(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3663659号
(P3663659)
(45)発行日　平成17年6月22日(2005.6.22)　(24)登録日　平成17年4月8日(2005.4.8)

(51)Int.Cl.[7]　F I
G06K 9/62　G06K 9/62 610D
G06K 9/20　G06K 9/20 340J

請求項の数 2　(全 23 頁)

(21)出願番号　特願平7-76692
(22)出願日　平成7年3月31日(1995.3.31)
(65)公開番号　特開平8-272905
(43)公開日　平成8年10月18日(1996.10.18)
審査請求日　平成13年11月19日(2001.11.19)

(73)特許権者　000002369
セイコーエプソン株式会社
東京都新宿区西新宿２丁目４番１号
(74)代理人　100095728
弁理士　上柳　雅誉
(72)発明者　村山　靖彦
長野県諏訪市大和３丁目３番５号　セイコーエプソン株式会社内

審査官　新井　則和

(56)参考文献　特開平０４－３４６１８８（ＪＰ，Ａ）
特開平０５－１６６００１（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.[7]，ＤＢ名)
G06K 9/00～9/76

(54)【発明の名称】画像処理方法および画像処理装置

(57)【特許請求の範囲】

【請求項１】

画像入力手段により入力された文書画像データから処理領域抽出工程により処理対象とする領域を抽出し、抽出された領域内における文書画像データから、その領域内の文字の大きさに対応したしきい値をしきい値算出工程により算出し、このしきい値を用いて、領域判定工程により、前記処理領域抽出工程により抽出された領域内において、前記しきい値算出工程により算出したしきい値以上の長さを有する連続した黒画素の総数と連続した黒画素の総数との比である直線密度を求め、この直線密度の値を基にして、当該領域内の言語の種類の判定を行うことを特徴とする画像処理方法。

【請求項２】

画像入力手段により入力された文書画像データから処理対象とする領域を抽出する処理領域抽出手段と、

この処理領域抽出手段により抽出された領域内における文書画像データから、その領域内の文字の大きさに対応したしきい値を算出するしきい値算出手段と、　このしきい値算出手段により算出されたしきい値を用いて、前記処理領域抽出手段により抽出された領域内において、前記しきい値算出手段により算出したしきい値以上の長さを有する連続した黒画素の総数と連続した黒画素の総数との比である直線密度を求め、この直線密度の値を基にして、当該領域内の言語の種類の判定を行う領域判定手段と、

を有することを特徴とする画像処理装置。

【発明の詳細な説明】

【０００１】
【産業上の利用分野】
本発明は、たとえば文字認識（ＯＣＲ）処理における文字の切り出しおよび認識処理の前処理として、処理対象とする領域を抽出し、抽出した領域の言語の種類（たとえば、日本語あるいは英文のようなアルファニューメリック）を判定する画像処理方法および画像処理装置に関する。
【０００２】
【従来の技術】
文字認識処理に先立って文字の切り出し処理を行う際、日本語あるいは中国語等文字のように外形が正方形に近い形の文字（以下では「日本語」と表記する）と、英文等のアルファニューメリックのように文字の外形が縦長の長方形となる文字（以下では「αＮ」と表記する）では、その文字の構成形態の違いにより、その形態に適した文字切り出しを行うのが一般的である。この方法は、日本語に対しては、日本語用の文字切り出し方法を用い、英文などに対しては、αＮ用の文字切り出し方法を用いて文字の切り出しを行うものである。また、切り出した後の文字認識処理においては、文字認識の精度を上げるため、処理対象とする領域に適した日本語用およびαＮ用の文字認識方法を用意し、文字認識を行う。このように、日本語の領域あるいはαＮの領域に対応した切り出しおよび認識処理を行うためには、処理に先立って、ユーザが処理対象とする領域の言語の種類を判断して、処理領域ごとに日本語であるか、αＮであるか等の指定を行う必要がある。
【０００３】
これに対して、日本語、αＮともに対応できる文字切り出し方法、文字認識方法を用いて、処理に先立つ領域の指定を必要としない文字切り出し、文字認識方法も存在する。
【０００４】
【発明が解決しようとする課題】
ところで、前者の方法、つまり、処理に先立って、ユーザが処理対象とする領域が日本語であるかαＮであるか等の指定を行い、領域に適した文字切り出し、あるいは文字認識を行う方法では、たとえば、日本語の場合、図２２（ａ）に示した「日本語文字認識」という文書における「語」を例に取ると、この「語」は「言」の部分ｅ１と「吾」の部分ｅ２とに分離されるが、このように分離される可能性のある文字であっても、それを積極的に統合して、「語」という文字として処理を行うことができる。あるいは、英文の場合にあっても、図２２（ｂ）に示した「Japanese character recognition」の単語と単語の間の空間部分ｆ１，ｆ２を活用して、単語を中心とした切り出し処理が可能となる。これは、それぞれの領域に適応した処理が行われるから可能となるのであり、これにより、文字切り出し率、文字認識率ともに好結果を得ることができる。しかし、この方法は、処理領域の言語の種類、つまり、処理領域の言語が日本語かαＮかの指定をユーザが行わなければならないという問題がある。
【０００５】
一方、後者の方法、つまり、日本語、αＮともに対応できる文字切り出し方法、文字認識方法を用いる場合には、領域の種類を指定しなければならないという手間は省けるが、前者の各領域に適した処理方法に比べると、前記したような分離される可能性のある文字を積極的に統合して１つの文字とする処理や単語を中心とした切り出しおよび認識処理が行えないため、文字切り出し率が落ちるとともに、認識率も低くなるという課題が生じる。
【０００６】
そこで、本発明はこれらの課題を解決するもので、たとえば文字認識処理の前処理として、処理対象とする領域の抽出を行うとともに、抽出された領域における言語の種類を自動的に判断することにより、文字の切り出しおよび認識処理を高精度に行うことを可能とした画像処理方法および画像処理装置を提供することを目的とする。
【０００７】
【課題を解決するための手段】
本発明の画像処理方法は、画像入力手段により入力された文書画像データから処理領域

抽出工程により処理対象とする領域を抽出し、抽出された領域内における文書画像データから、その領域内の文字の大きさに対応したしきい値をしきい値算出工程により算出し、このしきい値を用いて領域判定工程により、前記処理領域抽出工程により抽出された領域の言語の種類を判定するために必要な直線密度を抽出し、この直線密度により当該領域の言語の種類の判断を行う。

【０００８】

前記処理領域抽出工程は、処理対象とする文書画像データにおける段落を基準として処理領域の抽出を行なう。

【０００９】

また、前記処理領域抽出工程は、文書画像データをラスタスキャン方式により処理ラインに従って走査することで、文書画像データにおける画素の白黒反転回数があらかじめ設定した値に達した範囲を基準として処理領域の抽出を行なうようにしてもよい。

【００１０】

前記しきい値算出工程は、前記処理領域抽出工程により抽出された領域内に含まれる各文字の外接矩形を抽出し、各外接矩形の大きさの平均値をもとに、その領域内の文字の大きさに対応したしきい値の算出を行なう。

【００１１】

また、前記しきい値算出工程は、文書画像データをラスタスキャン方式により処理ラインに従って走査することで、前記処理領域抽出工程により抽出された領域内に含まれる文字の大きさまたは文字の太さにほぼ相当する黒画素の連続したつながりの平均的な長さを求め、求めた平均的な長さをもとに、その領域内の文字の大きさに対応したしきい値の算出を行うようにしてもよい。

【００１２】

前記領域判定工程は、前記処理領域抽出工程により抽出された領域内において、前記しきい値算出工程により算出したしきい値以上の長さを有する連続した黒画素の総数と連続した黒画素の総数との比である直線密度を求め、この直線密度の値を基にして、当該領域内の言語の種類の判断を行なう。

【００１３】

前記領域判定工程は、前記処理領域抽出工程により抽出された領域ごとに、その領域の言語の種類の判定を行った後、領域の種類の判定を行なう基準となる複数の処理領域の判定結果を用いて、誤判定とみなされる領域の判定結果の修正を行なう。

【００１４】

また、本発明の画像処理装置は、画像入力手段により入力された文書画像データから処理対象とする領域を抽出する処理領域抽出手段と、この処理領域抽出手段により抽出された領域内における文書画像データから、その領域内の文字の大きさに対応したしきい値を算出するしきい値算出手段と、このしきい値算出手段により算出されたしきい値を用いて、前記処理領域抽出手段により抽出された領域の言語の種類を判定するために必要な直線密度を抽出し、この直線密度により当該領域の言語の種類の判断を行う領域判定手段とを有している。

【００１５】

前記処理領域抽出手段は、処理対象とする文書画像データにおける段落を基準として処理領域の抽出を行なう。

【００１６】

また、前記処理領域抽出手段は、文書画像データをラスタスキャン方式により処理ラインに沿って走査し、文書画像データにおける画素の白黒反転回数があらかじめ設定した値に達した範囲を基準として処理領域の抽出を行なうようにしてもよい。

【００１７】

前記しきい値算出手段は、前記処理領域抽出手段により抽出された領域内に含まれる各文字の外接矩形を抽出し、各外接矩形の大きさの平均値をもとに、その領域内の文字の大きさに対応したしきい値の算出を行なう。

【００１８】
また、前記しきい値算出手段は、文書画像データをラスタスキャン方式により処理ラインに従って走査し、前記処理領域抽出手段により抽出された領域内に含まれる文字の大きさまたは文字の太さにほぼ相当する黒画素の連続したつながりの平均的な長さを求め、求めた平均的な長さをもとに、その領域内の文字の大きさに対応したしきい値の算出を行うようにしてもよい。
【００１９】
前記領域判定手段は、前記処理領域抽出手段により抽出された領域内において、前記しきい値算出手段により算出したしきい値以上の長さを有する連続した黒画素の総数と連続した黒画素の総数との比である直線密度を求め、この直線密度の値を基にして、当該領域内の言語の種類の判断を行なう。
【００２０】
前記領域判定手段は、前記処理領域抽出手段により抽出された領域ごとに、その領域の言語の種類の判定を行った後、領域の種類の判定を行なう基準となる複数の処理領域の判定結果を用いて、誤判定とみなされる領域の判定結果の修正を行なう。
【００２１】
【作用】
本発明は、画像入力手段により入力された文書画像データから処理領域の抽出を行い、これにより抽出された領域内における文書画像データから、その領域内の文字の大きさに対応したしきい値を算出し、このしきい値を用いて、前記抽出された領域の言語の種類を判定するために必要なパラメータ（直線密度）を抽出して、この直線密度の値をもとに、当該領域の言語の種類の判断を行なう。
【００２２】
このように、本発明では、処理領域を抽出し、抽出された処理領域単位でその領域が何の言語で書かれたものであるかの判定を行う。この判定を行う際には、当該領域内の画像データの直線成分を検出して、その検出結果を基に言語を判定する。すなわち、言語として、日本語とαＮを考えた場合、日本語は画像データに直線成分が多く、αＮは直線成分が殆どないことに着目して、直線密度を求めて、この直線密度によって、当該領域内の言語の種類の判定を行う。　これにより、入力された文書画像データから処理領域の抽出、抽出された領域の言語の種類の判定を自動的に行うことができ、この判定結果を文字の切り出しおよび文字認識処理に渡すことにより、的確な切り出しおよび認識処理が可能となる。
【００２３】
【実施例】
以下、本発明の実施例を説明する。なお、この実施例では、特に示さない限り処理対象とする画像の解像度は３００ＤＰＩ（ドット／インチ）であって、処理対象画像としては文字とし、この文字は白色の背景（白色の原稿用紙）に黒で描かれているものとする。
【００２４】
　図１は、本実施例における画像処理装置の構成を説明した図である。画像処理装置１０は、処理対象となる領域を抽出する処理領域抽出部１１、領域判定をするための直線密度を抽出する際に必要となるしきい値を算出するしきい値算出部１２、前記領域抽出部１１により抽出された領域が日本語領域であるかαＮ領域であるかを判断する領域判定部１３、さらに、これら各処理部を接続するためのバスよ１４により構成される。なお、領域判定部１３内には誤り訂正手段１３１が設けられているが、これについては後述する。
【００２５】
また、画像処理装置１０には、処理対象となる画像の入力を行うための画像入力装置１５が接続される。この画像入力装置１５は光学的な画像入力を行う場合は光学的画像入力装置（たとえばスキャナ）や通信による入力装置、あるいは画像データを蓄える記憶装置などである。なお、図１において、画像入力装置１５は画像処理装置１０に対し外部接続されているが、画像処理装置１０内部に含まれていてもよい。

【００２６】
また、前記処理領域抽出部１１、しきい値算出部１２、領域判定部１３などの各処理部は、ハードロジックにより構成されるものであっても、あるいはＣＰＵを付加することによりソフト的な処理により実現されるものであってもよい。
【００２７】
次に画像処理装置１０全体の概略的な処理の流れについて説明する。図２は画像処理装置１０の処理をソフト的な処理により実現した場合について示したものである。まず、処理対象となる画像データから領域の種類の判定単位となる処理領域の抽出を処理領域抽出部１１より行う（ステップｓ１１）。次に前記処理領域抽出部１１で抽出した領域から領域判断のための直線密度を抽出するめに必要となるしきい値（抽出された領域内の文字の大きさに対応した値であり、これについては後に詳細に説明する）をしきい値算出部１２により算出する（ステップｓ１２）。そして、しきい値算出部１１で算出したしきい値を用いて領域判断のための必要な直線密度（前記しきい値以上の長さを有する黒画素数をもとに求められた値であり、これについては後に詳細に説明する）の抽出を行い、この直線密度を用いて、処理対象となる領域が日本語であるかαＮであるかを領域判定部１３にて行う（ステップｓ１３）。そして、画像データからすべての処理対象となる領域の抽出が終了し、その領域の判定が終了したか否かを判断し（ステップｓ１４）、すべての領域の処理が終了していれば処理を終了する。
【００２８】
図２はソフト的な処理をする際の全体の処理の流れを示したが、ハードロジックにて構成することにより、処理領域抽出部１１、しきい値算出部１２、領域判定部１３はそれぞれ独立に並列処理を行うことが可能である。図３は並列処理を行った場合の処理の概要を示した図である。図３において、まず、第１の処理ステップＳ１において処理対象となる最初の領域Ａ１が処理領域抽出部１１により抽出される。領域Ａ１の抽出が終わると第２の処理ステップＳ２に移り、次の処理領域Ａ２の抽出とともに、処理領域Ａ１から直線密度を抽出するために必要となるしきい値の算出がしきい値算出部１２にて行われる。そして、領域Ａ２の抽出、領域Ａ１からのしきい値算出が終了すると、第３の処理ステップＳ３に移る。第３の処理ステップＳ３では、領域Ａ３の抽出、領域Ａ２のしきい値算出、領域Ａ１の領域判定が行われる。このような処理が順次行われる。つまり、各処理が並列的に処理されることになり、全体的な処理時間の短縮を図ることができる。
【００２９】
以上は本発明の全体的な処理の概略であるが、以下に、個々の処理について説明する。まず最初に処理領域抽出部１１の処理内容について以下に説明する。
【００３０】
処理領域抽出部１１は、前記したように、処理対象となる領域を所定の単位で抽出するものである。ところで、言語の種類などは、段落を単位として変化することが多い。例えば図４に示すように、論文のフロントページでは、要約部分（図示、Ａ１の領域）が日本語でかかれ、続いて英文の要約（図示、Ａ２の領域）が、そして日本語の本文（図示、Ａ３の領域）といった具合である。
【００３１】
そこで、処理領域抽出部１１では、領域抽出の第１の方法として、段落の抽出を行い、抽出した段落を１つの領域と考え、各段落を処理の単位とする。ここで、段落の抽出方法は、処理対処とする画像データを複数の単位ブロックに分割し、分割された複数の単位ブロック毎に縦、横方向の射影を求め、求めた射影の周期性を調べて文字のある領域を判別し、文字と判定されたブロックを統合することにより段落の抽出を行う方法（特開平３－４４７８８）や、黒画素の輪郭を追って黒領域に接する矩形領域を抽出し、各矩形領域の大きさによって文字領域を分離し、分離した文字領域を近接する矩形を統合することにより段落の抽出を行う方法（特開平３－４３８７９）等があり、このような方法を用いて抽出された段落をもとに領域の分離を行う。
【００３２】

また、段落の始めは、一般的に改行されていることから、この改行を判断することにより段落の抽出を行うことも可能である。

【００３３】

以上のように、段落を処理領域の１つの単位とすることにより、段落ごとの領域の判定が可能となる。

【００３４】

また、領域抽出の第２の方法として、文字行を基準として領域の抽出を行う方法もある。文字行ごとに判断する方がより細かな領域の種類の判断が可能となる。すなわち、前記のように段落ごとに領域の種類の判断をすると、同じ段落中に、日本語とαＮが混在する場合もあり、文字行を基準とすれば、このような場合にも対応が可能となる。そこで、以下では、行を処理単位とした処理領域抽出について説明する。

【００３５】

行の抽出方法を、図５に示すような文章を例にとって説明する。図５に示す方法は、文章の行方向に射影１０１，１０２，１０３をとり、その射影結果の谷間ｐ１，ｐ２を検出することによって、行Ｒ１，Ｒ２，Ｒ３の抽出を行い、これら各文字行を処理領域とする。なお、行の抽出方法は上記方法に限るものではなく、ラベリング処理等により抽出された複数個の文字塊ブロック間の位置関係に着目して文字行の抽出を行う方法（特開平３－６２２８４）や、画素の太線化処理および細線化処理を行うことにより文字を統合させ、行の抽出を行う方法（特開平３－４３８７９）等を用いてもよい。このような行を単位とした処理領域抽出処理により、段落を単位とした方法に比べ、より細かな対応が可能となる。

【００３６】

これまで述べた段落あるいは行を単位とした領域の抽出では、領域の抽出の際に、メモリのランダムアクセスを必要とするため、処理の高速化が困難となる。そこで、ラスタスキャンのみで、処理単位となる領域を抽出する方法について以下に説明する。

【００３７】

後に説明する領域判定部１３における領域判定（日本語であるかαＮであるかの判定）は、有効な直線の総数と、黒ラン（黒い画素の連続した直線的なつながりをここでは黒ランという）の総数をもとに領域の種類の判定を行うものである（この領域判定部１３の詳しい処理については後述する）。そこで、この処理領域抽出部１１での処理は、領域判定部１３による領域判定のために必要な情報が集まる程度の領域を処理単位として抽出する。なお、有効な情報量が得られるかは、黒ランの数により判断する。

【００３８】

図６により、処理領域抽出部１１による画像データからの処理領域抽出処理を説明する。まず、黒ランの数をカウントするカウンタをゼロにする初期化を行う（ステップｓ２１）。なお、このカウンタは図示されていないが処理領域抽出部１１内に設けられている。次に、処理対処となる走査ラインから黒ランの抽出を行い（ステップｓ２２）、抽出が行われるごとにカウンタのインクリメントを行う（ステップｓ２３）。そして、カウンタのカウント数があらかじめ定めた値Ｔ以上かどうかを判断して（ステップｓ２４）、カウンタのカウント数が値Ｔ以上になると、カウント数が値Ｔ以上となった走査ラインまでの領域を処理領域とする（ステップｓ２５）。なお、ステップｓ２４の判断において、カウント数が値Ｔに達しない場合には、次の黒ランの抽出を行う。つまり、ステップｓ２６（対象走査ラインの走査終了か否かの判断）からステップｓ２２に処理が戻って、黒ラン数のカウントを行い、前記同様の処理を行う。そして、対象となる走査ラインの走査が終了した場合（ステップｓ２６）には、文書画像全体の走査が終了したか否かを判断して（ステップｓ２７）、全ライン終了していれば、処理は終了し、全ライン終了していなければステップｓ２２に処理が戻る。なお、前記した値Ｔについては、後で説明する。

【００３９】

以上の処理において、黒ランの抽出（ステップｓ２２）からカウント数が或る値Ｔより大きいか否かの判断（ステップｓ２４）までの具体的な処理例を図１６を参照して説明する

。なお、図１６は本来、後で説明するしきい値算出部１２において、文字の大きさを基準としたしきい値算出を説明するための図であり、図中に記載された「３６」、「８」、「５」，「４」，「４０」などという数値は黒ランの長さ（画素数）を示すもので、これらはしきい値算出において必要とするものであり、これらについては後に詳細に説明する。また、図１６は、「日本」という文字の上端部からラスタスキャンした場合、第１，第２，第３の走査ラインＬ１，Ｌ２，Ｌ３の画像データを示すものである。図１６において、第１の走査ラインＬ１には、Ｂ１，Ｂ２で示す２本の黒ランが存在するので、第１の走査ラインＬ１の走査終了時点では、カウンタの黒ランカウント数は「２」となる。同様に、第２の走査ラインＬ１には、Ｂ３，Ｂ４，Ｂ５の３本の黒ランが存在するので、第２の走査ラインＬ２の走査終了時点では、カウンタの黒ランカウント数は前記第１の走査ラインのカウント数「２」との合計の「５」となる。さらに、第３の走査ラインＬ３には、Ｂ６，Ｂ７，Ｂ８の３本の黒ランが存在するので、第３の走査ラインＬ３の走査終了時点では、カウンタの黒ランカウント数は前記第２の走査ラインまでのカウント数「５」との合計して「８」となる。

【００４０】

仮に値Ｔの設定値が「４」であったとすると、第２の走査ラインの途中で、カウンタのカウント数が値Ｔに達するので、第１および第２の走査ラインを処理の単位である処理領域として決定し、次の処理領域の抽出を第３の走査ラインから始める。なお、ここでは、説明をわかりやすくするため値Ｔを「４」という数値に設定した例で説明したが、これは実用的な数値ではない。この値Ｔの実用的な数値に関しては、後で詳しく説明する。

【００４１】

以上のようにして、領域の抽出が行われる。なお、以上の説明では、ラスタスキャンによる処理領域の抽出において、黒ランの数をもとに処理領域の抽出を行う例を説明したが、黒ラン数ではなく、白画素と黒画素の反転回数（以下、白黒反転回数という）をカウントしてもよい。例えば、図１６の第１の走査ラインＬ１においては、ｇ１，ｇ２，ｇ３，ｇ４の部分で、白黒の反転があり、白黒反転回数は４となり、黒ランの数の２倍となる。よって、黒ランの数のかわりに、白黒反転回数をカウントした場合には、値Ｔを黒ランの数のときの２倍とすることで対応できる。なお、白黒反転は、図７に示すように、処理対象の走査ラインＬ内の注目画素Ｇ１とその１つ手前の画素Ｇ０のそれぞれのデータ（白を“０”、黒を“１”とする）を排他的論理和回路２１０に与えて、その出力から判断できる。つまり、排他的論理和回路２１０からの“１”出力をカウントすることにより白黒反転数をカウントできる。このようにすれば、処理の簡略化、回路規模の縮小も図れる。

【００４２】

なお、後述する領域判定部１３でも、黒ランのカウントを行うが、この領域抽出での処理により得られた結果を領域判定部１３でそのまま用いてもよく、また、黒ランの数かわりに白黒反転回数をカウントし、その値の１／２を用いて黒ランの数としてもよい。

【００４３】

次に処理領域の判断に用いられる前記した値Ｔについて説明する。図８に値Ｔ（黒ラン数）と領域判定部の判断結果の正解率の関係を示す。図８より、Ｔが２０００を境に２０００より小さくなると正解率が落ちていくのが分かる。これより、黒ランの数として２０００程度の情報量が必要であることが分かる。また、Ｔの値が少ないほど、処理領域が小さくなることになり、図３のような並列処理を行う場合、各処理のステップ間隔が狭くなり結果として、全体の処理時間の短縮にもつながる。よって、Ｔの値は余り大きく取らないで、２０００程度が好ましいといえる。なお、２０００程度の黒ランの数は、３００［ＤＰＩ］入力、１０ポイントの文字において、約２０文字強程度の文字数となり、このことより、先の例で示した行単位での処理でも、情報量として十分であることが分かる。

【００４４】

以上の処理により処理対象となる領域の抽出が行える。次に、抽出された領域が日本語であるかαＮであるかの判定を、或るしきい値を用いて判定する。処理の順序としては、領域判定を行うためのしきい値算出が先であるが、ここでは、まず、処理対象となる領域が

日本語であるかαＮであるかを判断する領域判定部１３の処理について説明する。

【００４５】

まず、領域判定部１３がどのような方法により領域の判定を行なうかについて、概念的な理解をするために図９を用いて、その判断方法について説明する。図９（ａ）は文字列「・・・日本語・・・」のうち、「日」の文字を拡大したもので、図９（ｂ）は文字列「・・・　Ｊａｐａｎｅｓｅ　・・・」のうち、「ａ」の文字を拡大したものである。一般的に用いられる文字の大きさは１０ポイント程度であり、日本語文字の場合は約３㎜角となる。よって、３００［ＤＰＩ］で画像の入力がされた場合、図９（ａ）で示す「日」の文字は、３６画素×３６画素程度の大きさとなる。同様に、同図（ｂ）で示す「ａ」の文字は、２０画素×２０画素程度の大きさとなる。図９からわかるように日本語の場合、多くの文字（特に漢字）は、ほとんど直線により構成される。一方、αＮの場合はほとんど曲線により構成される。よって、直線成分の存在の度合によって日本語領域かαＮ領域かの判定が可能である。

【００４６】

例えば図９（ａ）の場合には、「日」という文字には、横方向にａ１～ａ３の３つの直線が、縦方向にｂ１およびｂ２の２つの直線が存在する。一方、同図（ｂ）に示す「ａ」の文字には直線が存在しない。直線成分の存在の度合による領域の種類の判断において、単純に直線の本数のみでは、領域中に含まれる文字数によって変化するため適切な判断ができない。そこで、直線成分の存在する割合、すなわち、「直線密度」を計算するするこにより、直線密度が高い場合には日本語、低い場合にはαＮと判断する。なお、直線密度をどのように定義するかが問題となる。

【００４７】

ここで、直線密度を単位面積あたりの直線の本数とすると、文字が大きい場合には直線密度が小さくなり、文字が小さい場合には直線密度が大きくなる。すなわち、文字の大きさに直線密度が依存することになり好ましくない。そこで、文字の大きさに依存しないように、有効な直線の本数を黒ランの本数で除した値、すなわち、

直線密度＝有効な直線の総数　／　黒ランの総数　　・・・　〔１〕

とすることで、文字の大きさに依存しない値を得ることができる。ここでいう有効な直線とは、しきい値よりも大きな長さを有する直線成分のことをいう。

【００４８】

例えば、図９（ｃ）は、図９（ａ）に示す「日」の文字の大きさが縦横１／２の大きさ（１８画素×１８画素）である場合であり、このように文字の大きさが異なった場合の直線密度を求めてみる。図１０（ａ）は図９（ａ）で示すサイズの「日」における第１～第４の走査ライン部分を拡大して示すもので、図１０（ｂ）は図９（ｂ）で示すサイズの「日」における第１～第２の走査ライン部分を拡大して示すものである。ここでは、横方向の走査によって、有効な直線の総数および黒ランの総数を求めるものとし、また、図１０（ａ）における有効な直線をＸ１に示す長さ以上とすると、第１の走査ラインＬ１において、画素ｃ１～ｃ２の範囲の直線は、黒ランであり、かつ有効な直線となる。よって、この第１の走査ラインＬ１においては、有効な直線数が「１」、黒ラン数が「１」存在することになる。第２の走査ラインＬ２においても同様であり、有効な直線数が「１」、黒ラン数が「１」存在する。第３の走査ラインＬ３においては、画素ｃ３～ｃ４および画素ｃ５～ｃ６に示す直線は、ともにＸ１より短いことから有効な直線ではない。よって、第３の走査ラインＬ３においては、有効な直線数が「０」、黒ラン数が「２」存在する。また、第４の走査ラインＬ４においても同様である。

【００４９】

よって、図１０（ａ）における第１～第４の走査ラインにおける有効な直線の総数は「２」であり、黒ランの総数は「６」となることから、直線密度は0.33（＝２／６）となる。

【００５０】

同様に図１０（ｂ）における有効な直線をＸ２とすると、第１の走査ラインＬ２において、画素ｃ１０～ｃ２０の範囲の直線は、黒ランであり、かつ有効な直線となる。よって、

この第１の走査ラインＬ１においては、有効な直線数が「１」、黒ラン数が「１」存在する。また、第２の走査ラインＬ２においては、画素ｃ３０とｃ４０および画素ｃ５０とｃ６０に示す直線は、ともにＸ２より短いことから有効な直線ではない。よって、第３の走査ラインＬ３においては、有効な直線数が「０」、黒ラン数が「２」存在する。このように、図１０（ｂ）においては、第１および第２の走査ラインＬ１，Ｌ２に存在する有効な直線の総数は「１」、黒ランの総数は「３」となり、直線密度は0.33（＝１／３）となる。

【００５１】

上記例では「日」の一部において直線密度の計算例を示したが、「日」という文字全体を走査した場合においても、図９（ａ）に示す「日」の有効な直線の総数と図９（ｃ）に示す「日」の有効な直線の総数の比はほぼ２：１となり、同様に、黒ランの総数の比もほぼ２：１となる。よって、直線密度を計算した場合、その値はほぼ等しく、上記計算式〔１〕により文字の大きさに依存しない直線密度を得ることができる。

【００５２】

なお、図１０を用いて説明した前記直線密度の計算方法においては、横方向の走査のみにより値を求めたが、縦横両方向の走査結果を用いて直線密度の算出をしてもよく、また、縦方向の走査結果を用いて直線密度の算出をしてもよい。

【００５３】

ここで、有効な直線を求める際の基準となる長さ、すなわち、図１０におけるＸ１やＸ２をどのように設定するかが問題となるが、これに関しては、しきい値算出部１２の説明において詳しく説明する。

【００５４】

図１１は、前記処理領域抽出部１１により抽出された処理領域に対する領域判定部１３の処理を説明するフローチャートである。なお、この図１１のフローチャートは前記処理領域抽出部１１により抽出された或る１つの領域（たとえば、図４におけるＡ１の領域）における処理を示すものである。

【００５５】

図１１において、始めに、有効直線をカウントするカウンタ、黒ラン総数をカウントするカウンタの初期化を行う（ステップｓ３１）。なお、これらのカウンタは図示されていないが領域判定部１３内に設けられている。次に、走査対象となる走査ラインから黒ランの抽出を行い（ステップｓ３２）、黒ランの抽出が行われると黒ラン総数カウンタをインクリメントする（ステップｓ３３）。そして、抽出した黒ランの長さが、しきい値算出部１２で算出したしきい値以上か否かを判断し（ステップｓ３４）、しきい値算出部１２で算出したしきい値以上の場合には、有効直線カウンタのインクリメントを行う（ステップｓ３５）。次に、対象ラインの走査がすべて終了したか否かを判断して（ステップｓ３６）、対象となる処理ラインの走査がすべて終わるまで、黒ランの抽出を繰り返し、対象となる処理ラインの走査が終了したならば、次の走査ラインの走査に移る。そして、処理対象となる領域の全ラインの走査が終了したか否かを判断して（ステップｓ３７）、全ラインの走査が終了したら、式〔１〕に従い直線密度の算出を行う（ステップｓ３８）。この式〔１〕により求められた直線密度があらかじめ設定された値α以上であるか否かを判断し（ステップｓ３９）、直線密度がα以上の場合には、領域抽出部１１で抽出された領域は、「日本語」であると判定し（ステップｓ４０）、直線密度がα未満の時は、その領域は「αＮ」と判定する（ステップｓ４１）。なお、前記した値αをどのような値にすればよいかについては、後ほど述べる。

【００５６】

次にしきい値算出部１２の処理内容について以下に説明する。

【００５７】

しきい値算出部１２は、領域抽出部１１が抽出した領域から、有効な直線を決定するためのしきい値の算出を行う。まずは、しきい値を算出する方法の１つとして、外接矩形を求め、求めた外接矩形からしきい値を算出する方法について説明する。ここで外接矩形とは

、文字に外接する矩形のことをいい、図１２に示されるような矩形Ｋ１，Ｋ２，・・・である。なお、図において、外接矩形Ｋ１，Ｋ２，・・・は、文字に接触した状態で描かれていないが、これは、図面が繁雑とならないようにしたためである。

【００５８】

文字の切り出し方法としては、例えば、図１３に示すように文字行方向に対して、直交する方向に、黒画素の累積値である射影２０１，２０２，２０３・・・を求め、求めた射影の谷間ｐ１１，ｐ１２・・・を用いて、文字の存在する位置を確認するとともに、画素の連結状態から文字に外接する矩形の抽出を行う。なお、文字の切り出しは、上記に説明した射影を用いて求める方法に限らず、画素の境界を追跡し、追跡した境界から外接矩形を求める方法（特開３－３７７７８）など、どのような方法を用いてもよい。このようにして外接矩形が求められると、しきい値は、図１２のように求めた外接矩形Ｋ１，Ｋ２，・・・の横の長さｄ１，ｄ２，・・・，ｄｎの平均的な長さを求め、求めた平均的な長さに、ある係数β１を掛けることにより算出される。すなわち、しきい値は、

【００５９】

【数１】

$$\text{しきい値} = \beta 1 \cdot \sum_{i=1}^{n} di \Big/ n \qquad \cdots \quad 〔2〕$$

【００６０】

により算出される。ここで、β１の値は0.6～1.0の範囲の値が適当であり、実験より、βを0.7とすることにより、直線密度を用いた領域判断において良い結果が得られることが確認された。

【００６１】

なお、前記した例では、外接矩形の横の長さの平均値を用いてしきい値の算出を行っていたが、縦の長さの平均値を用いてしきい値の算出を行ってもよい。また、縦横両方の長さの平均を用いていもよい。ここで、横の長さの平均値、縦の長さの平均値、縦横の長さの平均値のいずれを用いてしきい値の算出を行うかは、領域判定部１３の有効な直線の数および黒ランの数をカウントする際に走査する方向に合せるとよい。

【００６２】

前記した方法によるしきい値算出部１２の処理の流れを図１４に示す。まず、処理対象となる領域から外接矩形の抽出を行い（ステップｓ４１）、次に、抽出した外接矩形から前記〔2〕式を用いてしきい値の算出を行う（ステップｓ４２）。このようにして、外接矩形を用いてしきい値の算出を行うことができる。

【００６３】

前記外接矩形を用いてしきい値の算出を行う方法では、外接矩形を求める際に、画像データに対してランダムアクセスが必要となることから、処理速度にやや難点が残るとともに処理が複雑になる。そこで、以下では、ラスタスキャン、すなわち、処理ラインを走査することによって、しきい値算出部１２によって、しきい値を算出する方法について説明する。

【００６４】

ラスタスキャンにより処理ラインに従って走査することで、しきい値の算出処理を行う方法を図１５および図１６を用いて説明する。図１５のフローチャートは、前記処理領域抽出部１１により抽出された或る１つの領域における処理を示すものである。

【００６５】

まず、文字の大きさγを初期化するとともに、文字の大きさγ以上の黒ランの個数をカウントするカウンタおよび、γ以上の黒ランの長さの累積値を記憶するためのアキュムレー

タの初期化を行う（ステップｓ５１）。なお、これらカウンタおよびアキュームレータは図示されていないが、しきい値算出部１２内に設けられている。

【００６６】

前記文字の大きさγの初期値は処理ラインをスキャンするごとに更新される。その初期値は、解像度が３００［ＤＰＩ］の場合、８ポイント（３㎜角弱の大きさ）の文字の約７割の大きさ程度にするのがよく、ここでは「２４（画素）」とする。２４画素としたのは、８ポイント以下の文字は通常あまり用いられることがないためである。この文字の大きさγは前記したように、処理ラインをスキャンするごとに更新される。つまり初期値としては８ポイントの大きさを基準としているが、処理領域中で実際に使われている文字の大きさは８ポイントとは限らない。そこで実際に使われている文字にあったしきい値の算出ができるように、走査ライン毎に文字の大きさγの値を更新しているのである。

【００６７】

また、文字の大きさγ以上の黒ランの個数をカウントするカウンタ、γ以上の黒ランの長さの累積値を記憶するためのアキュムレータの初期値は、ゼロでもよいが、最初の走査ラインにおいては、多くの黒ランが得られない可能性があるため、処理ラインごとに文字の大きさγの変更を行うとすると、最初の数ラインにおいて文字の大きさγの変動が激しくなる。図１６の例で言うと、第１の走査ラインＬ１において、文字の大きさγ以上の黒ランの個数は１個（図１６において、Ｂ１で示す黒ラン）であり、わずかこの１つの情報により文字の大きさγの更新が行われるため、γの精度が悪くなる。そこで、文字の大きさγ以上の黒ランの個数をカウントするカウンタには適当な個数、例えば２０程度の値を設定し、アキュムレータには、文字の大きさγの初期値と前記カウンタの値を掛けた値を設定しておくことにより上記問題を解決できる。なお、カウンタ、アキュムレータの初期値をゼロとし、そのかわり、あるライン数の走査が終了以降に、文字の大きさγの更新を行うようにしても、上記問題は解決できる。

【００６８】

次に対象とする走査ラインにおいて黒ランの抽出を行う（ステップｓ５２）。そして、抽出した黒ランの長さが、文字の大きさγより大きいか否かを判定し（ステップｓ５３）、黒ランの長さが大きければ、カウンタをインクリメントするとともに、アキュムレータにその長さを加える（ステップｓ５４）。次に、対象とする処理ラインの走査が終了したか否かを判断して（ステップｓ５５）、対象となる処理ラインの走査が終了していなければ、ステップｓ５２に処理が戻り、走査が終了していれば、文字の大きさγの更新を行う（ステップｓ５６）。ここで更新後の文字の大きさγは、

新しい文字の大きさγ＝β２×アキュムレータの数値／カウンタの値・・・・・・〔３〕

により求める。この〔３〕式において、β２は0.6～1.0の範囲で設定するが、実験では0.85とすると、直線密度算出において適切であることが確認された。

【００６９】

図１６を例にして、文字の大きさγ、カウンタ、アキュムレータの変化の状態を説明する。文字の大きさγの初期値を「２４」、文字の大きさγ以上の黒ラン数をカウントするカウンタの初期値を「２０」、アキュムレータの初期値を「４８０（＝２０×２４）」とし、また、β２の値を０．８５とすると、第１の走査ラインＬ１の走査が終了した段階での文字の大きさγは、前記〔３〕式より、

γ＝０．８５×｛（４８０＋３６）／（２０＋１）｝　２１

となる。ここで、前記数値「３６」は図１６からもわかるように、文字の大きさγ（＝２４）よりも大きな黒ランの長さ、つまり、黒ランＢ１の画素数「３６」であり、第１の走査ラインＬ１の走査終了時点で、この「３６」がアキュムレータの値に加算される。また、この第１の走査ラインＬ１の走査終了時点では、文字の大きさγ（＝２４）よりも大きな黒ランの数は「１」であるので、この「１」がカウンタの値に加算されている。これにより計算されると、γ　２１となり、この値が新しい文字の大きさγとなる。

【００７０】

また、第２の走査ラインＬ２の走査終了時点では、γ（＝２１）より大きな黒ランが存在

しないので、γの値は更新されず、γ　２１のままである。そして、第３の走査ラインＬ３に処理が移って、この第３の走査ラインＬ３の走査終了時点では、
γ＝０．８５×｛（５１６＋４０）／（２１＋１））｝　２２
と更新される。ここで、前記数値「４０」は図１６からもわかるように、第３の走査ラインＬ３において、文字の大きさγ（＝２１）よりも大きな黒ランの長さ、つまり、黒ランＢ８の画素数「４０」であり、第３の走査ラインＬ１の走査終了時点で、この「４０」がアキュームレータの値（第２の走査ラインＬ２の走査終了時点におけるアキュームレータの数値「５１６」）に加算される。また、この第３の走査ラインＬ３の走査終了時点では、文字の大きさγ（＝２１）よりも大きな黒ランの数は「１」であるので、この「１」がカウンタの値に加算されている。これにより計算されると、γ　２２となり、この値が新しい文字の大きさとなる。

【００７１】
なお、前記した例では、初期値「２４」より、対象とする画像の横線の長さが長いにもかかわらず、更新後の文字の大きさγの値か小さくなっているが、十分な数のデータ量が集まることにより、更新後の文字の大きさは初期値「２４」を越え、文字大きさに適した値に更新されていく。

【００７２】
このようにして、全ラインについて処理を行い、全ラインの走査が終了したか否かを判断して（ステップｓ５７）、全ラインの処理が終了すると、走査が終了した時点における文字の大きさγの値をしきい値とする（ステップｓ５８）。

【００７３】
以上説明した方法により、外接矩形を用いたしきい値の算出と同等の結果が得られるとともに、メモリアクセスがラスタスキャン的に行えるため、処理の高速化が図れる。

【００７４】
なお、前記した例では、ラスタスキャンを横方向に行っているが、横方向に限られるわけではなく、メモリアクセスが容易に行われる方向であれば、縦横いすれの方向であってもよい。ここで、いずれの方向にラスタスキャンするかは、領域判定部の有効な直線、黒ランをカウントする際の走査する方向に合せるとよい。

【００７５】
前記したように文字の大きさを基準としたラスタスキャンによるしきい値の算出では、文字が直線により構成され、その直線の長さにより文字の大きさが求められることを前提とした処理であるため、主として、対象文字が日本語の場合におけるしきい値設定方法であるといえる。しかし、対象文字がαＮの場合には、もともと直線により文字が構成されていないため、この方法では、十分な情報を得ることができず、文字の大きさに適したしきい値の設定が困難となる。

【００７６】
そこで、このような問題を解決すべく、文字の大きさを文字の太さをもとにして求め、求めた文字の太さから文字の大きさを推測し、しきい値の算出を行うようにする。以下に、図１７を用いて処理の内容を説明する。図１７のフローチャートは、前記処理領域抽出部１１により抽出された或る１つの領域における処理を示すものである。

【００７７】
図１７において、始めに、文字の太さを求めるため、長さがある値以下の黒ランの個数をカウントするカウンタ、カウントの対象となった黒ランの長さの総和を求めるためのアキュムレータの初期化を行う（ステップｓ６１）。なお、この場合は、初期値はゼロでよい。次に対象とする走査ラインにおいて黒ランの抽出を行う（ステップｓ６２）。そして、抽出した黒ランの長さが、値Ｆ（この値Ｆについては後述する）以下で有るか否かを判断して（ステップｓ６３）、値Ｆ以下であれば、カウンタをインクリメントするとともに、アキュムレータにその黒ランの長さ（画素数）を加える（ステップ６４）。次に、対象とする処理ラインの走査が終了したかを判断し（ステップｓ６５）、終了していなければ、ステップｓ６２に処理が戻って、次の黒ランの抽出を行う。

【００７８】
ここで、値Ｆをどのようにするか問題となるが、文字の大きさの約１／３～１／４に相当する値とするとよい。よって、３００［ＤＰＩ］入力時、値Ｆは６～１０の範囲で設定するとよい。ここで、Ｆを固定値としたのは、一般に使われる文字の大きさは８～１４ポイントで、その範囲の文字の大きさならば、固定値でも文字の太さに十分対応できるからである。
【００７９】
図１８を例して、カウンタ、アキュムレータの変化のしかたを説明する。図１８は、αＮの文字例として「Japanese」という単語の中の「ａｐ」の部分を上端部からラスタスキャンした場合、第１，第２，第３の走査ラインＬ１，Ｌ２，Ｌ３の画素データを示すものである。値Ｆをここでは「８」とすると、第１の走査ラインＬ１には「８」以下の黒ランが１つ存在し（Ｂ１１で示す黒ラン）、その長さ（画素数）は同図からわかるように「４」である。よって、第１の走査ラインＬ１走査終了時のカウンタのカウント値は「１」、アキュムレータの値は「４」となる。同様に、第２の走査ラインＬ２の走査終了時のカウンタのカウント値は、Ｂ１２，Ｂ１３，Ｂ１４，Ｂ１５，Ｂ１６で示す５個の黒ランのカウント数が前記カウント数「１」に加算されて「６（＝１＋５）」となり、また、アキュムレータの値は前記５個の黒ランの長さが前記「４」に加算されて「２７（＝４＋４＋６＋５＋３＋５）」となる。さらに、第３の走査ラインの走査終了時のカウンタのカウント値は、Ｂ１７，Ｂ１８，Ｂ１９，Ｂ２０で示す４個の黒ランのカウント数が、前記カウント数「６」に加算されて「１０（＝６＋４）」となり、また、アキュムレータの値は４個の黒ランの長さが前記「２７」に加算されて「５２（＝２７＋５＋７＋７＋６）」となる。
【００８０】
そして、全ラインの走査が終了したか否かを判断して（ステップｓ６６）、前ラインの走査が終了すると、走査が終了した時点におけるカウンタ、アキュムレータの値を用いて、文字の太さを基準にしたしきい値の算出を行う。この文字の太さを基準にしたしきい値の算出方法は、
しきい値＝β３×アキュムレータの数値／カウンタの値　　・・・　〔４〕
とする。ここで、β３は3.0～4.0の範囲とする。なぜならば、値Ｆは文字の大きさの１／３～１／４を基準に定められた値であり、この逆数を掛けることにより文字の大きさが推測できるからである。なお、β３の値は、実験より、直線密度の算出において好結果が得られるように、β３＝3.4とするのが適当である。
【００８１】
なお、上記例では、ラスタスキャンを横方向に行っているが、横方向に限られるわけではなく、メモリアクセスが容易に行われる方向であれば、縦横いすれの方向であってもかまわない。ここで、いずれの方向にラスタスキャンするかは、領域判定部の有効な直線、黒ランをカウントする際の走査する方向に合せるとよい。
【００８２】
前記した方法により、メモリアクセスがラスタスキャン的に行えるため、処理の高速化が図れるとともに、主に直線で構成された文字に限らず、主に曲線で構成された文字に対しても、文字の大きさに適したしきい値を算出できる。
【００８３】
以上のようにしてしきい値が求められる。このしきい値は領域抽出部１１によって抽出された領域をもとに、領域判定部１３によってその領域内に存在する有効な直線を検出するためのしきい値であり、領域判定部１３では、このしきい値よりも大きな直線成分を求めて、その直線成分の数を基にして、前記したような方法によって、直線密度を求める。そして、その直線密度と或る値αとを比較して、直線密度が値α以上の時はその領域は日本語と判定し、値α未満のときはその領域はαＮであると判定する。
【００８４】
ここで値αについて説明する。図１９は、複数のサンプルに対して、黒ランの数を基準とした処理領域抽出、文字の大きさを基準としたしきい値算出を行い、領域判定部１３によ

り、［１］式によって算出された直線密度と、各領域の種類（日本語またはαＮ）の度数分布を示すものであり、Ｅ１０で示す領域が「日本語」と判定された度数分布、Ｅ２０の領域が「αＮ」と判定された度数分布を示している。これより、領域の種類判断のためのα（直線密度）を適切な値に設定することにより、領域における言語の判定が可能であることが分かる。図１９の場合は、領域における言語の種類の判定のためのαを0.01から0.001の範囲で設定するとよい。特にαを0.025とするのが最も適切であることがわかる。なお、この値は、特にしきい値算出部１２において、どの方法を用いてしきい値を求めるかによりやや変動する。よって、αを設定するにあたっては、複数のサンプルを用意し、図１９に示すような度数分布図を作成し、事前に適切な値を設定すれば、未知の文書画像にも適用できるようになる。

【００８５】

このように、αをたとえば0.025と設定したとすると、領域判定部１３で求められた直線密度が0.025以上のときはその領域は日本語の領域であると判定し、0.025未満の時は、その領域はαＮの領域であるとの判定を行う。

【００８６】

図２０に、行を単位として処理領域を抽出した場合、あるいは、ラスタスキャンにより黒ランの数を基準に処理領域を抽出した場合の、各処理単位の領域判定部の判定結果の例を示す。

【００８７】

図２０において、Ｅ１，Ｅ２，・・・，Ｅ７は行あるいは黒ラン数を基準として抽出された領域を示している。図２０からもわかるように、ほとんどの領域の判定結果が「日本語」であるのに、部分的に「αＮ」と誤って判定される場合がある（領域Ｅ６）。これは、特に、行を単位とした処理領域の抽出において、その行に含まれる文字数が少ない場合に起こりやすい。

【００８８】

このような誤りに対応するために、領域判定部１３に誤り訂正手段１３１を設けている。これにより、行を単位とした、あるいは、ラスタスキャンにより黒ランの数を基準に処理領域抽出した場合の図２０で示すような誤りに対応できるようにする。

【００８９】

前記誤り訂正手段１３１による誤り訂正処理の内容を図２１に示す。図２１において、まず、対象とする処理領域の種類（日本語かαＮ）の判定が終わった段階で、現在判定した領域の判定結果と、１つ前の領域の判定結果を比較する（ステップｓ７１）。判定結果が異なる場合には、２つ前の領域の判定結果を参照し、現在の処理領域の判定結果と２つ前の領域の判定結果とを比較する（ステップｓ７２）。両者が等しければ、１つ前の判定結果を現在の判定結果と同じにする（ステップＳ７３）。

【００９０】

以上の処理を図２０を例にとって説明する。図２０において、領域Ｅ７の領域判定結果は「日本語」であり、これが現在判定した領域の判定結果であるとする。この判定結果と、１つ前の領域Ｅ６の判定結果とを比較すると、領域Ｅ６の判定結果は「αＮ」となっている。このように、現在判定した領域Ｅ７の判定結果と、１つ前の領域Ｅ６の判定結果を比較して、判定結果が異なる場合には、２つ前の領域Ｅ５の判定結果を参照する。この場合、領域Ｅ５の判定結果は「日本語」であり、現在判定した領域Ｅ７と２つ前の領域Ｅ５の判定結果は同じものとなっている。したがって、このような場合には、１つ前の領域Ｅ６の判定結果「αＮ」は誤りとみなして、領域Ｅ６の判定結果は現在判定した領域Ｅ７の判定結果と同じ「日本語」であると判定し直す。このような処理を行うことにより、図２０で説明したような誤りを訂正することができる。なお、誤り訂正の際に、判断結果の参照範囲を２つ前の領域よりさらに前の領域の判定結果を参照するようにすれば、より精度の高い誤り訂正を行うことも可能となる。

【００９１】

以上説明したように、本発明の実施例によれば、文書の段落や行を基準として、あるいは

黒ラン数、画素の白黒反転数などを基準として処理領域の抽出を行った後、抽出された領域内の文字列における直線成分を或るしきい値を基に検出して、直線成分の密度（直線密度）を計算して、計算された直線密度をあらかじめ設定した値と比較して、その比較結果からその領域が日本語であるかαＮであるかを自動的に判定するようにしている。従って、このように判定された結果を、文字切り出し手段、さらには文字認識手段に与えることで、正確な切り出しおよび認識が可能となる。

【００９２】

なお、以上説明した実施例では、入力解像度が３００[ＤＰＩ]であることを前提としていたが、図１５における文字の大きさγの初期値、図１１における文字の太さを算出する際の値Ｆは、入力解像度に比例して値を変化させることにより、入力解像度は３００［ＤＰＩ］に限られることなく、他の解像度に対しても対応が可能となる。

【００９３】

【発明の効果】

以上説明したように、本発明の画像処理方法では、請求項１によれば、画像入力手段により入力された文書画像データから処理対象とする領域を抽出し、抽出された領域内における文書画像データから、その領域の言語の種類を判定をする際に必要な文字の大きさの指標となるしきい値を算出し、このしきい値を用いて前記抽出された領域の言語の種類を判定するために必要な直線密度を抽出し、当該領域の言語の種類の判断を行うようにしたので、処理領域を自動的に抽出することができるとともに、抽出された処理領域単位でその領域内が何の言語で書かれたものであるかの判定を行うことができ、この判定結果を文字の切り出しおよび文字認識処理に渡すことにより、言語の種類に対応した的確な切り出しおよび認識処理が可能となり、認識率の大幅な向上に寄与できる。

【００９４】

また、請求項２によれば、処理領域抽出工程は、処理対象とする文書画像データにおける段落を基準として処理領域の抽出を行なうようにしている。これは、言語の種類は段落単位で変化している場合も多いということに鑑みてなされたもので、段落を処理領域として抽出することにより、段落単位で言語の種類の判定を行うことができ、また、段落単位の領域抽出は、従来の簡単な技術を用いて行うことが可能となる。

【００９５】

また、請求項３によれば、処理領域抽出工程は、文書画像データにおける文字行を基準として処理領域の抽出を行なうようにしている。このように文字行を基準として処理領域の抽出を行うことにより、同一領域内に違った種類の言語が混在する確率を少なくすることができ、領域判定の正解率の向上が図れ、また、文字行を基準とした領域抽出は、従来の簡単な技術を用いて行うことが可能となる。

【００９６】

また、請求項４によれば、処理領域抽出工程は、文書画像データをラスタスキャン方式により処理ラインに従った走査により、文書画像データにおける画素の白黒反転回数を検出し、画素の白黒反転数があらかじめ設定した値に達した範囲を基準として処理領域の抽出を行なうようにしたので、メモリをランダムにアクセスすることなく、処理ラインに従った走査にて処理領域の抽出処理が行え、処理の高速化が図れ、また、領域を決定する白黒反転数の設定値を的確に選ぶことにより、前記文字行を基準とした領域抽出同様、同一領域内に違った種類の言語が混在する確率を少なくすることができ、領域判定の正解率の向上を図ることができる。

【００９７】

また、請求項５によれば、しきい値算出工程は、前記処理領域抽出工程により抽出された領域内に含まれる各文字の外接矩形を抽出し、各外接矩形の大きさの平均値を基準としてしきい値の算出を行なうようにしたので、従来の比較的簡単な技術を用いて簡単に文字の大きさに対応したしきい値の算出を行うことができる。

【００９８】

また、請求項６によれば、しきい値算出工程は、文書画像データをラスタスキャン方式に

より処理ラインに従った走査により、前記処理領域抽出工程により抽出された領域内に含まれる文字の大きさにほぼ相当する黒画素の連続したつながりの平均的な長さを求め、求めた平均的な長さをもとにして、文字の大きさに対応したしきい値の算出を行なうようにしたので、メモリをランダムにアクセスすることなく、処理ラインに従った走査にて、しきい値算出処理が行え、処理の高速化が図れ、また、文字の大きさにほぼ相当する直線的な黒画素の連続したつながりの平均的な長さをもとに、文字の大きさに対応したしきい値を算出していることから、特に、直線の多い日本語文書におけるしきい値算出に有効なものとすることができる。

【００９９】

また、請求項７によれば、しきい値算出工程は、文書画像データをラスタスキャン方式により処理ラインに従った走査により、前記処理領域抽出工程により抽出された領域内に含まれる文字の太さにほぼ相当する黒画素の連続したつながりの平均的な長さを求め、求めた平均的な長さをもとに、文字の大きさに対応したしきい値の算出を行なうようにしたので、メモリをランダムにアクセスすることなく、処理ラインに従った走査にて、しきい値算出処理が行え、処理の高速化が図れ、また、文字の太さにほぼ相当する黒画素の連続したつながりの平均的な長さをもとに、文字の大きさに対応したしきい値を算出していることから、特に、直線の少ないαＮの文書におけるしきい値算出にも対応できる。つまり、一般的に、文字の大きさと太さは比例関係にあり、文字の太さから文字の大きさを判断することができると考えられ、直線的な画素の連続したつながりの少ないαＮの文書に対しては、文字の太さを基に、しきい値を算出しようとするものであり、これによって、直線の少ないαＮの文書におけるしきい値算出にも対応できる。

【０１００】

また、請求項８によれば、領域判定工程は、前記処理領域抽出工程により抽出された領域内において、前記しきい値算出工程により算出したしきい値以上の連続した黒画素の総数と、連続した黒画素の総数との比をとることで、文字の大きさに依存しない直線密度を求め、この直線密度の値を基にして、当該領域内の言語の種類の判断を行なうようにしている。すなわち、領域判定を行う際には、当該領域内の画像データの直線成分を検出して、文字の大きさに依存しない直線密度を求め、この直線密度を基にその領域内の言語の種類を判定している。これは、言語として、日本語とαＮを考えた場合、日本語は画像データに直線成分が多く、αＮは直線成分が殆どないことに着目して、直線密度を求めて、この直線密度によって、当該領域内の言語の種類の判定を行うものであり、これにより、抽出された領域の言語の種類の判定を確実にしかも簡単に行うことがでる。

【０１０１】

また、請求項９によれば、領域判定工程は、前記処理領域抽出工程により抽出された領域ごとに、その領域の言語の種類の判定を行った後、領域の種類の判定を行なう基準となる複数の処理領域の判定結果を用いて、誤判定とみなされる領域の判定結果の修正を行なうようにしたので、たとえば、行単位で領域判定が行われたような場合、ある行の文字数がきわめて少ないためにその行が誤判定されたような場合にもその誤りを訂正することができる。

【０１０２】

また、本発明の画像処理装置は、請求項１０によれば、画像入力手段により入力された文書画像データから処理対象とする領域を抽出する処理領域抽出手段と、この処理領域抽出手段により抽出された領域内における文書画像データから、その領域の言語の種類を判定をする際に必要な文字の大きさの指標となるしきい値を算出するしきい値算出手段と、このしきい値算出手段により算出されたしきい値を用いて、前記処理領域抽出手段により抽出された領域の言語の種類を判定するために必要な直線密度を抽出し、当該領域の言語の種類の判断を行う領域判定手段とを有したので、処理領域を自動的に抽出することができるとともに、抽出された処理領域単位でその領域内が何の言語で書かれたものであるかの判定を行うことができ、この判定結果を文字の切り出しおよび文字認識処理に渡すことにより、言語の種類に対応した的確な切り出しおよび認識処理が可能となり、認識率の大

幅な向上に寄与できる。

【０１０３】

また、請求項１１によれば、処理領域抽出手段は、処理対象とする文書画像データにおける段落を基準として処理領域の抽出を行なうようにしている。これは、言語の種類は段落単位で変化している場合も多いということに鑑みてなされたもので、段落を処理領域として抽出することにより、段落単位で言語の種類の判定を行うことができ、また、段落単位の領域抽出は、従来の簡単な技術を用いて行うことが可能となる。

【０１０４】

また、請求項１２によれば、処理領域抽出手段は、文書画像データにおける文字行を基準として処理領域の抽出を行なうようにしている。このように文字行を基準として処理領域の抽出を行うことにより、同一領域内に違った種類の言語が混在する確率を少なくすることができ、領域判定の正解率の向上が図れ、また、文字行を基準とした領域抽出は、従来の簡単な技術を用いて行うことが可能となる。

【０１０５】

また、請求項１３によれば、処理領域抽出手段は、文書画像データをラスタスキャン方式により処理ラインに従った走査により、文書画像データにおける画素の白黒反転回数を検出し、画素の白黒反転数があらかじめ設定した値に達した範囲を基準として処理領域の抽出を行なうようにしたので、メモリをランダムにアクセスすることなく、処理ラインに従った走査にて処理領域の抽出処理が行え、処理の高速化が図れ、また、領域を決定する白黒反転数の設定値を的確に選ぶことにより、前記文字行を基準とした領域抽出同様、同一領域内に違った種類の言語が混在する確率を少なくすることができ、領域判定の正解率の向上を図ることができる。

【０１０６】

また、請求項１４によれば、しきい値算出手段は、前記処理領域抽出手段により抽出された領域内に含まれる各文字の外接矩形を抽出し、各外接矩形の大きさの平均値を基準としてしきい値の算出を行なうようにしたので、従来の比較的簡単な技術を用いて簡単に文字の大きさの指標となるしきい値の算出を行うことができる。

【０１０７】

また、請求項１５によれば、しきい値算出手段は、文書画像データをラスタスキャン方式により処理ラインに従った走査により、前記処理領域抽出手段により抽出された領域内に含まれる文字の大きさにほぼ相当する黒画素の連続したつながりの平均的な長さを求め、求めた平均的な長さをもとに、文字の大きさに対応したしきい値の算出を行なうようにしたので、メモリをランダムにアクセスすることなく、処理ラインに従った走査にて、しきい値算出処理が行え、処理の高速化が図れ、また、文字の大きさにほぼ相当する直線的な黒画素の連続したつながりの平均的な長さをもとに、文字の大きさに対応したしきい値を算出していることから、特に、直線の多い日本語文書におけるしきい値算出に有効なものとすることができる。

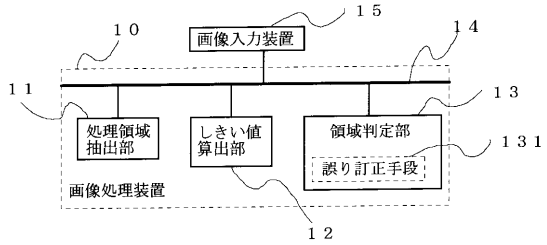
【０１０８】

また、請求項１６によれば、しきい値算出手段は、文書画像データをラスタスキャン方式により処理ラインに従った走査により、前記処理領域抽出手段により抽出された領域内に含まれる文字の太さにほぼ相当する黒画素の連続したつながりの平均的な長さを求め、求めた平均的な長さをもとに、文字の大きさに対応したしきい値の算出を行なうようにしたので、メモリをランダムにアクセスすることなく、処理ラインに従った走査にて、しきい値算出処理が行え、処理の高速化が図れ、また、文字の太さにほぼ相当する黒画素の連続したつながりの平均的な長さを基に文字の大きさに対応したしきい値を算出していることから、特に、直線の少ないαＮの文書におけるしきい値算出にも対応できる。つまり、一般的に、文字の大きさと太さは比例関係にあり、文字の太さから文字の大きさを判断することができると考えられ、直線的な画素の連続したつながりの少ないαＮの文書に対しては、文字の太さを基に、しきい値を算出しようとするものであり、これによって、直線の少ないαＮの文書におけるしきい値算出にも対応できる。

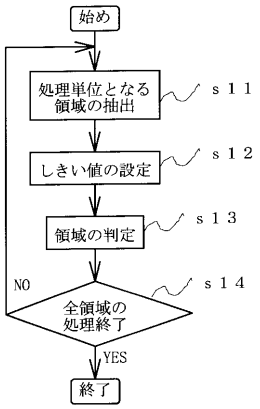
【０１０９】
また、請求項１７によれば、領域判定手段は、前記処理領域抽出手段により抽出された領域内において、前記しきい値算出手段により算出したしきい値以上の連続した黒画素の総数と、連続した黒画素の総数との比をとることで、文字の大きさに依存しない直線密度を求め、この直線密度の値を基にして、当該領域内の言語の種類の判断を行なうようにしている。すなわち、領域判定を行う際には、当該領域内の画像データの直線成分を検出して、文字の大きさに依存しない直線密度を求め、この直線密度を基にその領域内の言語の種類を判定している。これは、言語として、日本語とαＮを考えた場合、日本語は画像データに直線成分が多く、αＮは直線成分が殆どないことに着目して、直線密度を求めて、この直線密度によって、当該領域内の言語の種類の判定を行うものであり、これにより、抽出された領域の言語の種類の判定を確実にしかも簡単に行うことがでる。
【０１１０】
また、請求項１８によれば、領域判定手段は、誤り訂正手段を設け、前記処理領域抽出手段により抽出された領域ごとに、その領域の言語の種類の判定を行った後、領域の種類の判定を行なう基準となる複数の処理領域の判定結果を用いて、誤判定とみなされる領域の判定結果の修正を行なうようにしたので、たとえば、行単位で領域判定が行われたような場合、ある行の文字数がきわめて少ないためにその行が誤判定されたような場合にもその誤りを訂正することができる。
【図面の簡単な説明】
【図１】本発明の実施例の構成をを説明するブロック図。
【図２】同実施例の全体的な処理をソフトウエア的に行う場合のフローチャート。
【図３】同実施例の全体的な処理をハードウエアにより行う場合の動作説明図。
【図４】処理領域を段落単位で抽出するための文書例を示す図。
【図５】処理領域を行単位で抽出する例を説明する図。
【図６】処理領域の抽出処理を説明するフローチャート。
【図７】画素の白黒反転数を基に処理領域の抽出を行う場合、白黒反転数の検出例を説明する図。
【図８】処理領域の判定に用いられる黒ランの数と領域判定の正解率の関係を説明する図。
【図９】領域判定を行うために日本語の文字とαＮの文字との特質の違いを説明する図。
【図１０】領域判定を行うための直線密度の算出を説明する図。
【図１１】直線密度を基に領域判定処理を行うフローチャート。
【図１２】しきい値算出のための外接矩形を説明する図。
【図１３】外接矩形を求めるための文字の切り出し例を説明する図。
【図１４】しきい値算出処理の概略的なフローチャート。
【図１５】ラスタスキャンにより得られた文字の直線成分の長さを基にして、文字の大きさに対応したしきい値を決定する処理を説明するフローチャート。
【図１６】文字の直線成分の長さを基にして、文字の大きさに対応したしきい値を算出する具体例を説明する図。
【図１７】ラスタスキャンにより文字の太さを基にして、文字の大きさに対応したしきい値を決定する処理を説明するフローチャート。
【図１８】文字の太さを基にして文字の大きさの指標となるしきい値を算出する具体例を説明する図。
【図１９】領域判定を行う際の直線密度に対する領域の度数分布を示す図。
【図２０】行単位あるいは黒ラン数により領域判定をした場合における判定結果例を示す図。
【図２１】行単位あるいは黒ラン数により領域判定をした際の判定結果に誤りが有る場合の誤り訂正処理を説明する図。
【図２２】従来技術における文字切り出しおよび文字認識処理の問題点を説明する図。
【符号の説明】

１０・・・画像処理装置
１１・・・処理領域抽出部
１２・・・しきい値算出部
１３・・・領域判定部
１４・・・バス
１５・・・画像入力装置
１３１・・・誤り判定手段

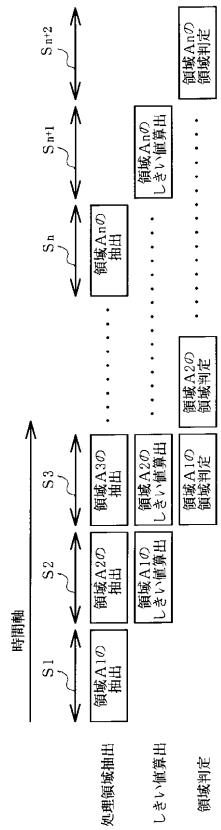
【図１】



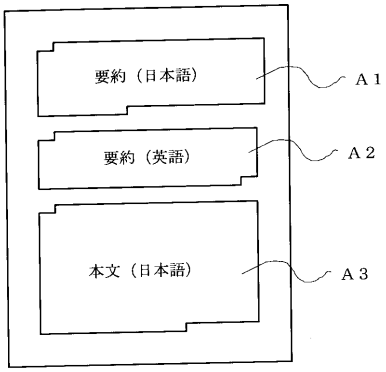

【図２】



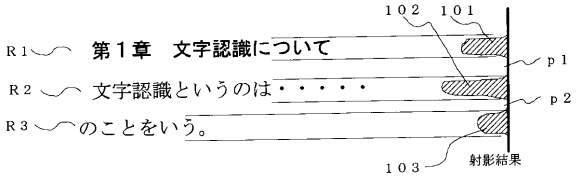

【図３】



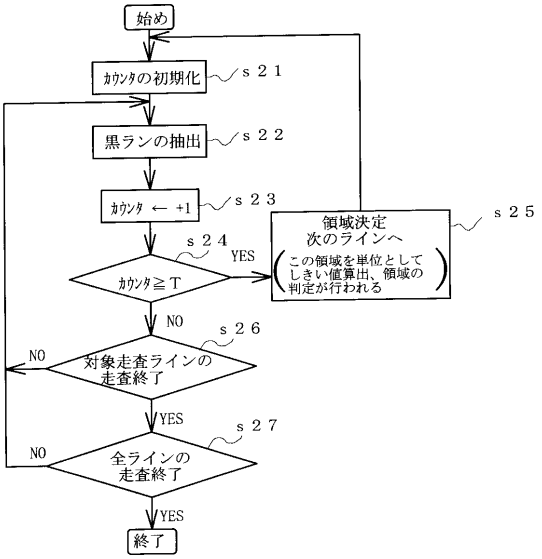
【図４】



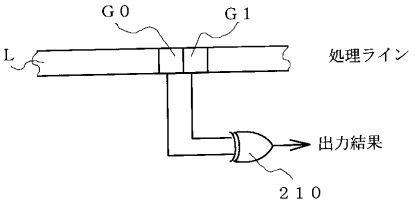

【図５】



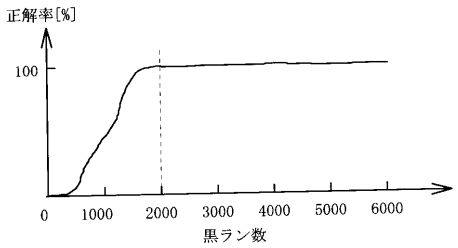

【図６】



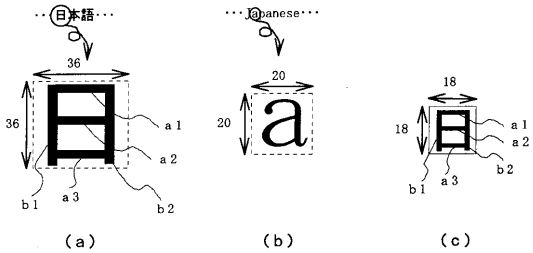

【図７】



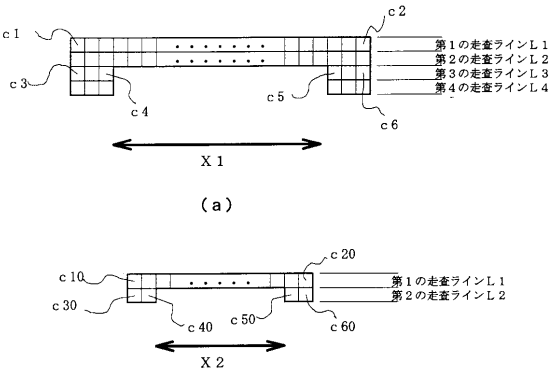

【図８】



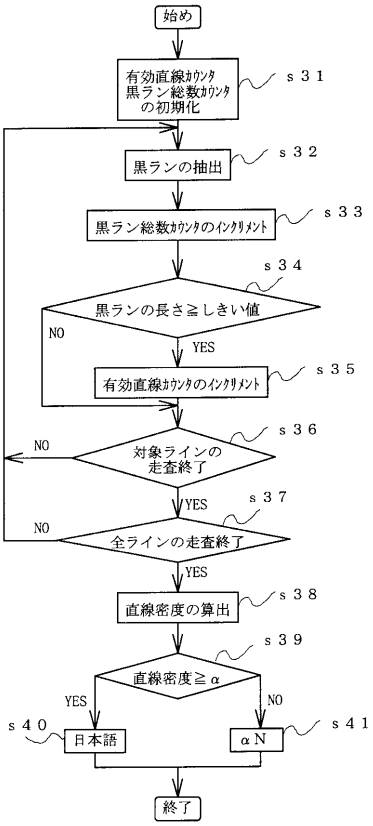

【図９】



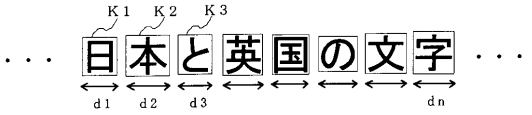

【図１０】



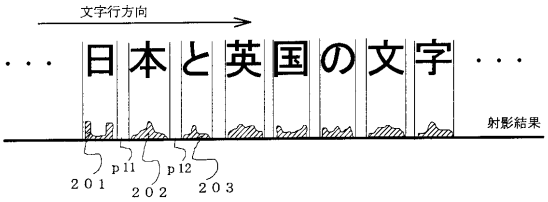
【図１１】



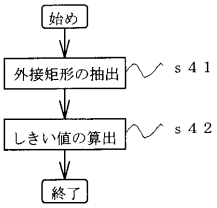

【図１２】



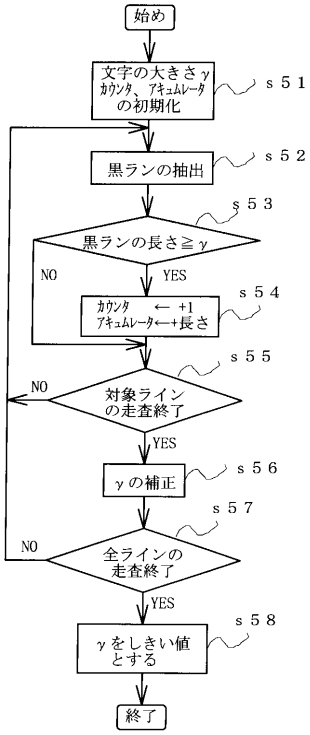

【図１３】



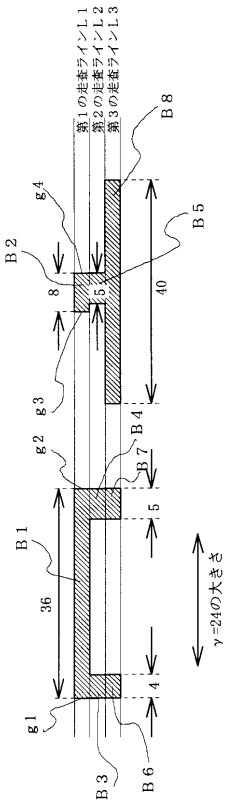

【図１４】



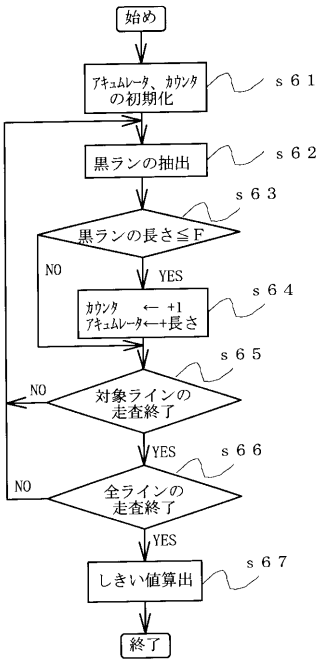

【図１５】



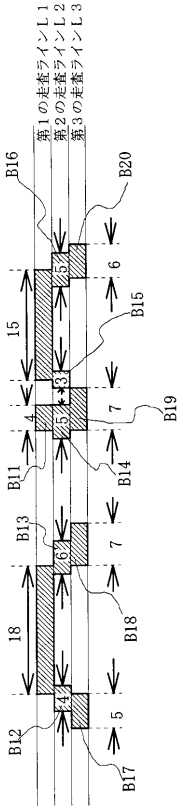

【図１６】



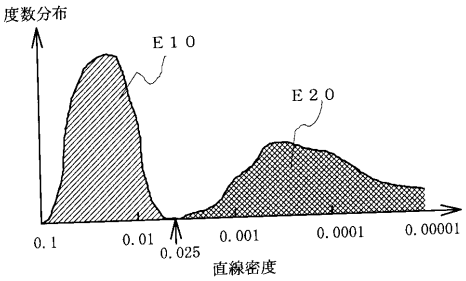
【図１７】



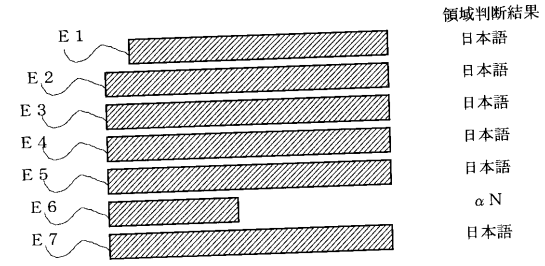

【図１８】



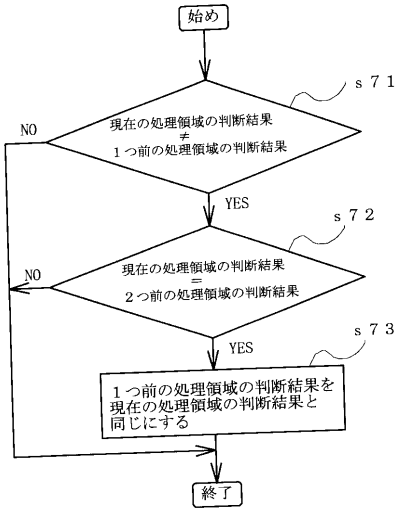

【図１９】



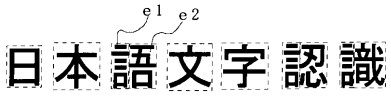

【図２０】



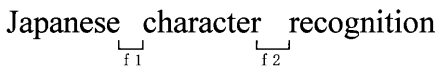

【図２１】



【図２２】




(a)




(b)