

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2017-162149

(P2017-162149A)

(43) 公開日 平成29年9月14日(2017.9.14)

(51) Int.Cl.		F I		テーマコード (参考)	
G06K	9/40	(2006.01)	G06K	9/40	5B029
G06T	5/00	(2006.01)	G06T	5/00	705
					5B057

審査請求 未請求 請求項の数 10 O L (全 11 頁)

(21) 出願番号	特願2016-45470 (P2016-45470)	(71) 出願人	000001007
(22) 出願日	平成28年3月9日 (2016.3.9)		キヤノン株式会社
			東京都大田区下丸子3丁目30番2号
		(74) 代理人	100126240
			弁理士 阿部 琢磨
		(74) 代理人	100124442
			弁理士 黒岩 創吾
		(72) 発明者	山▲崎▼ 妙子
			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
		Fターム(参考)	5B029 AA02 BB02 BB17 DD07 EE08 EE13 EE16 EE17 5B057 CA01 CA08 CA12 CA16 CB02 CB06 CB12 CB16 CE02 CE12 CE17 CF02 CH18 DA08 DA17 DB02 DB05 DB08 DC04 DC14

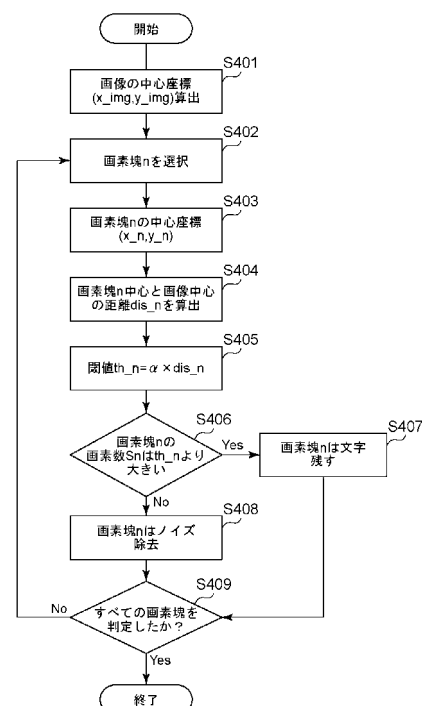
(54) 【発明の名称】 画像処理装置、プログラムおよび画像処理方法

(57) 【要約】

【課題】 周辺部にノイズの多い画像に好適なノイズ除去処理を行う。

【解決手段】 本発明は、画像内に含まれる画素塊について、当該画素塊のサイズが閾値以下であるか判定し、当該判定によりサイズが閾値以下であると判定された画素塊を、前記画像から除去する。前記閾値は、前記画素塊の位置に応じて動的に変化させる。

【選択図】 図4



【特許請求の範囲】**【請求項 1】**

画像内に含まれる画素塊について、当該画素塊のサイズが閾値以下であるか判定する判定手段と、

前記判定手段によりサイズが閾値以下であると判定された画素塊を、前記画像から除去する除去手段と、

を備え、

前記閾値は、前記画素塊の位置に応じて動的に変化させることを特徴とする画像処理装置。

【請求項 2】

前記閾値は、前記画像内の中心位置と前記画素塊の位置との間の距離に応じて動的に変化させ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】

前記画像は、文字が存在すると推定される領域の二値画像であることを特徴とする請求項 1 または 2 に記載の画像処理装置。

【請求項 4】

前記画素塊は、前記画像に対してラベリング処理を実行することにより判定され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】

前記画像に対して、背景の画素を膨張させる膨張手段を更に有し、

前記判定手段は、前記膨張手段で処理された後の画像内に含まれる画素塊について、当該画素塊のサイズが閾値以下であるか判定することを特徴とする請求項 1 乃至 4 のいずれか 1 項に記載の画像処理装置。

【請求項 6】

前記膨張手段で使用されるマスクは長方形のマスクであることを特徴とする請求項 5 に記載の画像処理装置。

【請求項 7】

前記画像に対して、背景の画素を膨張させる膨張手段と、

前記膨張手段で背景の画素が膨張された画像に対して、背景の画素を収縮させる収縮手段と、を更に有し、

前記判定手段は、前記収縮手段で処理された後の画像内に含まれる画素塊について、当該画素塊のサイズが閾値以下であるか判定することを特徴とする請求項 1 乃至 4 のいずれか 1 項に記載の画像処理装置。

【請求項 8】

前記除去手段で処理された後の画像に対して、文字認識処理を実行する文字認識手段を更に備えることを特徴とする請求項 1 乃至 7 のいずれか 1 項に記載の画像処理装置。

【請求項 9】

コンピュータを、請求項 1 乃至 8 のいずれか 1 項に記載の画像処理装置の各手段として機能させるためのプログラム。

【請求項 10】

画像内に含まれる画素塊について、当該画素塊のサイズが閾値以下であるか判定する判定ステップと、

前記判定ステップでサイズが閾値以下であると判定された画素塊を、前記画像から除去する除去ステップと、

を備え、

前記閾値は、前記画素塊の位置に応じて動的に変化させることを特徴とする画像処理方法。

【発明の詳細な説明】**【技術分野】****【0001】**

本発明は画像からノイズを除去する画像処理に関する。

【背景技術】

【0002】

スキャナやカメラから入力された画像を処理する画像装置において、文字以外のノイズを除去する方法が提案されている。特に、文字認識処理を精度よく行うには、できる限り文字部分に該当する画素のみを残し、文字以外のノイズを除いた二値画像を生成した後に文字認識処理を実行することが好ましい。

【0003】

特許文献1では、画素矩形の種類と、周辺に存在する画素矩形の種類の位置関係の組み合わせから、注目する画素矩形が、孤立点ノイズの矩形か点線を構成する矩形かの判定を行っている。

10

【先行技術文献】

【特許文献】

【0004】

【特許文献1】特開2003-259129号公報

【発明の概要】

【発明が解決しようとする課題】

【0005】

特許文献1では、孤立点ノイズと点線との識別を行うためのものであり、ノイズのサイズが大きいと除去できない。

20

【課題を解決するための手段】

【0006】

上記課題を解決するために、本発明の画像処理装置は、画像内に含まれる画素塊について、当該画素塊のサイズが閾値以下であるか判定する判定手段と、前記判定手段によりサイズが閾値以下であると判定された画素塊を、前記画像から除去する除去手段と、を備え、前記閾値は、前記画素塊の位置に応じて動的に変化させることを特徴とする。

【発明の効果】

【0007】

本発明によれば、周辺部にノイズの多い画像に好適なノイズ除去処理を行うことができる。

30

【図面の簡単な説明】

【0008】

【図1】実施形態1における画像処理装置の概略を示すブロック図

【図2】実施形態1における画像処理装置にて実行されるフローチャート

【図3】二値画像の一例

【図4】ステップS203の文字画素塊判定処理の詳細フローチャート

【図5】実施形態1における文字画素塊判定処理の出力結果の一例

【図6】実施形態2における画像処理装置にて実行されるフローチャート

【図7】白画素膨張処理への入力画像の一例

【図8】白画素膨張処理結果の一例

40

【図9】実施形態2における白画素膨張処理結果の一例

【図10】実施形態2における文字画素塊判定処理の出力結果の一例

【図11】実施形態3における画像処理装置にて実行されるフローチャート

【発明を実施するための形態】

【0009】

(実施形態1)

図1は、実施形態1に係る画像処理装置の概略を示すブロック図である。CPU101は、ROM102に格納されている制御プログラムを実行することにより、本装置全体の制御を行う。ROM102は、CPU101が実行するコンピュータプログラムや各種パラメータデータを格納する。コンピュータプログラムは、CPU101で実行されること

50

により、後述するフローチャートに示す各処理を実行するための各種手段（各処理部）として、当該装置（コンピュータ）を機能させる。なお、本実施形態では、ソフトウェア（プログラム）をコンピュータ（CPU）で実行することにより、後述するフローチャートの各ステップに対応する処理を実現するが、その処理の一部または全部を電子回路などのハードウェアで実現するようにしても良い。また、本発明の画像処理装置は、汎用パソコンを用いて実現しても良いし、画像処理専用装置として実現するようにしても良い。RAM 103は、画像や各種情報を記憶する。また、RAM 103は、CPUのワークエリアやデータの一時待避領域として機能する。

【0010】

外部記憶装置104は、辞書などの各種データを記憶する。外部記憶装置104は、例えば、ハードディスクやCD-ROM等を含む。なお、本発明の装置をコンピュータに実現させるためのコンピュータプログラムは、このコンピュータ読取可能な外部記憶媒体に格納されていても良いし、ネットワークを介して供給されるようにしても良い。ディスプレイ105は、例えば、LCDやCRTである。

【0011】

入力装置106は、例えば、デジタルカメラやスキャナ等の画像入力装置を接続するためのインターフェースであっても良いし、画像入力装置そのものであっても良い。また、スキャナやカメラの機能の1つとして実現するために、スキャナやカメラの内部に本発明の装置を組み込んで実現しても良い。また、画像処理装置は、スマートフォン等のモバイル端末であってもよく、その場合、入力装置は、モバイル端末に内蔵されたカメラユニットを用いればよい。

【0012】

バス107は、これらの各ユニット101～106を接続するためのバスである。また、画像処理装置は、不図示のネットワークインターフェースを介して、ネットワーク上に接続されている外部装置（例えば、サーバや画像入力装置等）と通信し、プログラムやデータを送受信してもよい。

【0013】

図2は、実施形態1に係る画像処理装置にて実行される処理を示すフローチャートである。

【0014】

ステップS201において、CPU101は、入力装置106から入力されたカラー画像に対して二値化処理を実行することにより二値画像へ変換する。

【0015】

二値化処理は、例えば、以下の手順で行えばよい。まず、カラー画像の各画素の輝度を $Y = 0.299R + 0.587G + 0.114B$ によって算出する。なお、輝度の算出式は上記式に限るものではなく、他の形式/変換式を用いて、輝度画像に変換してもよい。そして、この輝度画像から輝度ヒストグラムを作成し、この輝度ヒストグラムより文字（前景）の輝度と、背景の輝度との2つのピークを検出して、それらのピークの間の谷部分でピークを最も良く分離する閾値を求める。そして、輝度画像の各画素に対し、その輝度値が閾値未満の画素を文字に該当する黒画素として「1」、閾値以上の画素を背景に該当する白画素として「0」を割りあてることにより、二値画像を生成する。なお、二値化処理はこの手法に限るものではなく、公知の手法を適用することが可能である。

【0016】

図3は、1つの文字画像「0」を撮影し、ステップS201にて生成された二値画像の一例である。図3には、二値化処理時に黒画素と判断されたノイズが含まれている。よって、文字認識処理を精度よく行うために、後述の処理ステップを実行することで、文字以外の黒画素を除去し、できる限り文字部分に該当する画素のみを残した画像を生成する。

【0017】

なお、文字の二値画像を取得する方法は以下のようにして行われたものとする。例えば、スマートフォンのカメラ機能で、画面内に文字列撮影用のガイド（例えば、[] 状の

10

20

30

40

50

マーク)を表示し、そのガイド内に文字が収まるように撮影することで、文字画像を取得することができ、その文字画像を二値化すると図3のようになる。なお、文字画像を取得する方法は上記に限るものではない。例えば、帳票を撮影し、当該帳票画像から文字記入枠を検出し、当該検出した文字記入枠内の画像を切り出すことによって文字画像を取得することができる。また、入力画像上に、固定ピッチの文字が複数ある場合は、画像を固定ピッチの幅で等分し、そこから相対的に1文字の存在領域を算出してもよい。

【0018】

ステップS202において、CPU101は、ステップS201で生成された二値画像に対し、画素の連結性をもとに連続した画素に同じラベルを付与するラベリング処理を実行し、付与されたラベル値が同一である画素を1つの画素塊とみなす。

10

【0019】

ラベリング処理は、例えば以下のような手法が知られている。二値画像を形成する各ラインの画素をスキャンし、黒画素が連続する区間について、開始位置と終了位置をテーブルに格納する。全ラインについてこの処理を終えると、ライン間で少なくとも一部が接している区間に同じラベルを与え、同一のラベルを有する黒画素が連続する領域を形成することによりラベリング画像を生成することができる。ここで、ラベリング画像内で同一のラベルを持つ画素塊を各々独立した画素塊とみなす。

【0020】

図3の例では、ラベリング処理の結果、画素塊301、画素塊302、画素塊303、画素塊304に別々のラベルが付与され、各画素塊が連結領域として検出されたものとして説明を続ける。

20

【0021】

ステップS203において、CPU101は、ステップS202で生成された画素塊が、文字に該当する画素塊か否かを判定し、文字の画素塊以外を削除する。

【0022】

英数字において、文字と判定すべき画素塊は、ノイズの画素塊より画素数は大きい傾向にある。しかし、被写体のノイズや撮影状況によっては、二値化処理によって文字が細分化されている可能性がある(例えば図3の304)。また、カメラで文字が記載されている看板やプレートを撮影した画像を処理する場合、看板やプレートの縁より外側の輝度値が一樣ではない。この影響で、撮影した文字画像の縁の部分に、面積の大きなノイズの画素塊が残る可能性がある(例えば図3の301)。

30

【0023】

したがって、本実施形態では、文字が存在すると推定される領域の中心にある画素塊ほど、文字に該当する画素塊と判定されやすくするための判定基準を加える。例えば、黒画素塊のサイズが閾値以上であれば文字に該当すると判定する場合、中心に近い位置にある画素塊に対してはその閾値を小さくし、周辺にある画素塊に対してはその閾値を大きくする。このように、閾値を画素塊の位置に応じて動的に変動させることで、周辺にある大きなサイズのノイズの画素塊は削除され、且つ、中心部に近い位置にある文字の一部を構成する画素塊は残ることになる。この処理の詳細は図4を用いて後述する。

【0024】

ステップS204において、CPU101は、ステップS203で選択された文字画素塊で構成される二値画像を公知の文字認識処理に入力し、文字認識結果の文字コードを出力する。

40

【0025】

図4は、ステップS203の文字画素塊判定処理の詳細を示すフローチャートである。

【0026】

ステップS401において、CPU101は、画像中の文字が存在すると推定される文字領域の中心座標(X_img , Y_img)を算出する。本実施例では、図3のように、画像上に存在する文字は画像の中心に1文字が存在すると予めわかっているものとし、画像の中心座標(中心位置)を算出する。

50

【 0 0 2 7 】

ステップ S 4 0 2 において、CPU 1 0 1 は、文字が存在すると推定される矩形領域上にある画素塊の 1 つを処理対象として選択する。以下、処理対象となる画素塊を文字塊 n とし説明を続ける。

【 0 0 2 8 】

ステップ S 4 0 3 において、CPU 1 0 1 は、ステップ S 4 0 2 で選択された画素塊 n の中心座標 (X _ n , Y _ n) を算出する。画素塊の中心座標は、当該画素塊に外接する外接矩形の中心位置とする。

【 0 0 2 9 】

ステップ S 4 0 4 において、CPU 1 0 1 は、ステップ S 4 0 1 で算出した文字領域の中心座標 (X i m g , Y i m g) と、ステップ S 4 0 3 で算出した画素塊 n の中心座標 (X n , Y n) とに基づいて、両座標間の距離 d i s _ n を算出する。距離計算手法は直線距離でもよいし、計算量削減の観点から単純なシティブロック距離でも構わない。

10

【 0 0 3 0 】

ステップ S 4 0 5 において、CPU 1 0 1 は、ステップ S 4 0 4 で算出した距離に基づいて、処理対象の画素塊 n が文字の画素塊かを判定するための閾値 T h _ n を設定する。本実施例では、既定の値 α に対し、距離 d i s _ n を乗算した値を閾値 T h とする。

【 0 0 3 1 】

ステップ S 4 0 6 において、CPU 1 0 1 は、画素塊 n の画素数 S n (すなわち画素塊 n のサイズ) を、ステップ S 4 0 5 で算出した閾値 T h _ n と比較する。

20

【 0 0 3 2 】

画素数 S n が閾値 T h _ n より大きければ (Y e s であれば) 、 S 4 0 7 に進み、当該画素塊 n は文字画素塊であるとして残すものとする。

【 0 0 3 3 】

一方、画素数 S n は閾値 T h _ n 以下であれば (N o であれば) 、 S 4 0 8 に進み、当該画素塊 n はノイズであるとして除去 (すなわち、白画素に置換) する。

【 0 0 3 4 】

このように、画素数の少ない画素塊 (小さいサイズの黒画素塊) でも、中心に近ければ文字を構成している可能性が高いと判断して残し、画素数の多い画素塊 (大きなサイズの黒画素塊) でも、中心から距離が離れていればノイズの可能性が高いと判定して削除できる。

30

【 0 0 3 5 】

ステップ S 4 0 9 において、CPU 1 0 1 は、すべての画素塊を処理対象として判定したか判断し、すべての画素塊を判定し終われば、処理を終了する。一方、まだ処理対象にしていない画素塊が残っていると判断した場合 (N o であれば) 、ステップ S 4 0 2 に戻る。

【 0 0 3 6 】

図 5 は、図 3 の二値画像に対して、ステップ S 2 0 3 の処理を実行した結果の例である。図 3 の画素塊 3 0 4 は、画素数の少ない画素塊だが、中心からの距離が近いので、文字画素塊と判定されて残っている。一方、画素塊 3 0 1 , 3 0 3 は画素数の多い画素塊だが、中心からの距離が遠いため、ノイズと判定されて除去されている。

40

【 0 0 3 7 】

以上説明したように、本実施形態 1 によれば、文字画像の中心からの距離に応じて判定用の閾値を動的に変化させることで、周辺部にノイズの多い画像に好適なノイズ除去処理を行うことができる。

【 0 0 3 8 】

(実施形態 2)

本実施形態では、文字にノイズが付着した状態の画素塊から、ノイズに該当する画素を切断する例を説明する。この例では、看板やプレートなどで発生する直線状の錆や背景ムラの影響により、二値化処理後の画像において、文字部分の黒画素にノイズの黒画素が付

50

着している場合を想定している。本実施形態では、特に、横線状のノイズが付着している場合に好適な処理である。

【 0 0 3 9 】

図 6 は、実施形態 2 に係る画像処理装置にて実行される処理を示すフローチャートである。図 2 と同一の符号が付与されたステップは、実施形態 1 と同様の処理であるため、説明を省略する。

【 0 0 4 0 】

ステップ S 2 0 2 でラベリング処理を行った後、ステップ S 6 0 1 において、CPU 1 0 1 は、二値化処理で背景部分として割り当てられた白画素「0」の膨張処理（言い換えると黒画素の収縮処理）を行う。具体的には、注目画素を中心としたマスク内に 1 画素でも白画素があれば、当該注目画素を白に置き換えるマスク処理を実行する。

10

【 0 0 4 1 】

なお、本実施形態において、図 3 の 3 0 2 のように、横線状のノイズが付着している場合を想定している。そこで、本実施形態では、縦方向の辺が長い長方形のマスクを適用し、白画素を縦方向にのみ膨張させることで、横方向ノイズを文字から分離しやすくする。

【 0 0 4 2 】

図 7 は白画素膨張処理を実行する前の画像を拡大表示した例であり、図 8 は、図 7 の画像に対して、1 × 2 画素サイズの縦方向に長い長方形マスクを用いて白画素膨張処理（マスク処理）した結果を示す例である。この場合、縦方向に連結した黒画素は残り、横方向にのみ連結している黒画素が除去できる。

20

【 0 0 4 3 】

上述したステップ S 6 0 1 の処理が実行された後、ステップ S 2 0 3 に進み、実施形態 1 で説明した文字画素塊判定処理が実行される。

【 0 0 4 4 】

図 9 は、図 3 の二値画像に対して、ステップ S 6 0 1 の白画素膨張処理を適用した結果を示すものである。文字に横線状のノイズが付着した図 3 の 3 0 2 は、ノイズの細い部分で切断されるので、9 0 1、9 0 2 の 2 つの画素塊に切断できている。さらに、図 9 の画像に対してステップ S 2 0 3 の文字画素塊判定処理を実行すると、図 1 0 のような画像になる。9 0 1 の画素塊は画素数が少なく、中心からの距離も遠いためノイズと判断され、図 1 0 では、除去されている。

30

【 0 0 4 5 】

以上説明したように、本実施形態によれば、横線状のノイズが付着した文字画像に対して、縦長の長方形のマスクを使用して白画素膨張処理を行うことで、横線状のノイズが切断されやすくなる。そして切断されたノイズは、文字画素塊判定処理で除去されるので、さらに文字認識処理に好適なノイズ除去を行うことができる。

【 0 0 4 6 】

（実施形態 3）

実施形態 2 では、背景に該当する白画素の膨張処理を行うことで、ノイズと文字の画素塊を分断している。しかし、文字に該当する黒画素も一部収縮し、画素塊を構成する画素数が減少する場合がある。そこで、本実施形態 3 では、一度白画素の膨張処理を行った後に、白画素の収縮処理のステップを追加する。

40

図 1 1 は、実施形態 3 に係る画像処理装置にて実行される処理を示すフローチャートである。図 2、6 と同一の符号が付与されたステップは、実施形態 1、2 と同様の処理であるため、説明を省略する。

【 0 0 4 7 】

ステップ S 6 0 1 で白画素の膨張処理（言い換えると黒画素の収縮処理）を実行した後、ステップ S 1 1 0 1 において、白画素の収縮処理（言い換えると黒画素の膨張処理）を行う。具体的には、注目画素を中心としたマスク内に 1 画素でも黒画素があれば、当該注目画素を黒に置き換えるマスク処理を行う。マスクのサイズは S 6 0 1 と同じサイズのマスクを用いればよい。

50

【 0 0 4 8 】

以上説明したように、本実施形態 3 によれば、ノイズ切断のために収縮してしまった文字画素を膨張させることで、文字認識処理に好適な画像を得ることができる

(実施形態 4)

実施例 3 , 4 では、文字に横線状のノイズが付着している場合のノイズ切断処理について説明した。なお、被写体に縦方向のノイズが多い場合 (例えば、看板の上部から雨が滴った場合などの雨だれによるノイズなど) は、ステップ S 6 0 1、S 1 1 0 1 におい横長の長方形のマスクを用いて処理すればよい。

【 0 0 4 9 】

これにより、縦方向のノイズも切断されやすくなり、文字画素塊判定処理をより好適に適用できる。

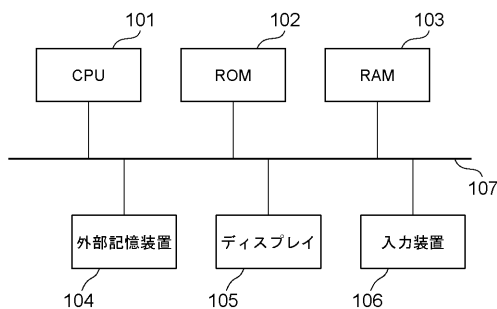
10

【 0 0 5 0 】

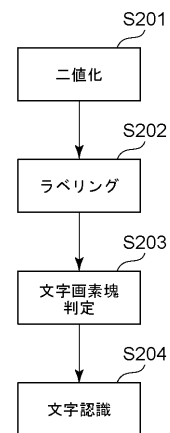
(その他の実施形態)

本発明は、上述の実施形態の 1 以上の機能を実現するプログラムを、ネットワーク又は記憶媒体を介してシステム又は装置に供給し、そのシステム又は装置のコンピュータにおける 1 つ以上のプロセッサがプログラムを読み出し実行する処理でも実現可能である。また、1 以上の機能を実現する回路 (例えば、A S I C) によっても実現可能である。

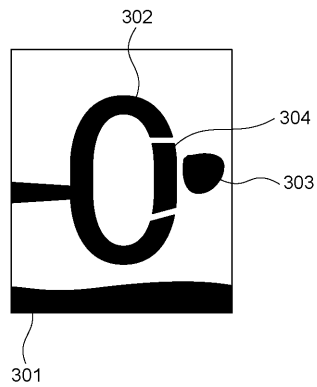
【 図 1 】



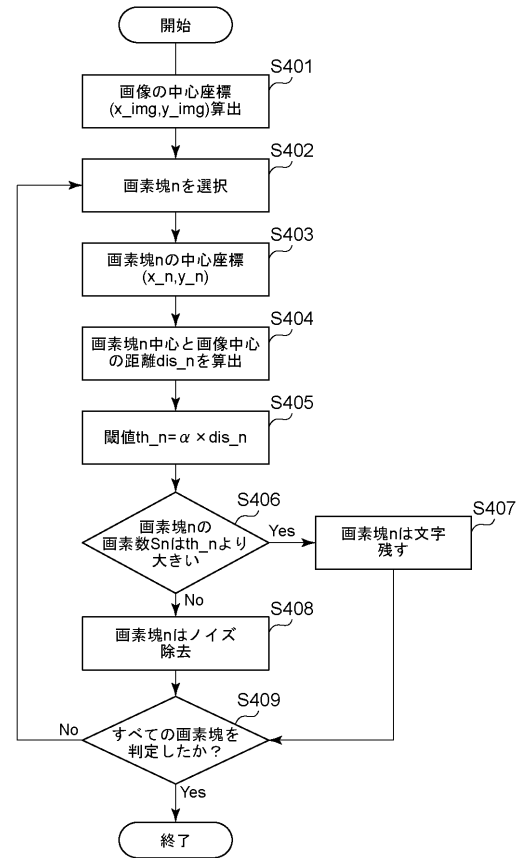
【 図 2 】



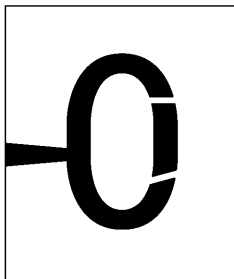
【図 3】



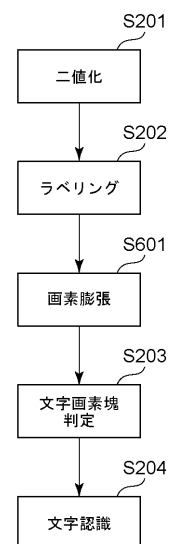
【図 4】



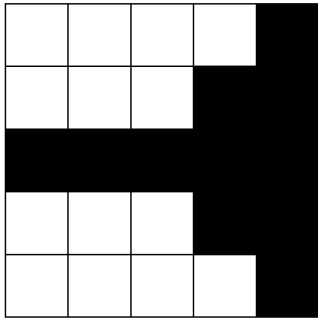
【図 5】



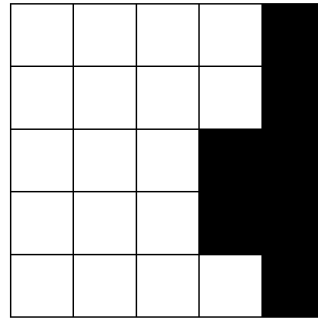
【図 6】



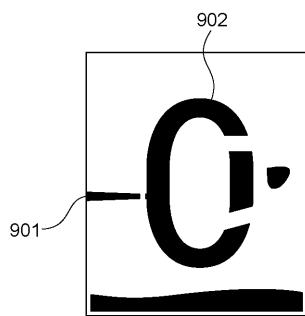
【 図 7 】



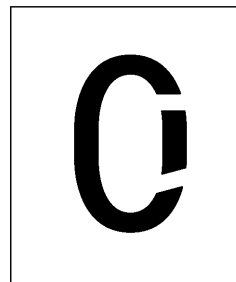
【 図 8 】



【 図 9 】



【 図 1 0 】



【図 11】

