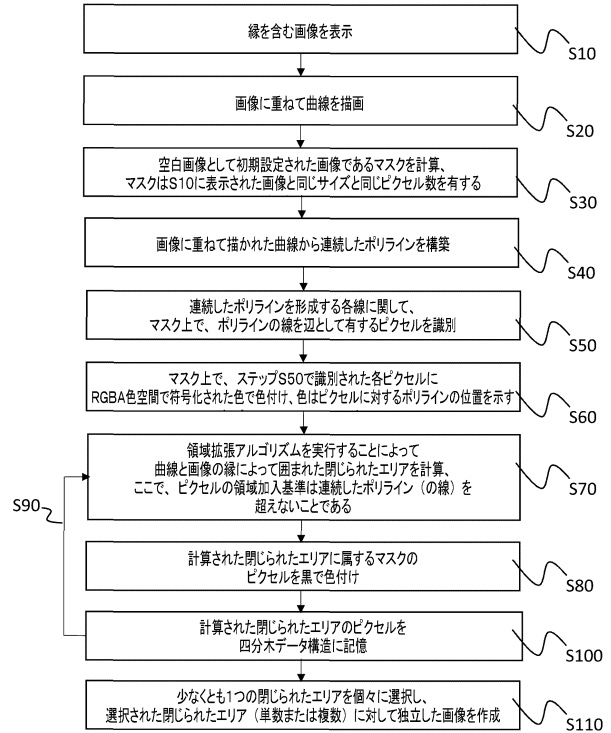
【 図 5 】



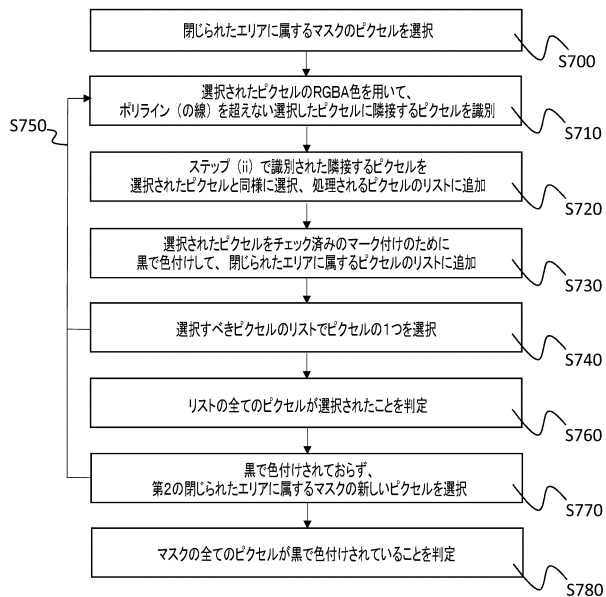
【図 6】



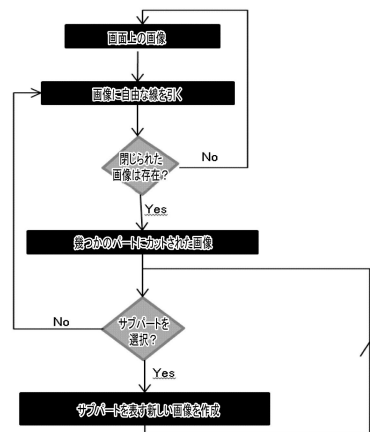
【図 7】



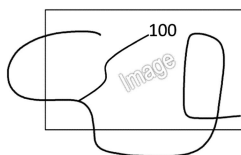
【図 8】



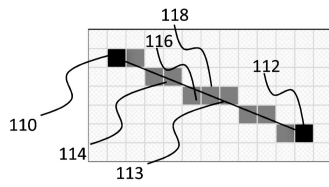
【図 9】



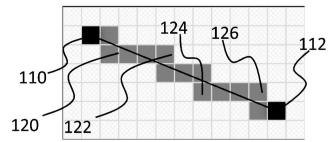
【図 10】



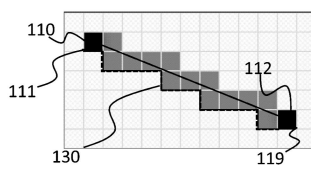
【 図 1 1 】



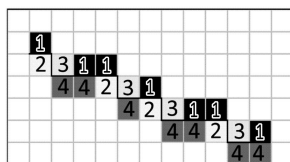
【 図 1 2 】



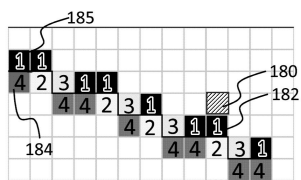
【 図 1 3 】



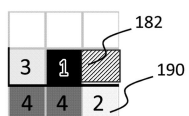
【 圖 1 7 】



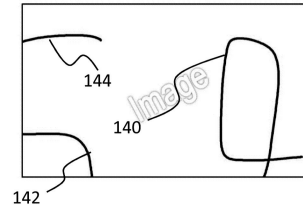
【 図 1 8 】



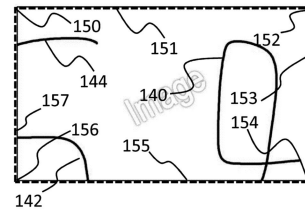
【 图 19 】



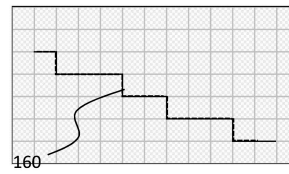
【 図 1 4 】



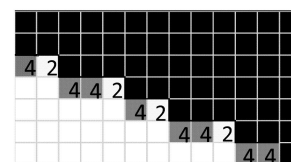
【 図 1 5 】



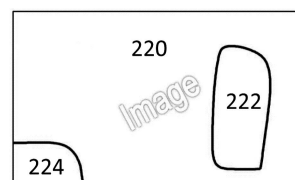
【 図 1 6 】



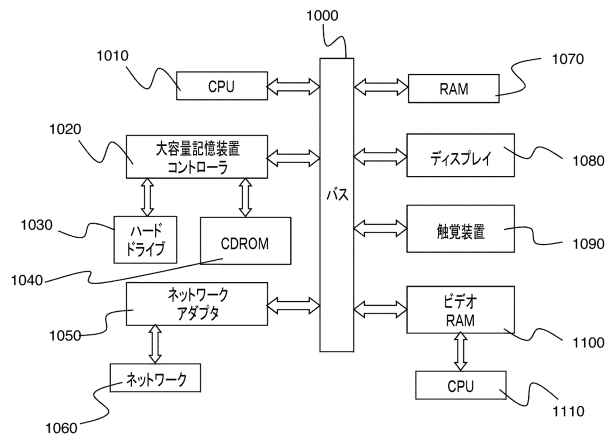
【 図 2 0 】



【 図 2 1 】



【図 22】



フロントページの続き

(72)発明者 フレデリック レッツェルター
フランス 9 1 1 6 0 ロンジュモー リュ ドゥ ペリー 5

審査官 新井 則和

(56)参考文献 特開 2 0 1 0 - 2 5 7 3 1 5 (J P , A)
特開平 0 9 - 1 2 0 2 7 8 (J P , A)
特開平 1 0 - 1 9 1 0 2 0 (J P , A)
特開 2 0 1 4 - 2 0 9 3 2 7 (J P , A)
米国特許出願公開第 2 0 0 4 / 0 1 1 4 8 0 5 (U S , A 1)
特開 2 0 1 2 - 1 2 9 6 8 5 (J P , A)
特表 2 0 0 6 - 5 0 6 1 6 3 (J P , A)
Wen-Chung Kao et al. , Hardware Engine for Real-Time Pen Tracking on Electrophoretic Displays , Journal of Display Technology , JOURNAL OF DISPLAY TECHNOLOGY, VOL. 9, NO. 3 , 2 0 1 3 年 3 月 , VOL. 9, NO. 3 , pp. 139-145 , <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6291759>

(58)調査した分野(Int.Cl. , D B 名)
G 0 6 T 7 / 0 0 - 7 / 9 0
G 0 6 T 1 1 / 8 0