

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4266784号  
(P4266784)

(45) 発行日 平成21年5月20日 (2009.5.20)

(24) 登録日 平成21年2月27日 (2009.2.27)

(51) Int. Cl.			F I		
<b>G06T</b>	<b>11/60</b>	<b>(2006.01)</b>	G06T	11/60	100C
<b>G06K</b>	<b>9/00</b>	<b>(2006.01)</b>	G06K	9/00	S
<b>G06T</b>	<b>1/00</b>	<b>(2006.01)</b>	G06T	1/00	500B
<b>H04N</b>	<b>1/00</b>	<b>(2006.01)</b>	H04N	1/00	107Z
<b>H04N</b>	<b>1/387</b>	<b>(2006.01)</b>	H04N	1/387	

請求項の数 19 (全 29 頁)

(21) 出願番号	特願2003-385184 (P2003-385184)	(73) 特許権者	000001007
(22) 出願日	平成15年11月14日 (2003.11.14)		キヤノン株式会社
(65) 公開番号	特開2005-149096 (P2005-149096A)		東京都大田区下丸子3丁目30番2号
(43) 公開日	平成17年6月9日 (2005.6.9)	(74) 代理人	100076428
審査請求日	平成18年11月14日 (2006.11.14)		弁理士 大塚 康德
前置審査		(74) 代理人	100112508
			弁理士 高柳 司郎
		(74) 代理人	100115071
			弁理士 大塚 康弘
		(74) 代理人	100116894
			弁理士 木村 秀二
		(74) 代理人	100130409
			弁理士 下山 治
		(74) 代理人	100134175
			弁理士 永川 行光

最終頁に続く

(54) 【発明の名称】 画像処理システム及び画像処理方法

(57) 【特許請求の範囲】

【請求項1】

原稿を読み取ってイメージ情報を取得する読取手段と、  
 前記読取手段により得られたイメージ情報の特徴を認識する認識手段と、  
 前記認識手段で認識された特徴に応じて前記読取手段により得られたイメージ情報を文字コード化する文字処理手段と、  
 前記認識手段で認識された特徴に応じて前記読取手段により得られたイメージ情報をベクトルデータに変換するベクトル化手段と、  
 前記認識手段で認識された特徴に応じて前記読取手段により得られたイメージ情報を予め決められた画像形式に変換する画像変換手段と、  
 前記文字処理手段、前記ベクトル化手段、前記画像変換手段のそれぞれを動作させるか否かを示すパラメータを、ユーザによる指定に応じて可変設定する設定手段と、  
 前記設定手段で設定されたパラメータに応じて前記文字処理手段、前記ベクトル化手段、前記画像変換手段の動作を制御する制御手段とを備え、  
 前記制御手段は、前記文字処理手段、前記ベクトル化手段、前記画像変換手段のうち前記設定手段で設定されたパラメータに応じた手段を動作させることを特徴とする画像処理システム。

【請求項2】

サーバが保持する電子ファイルの中から前記読取手段により得られたイメージ情報に対応する電子ファイルを検索する検索手段をさらに備え、

前記制御手段は、前記検索手段により電子ファイルが検索できない場合、前記読取手段により得られたイメージ情報に基づき前記文字処理手段、前記ベクトル化手段、前記画像変換手段の少なくともいずれかを用いた変換を実行することを特徴とする請求項 1 に記載の画像処理システム。

【請求項 3】

前記検索手段が、前記イメージ情報から前記原稿中に付加されている該原稿に対応する電子ファイルの格納場所を示す情報を認識することによって、該電子ファイルを検索することを特徴とする請求項 2 に記載の画像処理システム。

【請求項 4】

前記検索手段が、前記イメージ情報から前記原稿中に埋め込まれている情報に基づいて前記サーバが保持する電子ファイルの中から前記原稿に対応する電子ファイルを検索することを特徴とする請求項 2 に記載の画像処理システム。

10

【請求項 5】

前記ベクトル化手段が、  
前記読取手段で取得された前記イメージ情報に対してアウトライン化を行うアウトライン化手段と、  
アウトライン化された前記イメージ情報に関するアウトラインデータに基づいて、該イメージ情報の直線近似、曲線近似又は図形近似を行う関数近似手段と、  
を備えることを特徴とする請求項 1 から 4 までのいずれか 1 項に記載の画像処理システム。

20

【請求項 6】

前記文字処理手段が、  
原稿中の文字を文字認識処理する文字認識手段と、  
前記文字認識手段による文字認識処理結果に基づいて前記文字をフォントデータに置き換えるフォント化手段と、  
を備えることを特徴とする請求項 1 から 5 までのいずれか 1 項に記載の画像処理システム。

【請求項 7】

前記制御手段が、前記イメージ情報を、前記認識手段による認識結果に応じて複数のオブジェクトに分割し、それぞれのオブジェクトを前記文字処理手段、前記ベクトル化手段、前記画像変換手段のいずれかによって変換処理することを特徴とする請求項 1 から 6 までのいずれか 1 項に記載の画像処理システム。

30

【請求項 8】

前記制御手段が、前記文字処理手段、前記ベクトル化手段、前記画像変換手段の少なくともいずれかで変換が実行された後の情報を予め決められたソフトウェアアプリケーションで取り扱い可能なフォーマットに変換するフォーマット変換手段をさらに備えることを特徴とする請求項 7 に記載の画像処理システム。

【請求項 9】

前記文字処理手段、前記ベクトル化手段、前記画像変換手段の少なくともいずれかで変換されたベクトルデータを予め決められた格納場所に格納する格納手段と、  
前記ベクトルデータが格納された格納場所に関する情報を付加情報として生成する生成手段と、  
前記付加情報を通知する通知手段と、  
をさらに備えることを特徴とする請求項 1 から 8 までのいずれか 1 項に記載の画像処理システム。

40

【請求項 10】

前記検索手段が、さらに、前記読取手段によって取得された前記原稿のイメージ情報中に予め決められた形式の情報が含まれている場合に前記電子ファイルを検索することを特徴とする請求項 3 に記載の画像処理システム。

【請求項 11】

50

原稿を読み取ってイメージ情報を取得する読取手段と、  
前記読取手段により得られたイメージ情報を文字コード化する文字処理手段と、  
前記読取手段により得られたイメージ情報をベクトルデータに変換するベクトル化手段と、

前記読取手段により得られたイメージ情報を予め決められた画像形式に変換する画像変換手段と、

前記文字処理手段、前記ベクトル化手段、前記画像変換手段のいずれを用いるかを、ユーザによる指定に応じて設定する設定手段と、

前記読取手段により得られたイメージ情報を、前記文字処理手段、前記ベクトル化手段、前記画像変換手段のうち、前記設定手段で設定された手段を用いて変換するよう制御する制御手段と、

10

を備えることを特徴とする画像処理システム。

【請求項 1 2】

サーバが保持する電子ファイルの中から前記読取手段により得られたイメージ情報に対応する電子ファイルを検索する検索手段をさらに備え、

前記文字処理手段、前記ベクトル化手段、前記画像変換手段は、前記検索手段により検索できない場合、前記読取手段により得られたイメージ情報に基づく変換を実行することを特徴とする請求項 1 1 に記載の画像処理システム。

【請求項 1 3】

前記原稿に対応する電子ファイルの有無に関わらず、前記検索手段による電子ファイルの検索処理を実行させるか否かを選択する選択手段をさらに備えることを特徴とする請求項 2 から 4 まで、及び 1 2 のいずれか 1 項に記載の画像処理システム。

20

【請求項 1 4】

前記設定手段が、

前記原稿を文字、写真、線画、又は表のいずれかの属性に設定する第 1 の設定手段と、前記文字処理手段による変換時の文字認識処理、フォント化処理、フォント指定処理、前記ベクトル化手段による関数近似処理、又は前記画像変換手段による圧縮処理におけるパラメータを設定する第 2 の設定手段と、

を備えることを特徴とする請求項 1 から 1 3 までのいずれか 1 項に記載の画像処理システム。

30

【請求項 1 5】

前記制御手段が、前記第 1 又は第 2 の設定手段によって指定された内容に基づいて、前記イメージ情報の変換処理の内容を切り替えることを特徴とする請求項 1 4 に記載の画像処理システム。

【請求項 1 6】

アクセス制限のある電子ファイルを保持するサーバから取得するに先立って、ユーザに対して該サーバから認証情報の入力を要求するパスワード要求手段と、

入力された認証情報が正しいかどうかを判定するパスワード判定手段と、

正しい認証情報が入力された場合、前記サーバに対する前記電子ファイルへのアクセスを許可する許可手段と、

40

をさらに備え、

前記許可手段によって前記電子ファイルへのアクセスが許可されたことを条件として該電子ファイルを取得することを特徴とする請求項 2 から 4 まで、及び 1 2 のいずれか 1 項に記載の画像処理システム。

【請求項 1 7】

読取手段、認識手段、文字処理手段、ベクトル化手段、画像変換手段を有する画像処理装置を用いた画像処理方法であって、

前記読取手段を用いて原稿を読み取ってイメージ情報を取得する読取工程と、前記読取工程により得られたイメージ情報の特徴を、前記認識手段を用いて認識する認識工程と、

前記認識工程で認識された特徴に応じて前記読取工程により得られたイメージ情報を、

50

前記文字処理手段を用いて文字コード化する文字処理工程と、

前記認識工程で認識された特徴に応じて前記読取工程により得られたイメージ情報を、前記ベクトル化手段を用いてベクトルデータに変換するベクトル化工程と、

前記認識工程で認識された特徴に応じて前記読取工程により得られたイメージ情報を、前記画像変換手段を用いて予め決められた画像形式に変換する画像変換工程と、前記文字処理手段、前記ベクトル化手段、前記画像変換手段のそれぞれを動作させるか否かを示す情報を含むパラメータを、ユーザによる指定に応じて可変設定する設定工程と、

前記設定工程で設定されたパラメータに応じて前記文字処理手段、前記ベクトル化手段、前記画像変換手段の動作を制御する制御工程とを備え、

前記制御工程は、前記文字処理手段、前記ベクトル化手段、前記画像変換手段のうち前記設定工程で設定された情報に応じた手段を動作させることを特徴とする画像処理方法。

【請求項 18】

読取手段、文字処理手段、ベクトル化手段、画像変換手段を有する画像処理装置を用いた画像処理方法であって、

前記読取手段を用いて原稿を読み取ってイメージ情報を取得する読取工程と、前記読取工程により得られたイメージ情報を、前記文字処理手段を用いて文字コード化する文字処理工程と、

前記読取工程により得られたイメージ情報を、前記ベクトル化手段を用いてベクトルデータに変換するベクトル化工程と、

前記読取工程により得られたイメージ情報を、前記画像変換手段を用いて予め決められた画像形式に変換する画像変換工程と、

前記文字処理手段、前記ベクトル化手段、前記画像変換手段のいずれを用いるかを、ユーザによる指定に応じて設定する設定工程と、

前記読取工程により得られたイメージ情報を、前記文字処理手段、前記ベクトル化手段、前記画像変換手段のうち、前記設定工程で設定された手段を用いて変換するよう制御する制御工程と

を有することを特徴とする画像処理方法。

【請求項 19】

請求項 1 乃至 16 のいずれか 1 項に記載の画像処理システムをコンピュータを用いて実現させるための記憶媒体に格納可能なプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、複写機等の画像処理装置で読み取った画像を既存の文書作成アプリケーションソフトで再編集が容易な文字コードやベクトルデータ等に変換する画像処理システム及び画像処理方法に関する。

【背景技術】

【0002】

近年、環境問題がクローズアップされている中で、オフィス等でのペーパーレス化が急速に進んでいる。そこで、従来からバインダー等で蓄積・保存された紙文書をスキャナで読み取って、ポータブルドキュメントフォーマット（以下、「PDF」と略す。）に変換し、画像記憶装置（データベース）に蓄積・保存する文書管理システムが構築されている。

【0003】

一方、機能が拡張されたデジタル複合機（以下、「MFP」と略す。）には、画像（イメージデータ）を紙文書として印刷・出力する際に、当該画像が記憶されている画像記憶装置内のポイント情報を紙文書の表紙或いは記載情報中に付加情報として記録しておくものがある。そして、当該画像が記録された紙文書を複写する場合には、表紙等に付加されているポイント情報からオリジナルの画像が格納されている画像記憶装置内の格納場所を検出し、紙文書から読み取られた画像を再印刷等に用いるのではなく、画像記憶装置内に

10

20

30

40

50

記憶されているオリジナルの画像を編集や再印刷に直接用いることができる。これにより、紙文書として保存する必要がなく、紙文書から読み取られた画像を何度も再利用することによって生じる画質の劣化等の問題を防ぐことができる（例えば、特許文献1参照。）

【特許文献1】特開平10-143414号公報

【発明の開示】

【発明が解決しようとする課題】

【0004】

しかしながら、前者は紙文書をコンパクトな情報量のPDFファイルとして保存することが可能であるが、ファイル全体が1つのイメージデータとして保存されるので、当該文書の一部のオブジェクトだけを部分的に検索して再利用することは難しい。従って、当該データを再利用する場合は、図や表等については、新たにアプリケーションソフトを用いて作成し直さなければならない。

10

【0005】

また、後者は出力された紙文書に対応するオリジナルの電子ファイルに直接アクセスできるため容易に再利用・再編集することができる。しかし、外部から入手した新しい原稿やオリジナルの電子ファイルの所在が不明な古い紙文書には対応することができない。そこで、原稿が読み取られた際に、対応する電子ファイルが既に存在する場合は効率良くそれらを検索すると共に、存在しない場合には次回以降検索できるように効率良くベクトル化して保存しておく必要がある。

20

【0006】

本発明は、このような事情を考慮してなされたものであり、紙文書等の原稿を再利用・再編集を容易とする形式として好適に保存することができる画像処理システム及び画像処理方法を提供することを目的とする。

【課題を解決するための手段】

【0007】

上記目的を達成するため、本発明に係る画像処理システムは、  
 原稿を読み取ってイメージ情報を取得する読取手段と、  
 前記読取手段により得られたイメージ情報の特徴を認識する認識手段と、  
 前記認識手段で認識された特徴に応じて前記読取手段により得られたイメージ情報を文字コード化する文字処理手段と、  
 前記認識手段で認識された特徴に応じて前記読取手段により得られたイメージ情報をベクトルデータに変換するベクトル化手段と、  
 前記認識手段で認識された特徴に応じて前記読取手段により得られたイメージ情報を予め決められた画像形式に変換する画像変換手段と、  
 前記文字処理手段、前記ベクトル化手段、前記画像変換手段のそれぞれを動作させるか否かを示すパラメータを、ユーザによる指定に応じて可変設定する設定手段と、  
 前記設定手段で設定されたパラメータに応じて前記文字処理手段、前記ベクトル化手段、前記画像変換手段の動作を制御する制御手段とを備え、  
 前記制御手段は、前記文字処理手段、前記ベクトル化手段、前記画像変換手段のうち前記設定手段で設定されたパラメータに応じた手段を動作させることを特徴とする。

30

40

【0008】

上記目的を達成するため、本発明に係る画像処理システムは、  
 原稿を読み取ってイメージ情報を取得する読取手段と、  
 前記読取手段により得られたイメージ情報を文字コード化する文字処理手段と、  
 前記読取手段により得られたイメージ情報をベクトルデータに変換するベクトル化手段と、  
 前記読取手段により得られたイメージ情報を予め決められた画像形式に変換する画像変換手段と、  
 前記文字処理手段、前記ベクトル化手段、前記画像変換手段のいずれを用いるかを、ユ

50

ーザによる指定に応じて設定する設定手段と、

前記読取手段により得られたイメージ情報を、前記文字処理手段、前記ベクトル化手段、前記画像変換手段のうち、前記設定手段で設定された手段を用いて変換するよう制御する制御手段と、

を備えることを特徴とする。

【0009】

上記目的を達成するため、本発明に係る画像処理方法は、

読取手段、認識手段、文字処理手段、ベクトル化手段、画像変換手段を有する画像処理装置を用いた画像処理方法であって、

前記読取手段を用いて原稿を読み取ってイメージ情報を取得する読取工程と、前記読取工程により得られたイメージ情報の特徴を、前記認識手段を用いて認識する認識工程と、

前記認識工程で認識された特徴に応じて前記読取工程により得られたイメージ情報を、前記文字処理手段を用いて文字コード化する文字処理工程と、

前記認識工程で認識された特徴に応じて前記読取工程により得られたイメージ情報を、前記ベクトル化手段を用いてベクトルデータに変換するベクトル化工程と、

前記認識工程で認識された特徴に応じて前記読取工程により得られたイメージ情報を、前記画像変換手段を用いて予め決められた画像形式に変換する画像変換工程と、前記文字処理手段、前記ベクトル化手段、前記画像変換手段のそれぞれを動作させるか否かを示す情報を含むパラメータを、ユーザによる指定に応じて可変設定する設定工程と、

前記設定工程で設定されたパラメータに応じて前記文字処理手段、前記ベクトル化手段、前記画像変換手段の動作を制御する制御工程とを備え、

前記制御工程は、前記文字処理手段、前記ベクトル化手段、前記画像変換手段のうち前記設定工程で設定された情報に応じた手段を動作させることを特徴とする。

【0010】

上記目的を達成するため、本発明に係る画像処理方法は、

読取手段、文字処理手段、ベクトル化手段、画像変換手段を有する画像処理装置を用いた画像処理方法であって、

前記読取手段を用いて原稿を読み取ってイメージ情報を取得する読取工程と、前記読取工程により得られたイメージ情報を、前記文字処理手段を用いて文字コード化する文字処理工程と、

前記読取工程により得られたイメージ情報を、前記ベクトル化手段を用いてベクトルデータに変換するベクトル化工程と、

前記読取工程により得られたイメージ情報を、前記画像変換手段を用いて予め決められた画像形式に変換する画像変換工程と、

前記文字処理手段、前記ベクトル化手段、前記画像変換手段のいずれを用いるかを、ユーザによる指定に応じて設定する設定工程と、

前記読取工程により得られたイメージ情報を、前記文字処理手段、前記ベクトル化手段、前記画像変換手段のうち、前記設定工程で設定された手段を用いて変換するよう制御する制御工程と

を有することを特徴とする。

【0011】

本発明に係るプログラムは、上記画像処理システムの各手段をコンピュータに実現させることを特徴とする。

【発明の効果】

【0038】

本発明によれば、紙文書等の原稿を再利用・再編集が容易な形式のデータとして取得することができる。しかも、イメージ情報の特徴認識の際のパラメータを可変設定したり、変換の際にどの変換処理を用いるかを設定したりすることにより容易に所望の形式のデータを取得することが可能となる。

【発明を実施するための最良の形態】

10

20

30

40

50

## 【 0 0 3 9 】

以下、図面を参照して、本発明の実施形態について説明する。

## 【実施例 1】

## 【 0 0 4 0 】

図 1 は、本発明の第 1 の実施例に係る画像処理システムの構成を示すブロック図である。図 1 に示す画像処理システムは、オフィス 10 とオフィス 20 とをインターネット等のネットワーク 104 で接続された環境で実現される。

## 【 0 0 4 1 】

オフィス 10 内に構築された LAN 107 には、MFP 100 と、MFP 100 を制御するマネジメント PC 101 と、クライアント PC 102 と、文書管理サーバ 106a と、そのデータベース 105a 及びプロキシサーバ 103a が接続されている。また、オフィス 20 内に構築された LAN 108 には、文書管理サーバ 106b と、そのデータベース 105b 及びプロキシサーバ 103b が接続されている。尚、クライアント PC 102 は、外部記憶部、検索イメージ入力部及び検索結果出力部を備えている。また、LAN 107 及びオフィス 20 内の LAN 108 は、プロキシサーバ 103a、103b を介してインターネット等のネットワーク 104 に接続されている。

## 【 0 0 4 2 】

MFP 100 は、本実施形態において紙文書を光学的に読み取って画像信号に変換する画像読み取り処理と、読み取った画像信号に対する画像処理の一部を担当し、画像信号は LAN 109 を用いてマネジメント PC 101 に入力する。尚、マネジメント PC 101 は、通常の PC でも実現可能であり、内部に画像記憶部、画像処理部、表示部及び入力部を備える。尚、マネジメント PC 101 は、その一部又は全部を MFP 100 と一体化して構成してもよい。

## 【 0 0 4 3 】

図 2 は、本発明の第 1 の実施例に係る MFP 100 の構成を示すブロック図である。図 2 において、オートドキュメントフィーダ（以下、「ADF」と略す。）を含む画像読み取り部 110 は、束状或いは 1 枚の原稿画像を内部に備える光源で照射し、原稿反射像をレンズで固体撮像素子上に結像し、固体撮像素子からラスタ状の画像読み取り信号を例えば 600 dpi の密度のイメージ情報として得る。そして通常の複写機能を用いる場合は、この画像信号をデータ処理部 115 で記録信号へ画像処理し、複数毎複写の場合は記憶装置 111 に一旦 1 ページ分の記録データを保持した後、形成装置 112 に順次出力して紙上に画像を形成する。

## 【 0 0 4 4 】

一方、クライアント PC 102 から出力されるプリントデータは、LAN 107 から MFP 100 に入力され、ネットワーク IF 114 を経てデータ処理装置 115 で記録可能なラスタデータに変換された後、形成装置 112 に出力して紙上に記録画像として形成される。

## 【 0 0 4 5 】

MFP 100 への操作者の指示は、MFP 100 に装備されているキー等の入力装置 113、或いはマネジメント PC 101 のキーボードやマウス等からなる入力装置から行われ、これら一連の動作はデータ処理装置 115 内の制御部で制御される。

## 【 0 0 4 6 】

一方、操作入力の状態表示及び処理中の画像データの表示は、MFP 100 の表示装置 116 又は、マネジメント PC 101、クライアント PC 102 のモニタ等で行われる。尚、記憶装置 111 は、マネジメント PC 101 から制御され、MFP 100 とマネジメント PC 101 とのデータの授受及び制御は、ネットワーク IF 117 及び直結した LAN 109 を用いて行われる。

## 【 0 0 4 7 】

## 〔処理概要〕

次に、本発明の一実施形態に係る画像処理システムによる画像処理全体の概要について

10

20

30

40

50

説明する。図3は、本発明の第1の実施例に係る画像処理システムによる画像処理の手順について説明するためのフローチャートである。

【0048】

ユーザは、MFP100の操作部(UI)等を用いて各種設定等の入力操作を行う(ステップS120)。尚、ステップS120の入力操作の詳細については後述する。

【0049】

次に、MFP100における画像読み取り部110を動作させて1枚の原稿をラスト走査し、例えば、600dpi、8ビットの画像信号を得る(イメージ情報入力処理:ステップS121)。尚、当該画像信号は、データ処理装置115で前処理を施して記憶装置111に1ページ分の画像データとして保存する。

10

【0050】

次に、マネージメントPC101のCPUにより、記憶装置111に格納された画像信号から、まず文字/線画部分とハーフトーンの画像部分とに領域を分離する。そして、文字部分はさらに段落で塊として纏まっているブロック毎に、或いは、線で構成された表、図形に分離し各々セグメント化する。一方、ハーフトーンで表現される画像部分は、矩形に分離されたブロックの画像部分、背景部等のいわゆるブロック毎に独立したオブジェクトに分割する(BS処理:ステップS122)。

【0051】

このとき、原稿画像中に付加情報として記録された2次元バーコード、或いはURLに該当するオブジェクトを検出して、URLはOCR処理(光学的文字認識処理)で文字認識し、2次元バーコード又は当該オブジェクトを解読する(ステップS123)。

20

【0052】

次いで、原稿のオリジナル電子ファイルが格納されている記憶装置内のポインタ情報を検出する(ステップS124)。尚、ポインタ情報を付加する手段としては、他に文字と文字の間隔に情報を埋め込む方法や、ハーフトーンの画像に埋め込む方法等の直接可視化されない電子透かしによる方法を用いた場合であってもよい。このように付加情報が電子透かしとして埋め込まれている場合は、ステップS125では透かし情報を検出して解読することとなる。

【0053】

そして、ポインタ情報が検出されたか否かを判定し(ステップS125)、ポインタ情報が検出された場合(Yes)は、ポインタで示されたアドレスから原稿に対応する電子ファイルを検索することによって特定する(ステップS126)。電子ファイルは、図1においてクライアントPC102内のハードディスク内、或いはオフィス10、20のLAN107、108に接続された文書管理サーバ106a、106b内のデータベース105a、105b、或いはMFP100自体が有する記憶装置111のいずれかに格納されている。そして、ステップS125で得られたアドレス情報に従って、これらの記憶装置内から検索される。

30

【0054】

その結果、ステップS126で電子ファイルが見つからなかった場合(No)、又は、見つかったがJPEGやtiffに代表されるようなイメージファイルであった場合(すなわち、既存の文書作成ソフトウェア等では再利用・再編集が困難なイメージファイルの場合)は、ステップS127に分岐する。また、ステップS125で、ポインタ情報自体が存在しなかった場合(No)も、ステップS127に分岐する。

40

【0055】

ステップS127は、いわゆる文書検索処理ルーチンである。まず、ステップS122で各テキストブロックに対して行ったOCRの結果から単語を抽出して、全文検索或いは各オブジェクトの配列と各オブジェクトの属性からいわゆるレイアウト検索を行う。検索の結果、類似度の高い電子ファイルが見つかった場合はそのサムネイル等を表示する(ステップS128)。また、検索された電子ファイルが複数存在する場合等の操作者の選択が必要な場合は、操作者の入力操作によってファイルの特定を行う。尚、候補の電子ファイ

50



ルが1つしかない場合、自動的にステップS 1 2 9からステップS 1 3 4に分岐して、格納アドレスを通知する。

【0056】

また、ステップS 1 2 7の検索処理で電子ファイルが見つからなかった場合、或いは、見つかったがJ P E Gやt i f fに代表されるイメージファイルであった場合はステップS 1 3 0に分岐してベクトル化処理を行う。

【0057】

ステップS 1 3 0では、ラスタイメージデータからベクトルデータへの変換処理を行って、オリジナルの電子ファイルに近い、編集容易で容量の小さい電子ファイルに変換する。すなわち、ステップS 1 2 2でO C R処理された文字ブロックに対して、文字種、文字のサイズ、スタイル及び字体等を認識し、原稿を走査して得られた文字に可視的に忠実なフォントデータに変換し、文字コード化する(文字コード、サイズ、スタイル、字体を示すコードとフォントデータとを対応付ける。)。一方、線や曲線で構成される線画、表、又は図形ブロック等に対してはアウトライン化、関数近似化(直線近似、曲線近似、図形近似等)を行う。さらに、画像ブロックに対しては、イメージデータを圧縮して個別のJ P E Gファイル等として処理する。

【0058】

尚、ここでは文字コード化、イメージデータの圧縮も含めてベクトルデータへの変換処理として説明したが、原稿によっては当然にベクトル化されるデータが含まれない(例えば、写真のみの原稿や文字のみの原稿)があるが、ここではこのように画像の属性(文字、線画、自然画等)に応じた形式に変換する処理をベクトル化処理と称することとする。

【0059】

尚、これらのベクトル化処理は、各オブジェクト毎に行い、更に各オブジェクトのレイアウト情報を保存して、例えば、r t fやs v g等のアプリデータに変換し(ステップS 1 3 1)、変換後のファイルを電子ファイルとして記憶装置1 1 1等に格納する(ステップS 1 3 2)。

【0060】

ベクトル化された原稿画像は、以降同様の処理を行う際に直接電子ファイルとして検索することができるように、検索のためのインデックス情報を生成して(ステップS 1 3 3)、検索用インデックスファイルに追加する。そして、格納アドレスを操作者(クライアント)に通知する(ステップS 1 3 4)。

【0061】

さらに、操作者が行おうしている処理が記録であるか否かが判断され(ステップS 1 3 5)、記録の場合(Y e s)はステップS 1 3 7に分岐して、ポインタ情報をイメージデータとしてファイルに付加する。

【0062】

一方、検索処理で電子ファイルが特定できた場合も同様に、ステップS 1 2 9からステップS 1 3 4に分岐し、格納アドレスをクライアントに通知するとともに(ステップS 1 3 4)、用紙等の媒体上に記録する場合は同様にポインタ情報を電子ファイルに付加する(ステップS 1 3 6)。

【0063】

尚、以上本実施例によって得られた電子ファイル自体を用いて、例えば文書の加工、画像の編集、蓄積、伝送、記録をステップS 1 3 7で行うことが容易になる。

【0064】

上述したような処理によって、イメージデータをそのままの形で取得するような場合に比べて、取得後の当該電子ファイルの再利用・再編集が容易になるだけでなく、情報量を削減することにより蓄積効率が高まり、伝送時間が短縮され、記録表示する際には高品位なデータとして非常に優位となるという効果が得られる。

【0065】

以下各処理ブロックに対して詳細に説明する。

10

20

30

40

50

## 【 0 0 6 6 】

## 〔 入力操作 〕

まず、ステップ S 1 2 0 で行われる入力操作について説明する。図 2 0 は、ステップ S 1 2 0 の入力操作の手順を詳細に説明するためのフローチャートである。ユーザは、M F P 1 0 0 の入力装置 1 1 3 等の操作部 ( U I ) を用いて、原稿に付加されているポインタ情報より電子ファイルを取得するための検索処理 ( 特定処理 ) を行うか否か ( O N / O F F ) を設定する ( ステップ S 2 0 0 1 ) 。これは、ポインタ情報が示すオリジナルの電子ファイルが、検索対象データベース ( ファイルサーバ ) に存在しないことが予め判っているときなどの用途で用いられる。或いは、ポインタ情報が示す電子ファイルが存在すると判っていても、その電子ファイルに拘らないという場合には、検索処理を O F F に設定することが処理時間上からも望ましい。従って、本実施例では、特定処理の実行の可否を選択させることとしている。

10

## 【 0 0 6 7 】

次に、読み取り原稿の属性を示すモード ( 例えば、文字、写真、線画、表等 ) の設定を行う。ステップ S 2 0 0 2 で設定される属性は、複数の選択を可能とし、例えば、文字原稿に対しては、「文字 ( Text ) 」の属性を選択する。従って、ステップ S 1 2 3 のブロックセレクション処理の結果は、Text の属性に限定される。

## 【 0 0 6 8 】

また、文字と線画が混在するような原稿に対しては、「文字」と「線画 ( Line ) 」の属性を選択する。従って、ステップ S 1 2 2 のブロックセレクション処理の結果は、Text が Line の 2 種類の属性に限定される。

20

## 【 0 0 6 9 】

これらは、ユーザが読み取り原稿を見て判断しなければならないが、ユーザが属性を指定することにより、例えば、全面が写真である原稿に対しては、ブロックセレクション処理や O C R 処理、又は、関数近似処理といった処理においては、指定された属性に基づいて結果を得ることができるので、実質的な処理に行わずに済むので、処理の高速化及び高画質化を図ることができる。

## 【 0 0 7 0 】

また、もちろん原稿の内容に直接従う必要はなく、種々の種類のオブジェクトを含む原稿のうち必要なオブジェクトのみ属性に従ったベクトル化 ( 例えば文字のみ文字コード化するようにして他は線画であったとしてもイメージのままとするなど ) してもよい。

30

## 【 0 0 7 1 】

次に、所望とするその他処理 ( O C R 処理、フォント化処理、フォント指定処理、関数近似処理、圧縮処理のための圧縮率の設定処理等 ) の設定を行う ( ステップ S 2 0 0 3 ) 。

## 【 0 0 7 2 】

すなわち、O C R 処理に関する設定は、O C R 処理を行うか否かを指定し、O C R 処理を行うと指定された場合は、ブロックセレクション処理の結果で Text と判定されたブロックに対して O C R 処理を行う。また、フォント化処理に関する設定は、O C R 処理結果を利用して、原稿中のフォントに近いフォントデータに置き換える処理を行うか否かを指定する。さらに、フォント指定処理の設定は、原稿中のフォントに関係なくユーザの所望とするフォントに置き換える処理を行うか否かを指定する。さらにまた、関数近似処理のための設定は、ブロックセレクション処理の結果で、Text や Line や Table ( 表 ) と判定されたブロックに対して、関数近似 ( 直線近似、曲線近似、図形近似等 ) を行うか否かを設定する。さらにまた、圧縮率の設定は、ブロックセレクション処理の結果で、Photo と判定されたブロックに対して J P E G 圧縮等を施す場合の圧縮率の設定を行う。

40

## 【 0 0 7 3 】

尚、属性の設定及び所望とするその他処理の設定は、上述したものに限定されることなく、その他の設定を同様に行ってもよい。そして、以上のパラメータ等を設定した後、入力操作処理 ( ステップ S 1 2 0 ) を終了する。

50

## 【 0 0 7 4 】

次に、ステップ S 1 2 2 で示すブロックセレクション ( B S ) 処理について説明する。

## 【 0 0 7 5 】

## 〔 ブロックセレクション処理 〕

図 4 は、ブロックセレクション処理によって読み取った 1 枚のイメージデータを属性を判定し複数のブロックに分割する様子を示す図である。すなわち、ブロックセレクション処理とは、符号 4 1 に示すステップ S 1 2 1 で読み取った一頁のイメージデータを、符号 4 2 に示すようにオブジェクト毎の塊として認識し、それぞれのブロックを文字 ( Text ) 、写真 ( Photo ) 、線 ( Line ) 、表 ( Table ) 等の属性に判定し、異なる属性を持つ領域 ( ブロック ) に分割する処理である。

10

## 【 0 0 7 6 】

ブロックセレクション処理の一実施例を以下に説明する。

## 【 0 0 7 7 】

まず、入力画像を白黒に 2 値化して、輪郭線追跡を行って黒画素輪郭で囲まれる画素の塊を抽出する。そして、面積の大きい黒画素の塊については、内部にある白画素に対して同様に輪郭線追跡を行って白画素の塊を抽出する。さらに、一定面積以上の白画素の塊の内部からも再帰的に黒画素の塊を抽出する。尚、上記処理は、白地に黒字等で記載されている原稿の場合の処理であって、それ以外の場合は背景に相当する色を「白」、オブジェクトに相当する色を「黒」とすることにより同様に処理することができる。

## 【 0 0 7 8 】

このようにして得られた黒画素の塊を、大きさ及び形状等で分類し、異なる属性を持つ領域へ分類する。例えば、縦横比が 1 に近く、大きさが一定の範囲のものを文字相当の画素塊とし、さらに近接する文字が整列良くグループ化可能な部分を文字領域とする。また、扁平な画素塊を線領域、一定の大きさ以上でかつ四角系の白画素塊を整列よく内包する黒画素塊の占める範囲を表領域、不定形の画素塊が散在している領域を写真領域、それ以外の任意形状の画素塊を図画領域等とする。

20

## 【 0 0 7 9 】

図 5 は、ブロックセレクション処理で得られた各ブロックに対するブロック情報の一例について示す図である。図 5 に示されるブロック毎の情報は、後述するベクトル化、検索処理のための情報として用いられる。

30

## 【 0 0 8 0 】

## 〔 イメージデータからポインタ情報の検出 〕

まず、ステップ S 1 2 3 で示す電子ファイルの格納位置を、読み取られたイメージデータから抽出するための O C R / O M R 処理について説明する。

## 【 0 0 8 1 】

図 6 は、原稿画像中に付加された 2 次元バーコード ( Q R コードシンボル ) を復号してデータ文字列を出力する手順を説明するためのフローチャートである。図 7 は、2 次元バーコードが付加された原稿 3 1 0 の一例を示す図である。

## 【 0 0 8 2 】

まず、データ処理装置 1 1 5 内のページメモリに格納された原稿 3 1 0 を読み取って得られたイメージデータを内部の C P U で走査して、前述したブロックセレクション処理の結果から所定の 2 次元バーコードシンボル 3 1 1 の位置を検出する。Q R コードの位置検出パターンは、シンボルの 4 隅のうちの 3 隅に配置される同一の位置検出要素パターン 3 1 1 a ~ 3 1 1 c から構成される ( ステップ S 3 0 0 ) 。

40

## 【 0 0 8 3 】

次に、位置検出パターンに隣接する形式情報を復元し、シンボルに適用されている誤り訂正レベル及びマスクパターンを得る ( ステップ S 3 0 1 ) 。さらに、シンボルの型番を決定した後 ( ステップ S 3 0 2 ) 、形式情報で得られたマスクパターンを使って符号化領域ビットパターンを X O R 演算することによってマスク処理を解除する ( ステップ S 3 0 3 ) 。

50

## 【 0 0 8 4 】

そして、モデルに対応する配置規則に従い、シンボルキャラクタを読み取り、メッセージのデータ及び誤り訂正コード語を復元する（ステップ S 3 0 4）。次いで、復元されたコード上に誤りがあるかどうかの検出を行う（ステップ S 3 0 5）。その結果、誤りが検出された場合（Yes）は当該誤りを訂正する（ステップ S 3 0 6）。そして、誤り訂正されたデータより、モード指示子及び文字数指示子に基づいて、データコード語をセグメントに分割する（ステップ S 3 0 7）。最後に、仕様モードに基づいてデータ文字を復号し、結果を出力する（ステップ S 3 0 8）。尚、ステップ S 3 0 5 で誤りが検出されなかった場合（No）は、上記ステップ S 3 0 7 に進む。

## 【 0 0 8 5 】

ここで、2次元バーコード内に組み込まれたデータは、対応する電子ファイルが格納されているサーバアドレス情報（ポインタ情報）を表しており、例えばファイルサーバ名及びサーバアドレスを示すIPアドレス、対応するURL等からなるパス情報で構成される。

## 【 0 0 8 6 】

本実施例では上述したように、ポインタ情報が2次元バーコードを用いて付与された原稿 3 1 0 を例に挙げて説明したが、直接文字列でポインタ情報が記録される場合には、所定のルールに従った文字列のブロックを前述したブロックセレクション処理で検出し、当該ポインタ情報を示す文字列の各文字を文字認識することで、直接、オリジナルの電子ファイルが保存されているサーバのサーバアドレス情報を得ることが可能である。

## 【 0 0 8 7 】

また、図 7 の文書 3 1 0 の文字ブロック 3 1 2 や文字ブロック 3 1 3 の文字列に対して、隣接する文字と文字の間隔等に視認し難い程度の変調を加え、当該文字間隔を用いた透かし情報を埋め込むことでもポインタ情報を付与することができる。このような透かし情報は、後述する文字認識処理を行う際に各文字の間隔を検出することによって、ポインタ情報を得ることができる。また、自然画ブロック 3 1 4 の中に電子透かしとしてポインタ情報を付加することも可能である。

## 【 0 0 8 8 】

## [ ポインタ情報によるファイル検索 ]

次に、図 3 を用いて説明したステップ S 1 2 6、S 1 2 9 の処理で行われるポインタ情報からの電子ファイルが格納されているサーバの検索処理について詳細に説明する。図 8 は、検出されたポインタ情報から電子ファイルが格納されているサーバを検索する処理手順について説明するためのフローチャートである。

## 【 0 0 8 9 】

まず、ポインタ情報に含まれるサーバアドレスに基づいて、当該電子ファイルが格納されているファイルサーバを特定する（ステップ S 4 0 0）。ここでファイルサーバとは、クライアント PC 1 0 2 や、データベース 1 0 5 a、b を内蔵する文書管理サーバ 1 0 6 a、b や、記憶装置 1 1 1 を内蔵する MFP 1 0 0 自身を指す。また、アドレスとは、URL やサーバ名からなるパス情報である。

## 【 0 0 9 0 】

そして、ファイルサーバを特定した後、図 3 を用いて説明したステップ S 1 2 7 におけるポインタ情報が示すサーバ（ファイルサーバ）内を検索するための準備をファイルサーバに対して要求する（ステップ S 4 0 1）。ファイルサーバは、ステップ S 1 2 8 のファイル検索処理に従って、該当する電子ファイルを検索する（ステップ S 4 0 2）。そして、電子ファイルが存在するか否かを判定する（ステップ S 4 0 3）。

## 【 0 0 9 1 】

この結果、電子ファイルが存在しない場合（No）は、MFP 1 0 0 に対してその旨を通知して終了する。一方、電子ファイルが存在する場合（Yes）は、図 3 を用いて前述したように、ステップ S 1 2 9 からステップ S 1 3 3 までの処理を行うために候補表示を行い、その後、該当する電子ファイルのアドレスを通知すると共に、当該電子ファイルを

10

20

30

40

50

ユーザ（すなわち、MFP100）に対して転送する（ステップS408）。

【0092】

〔ファイル検索処理〕

次に、図3のステップS127で示すファイル検索処理の詳細について図5及び図10を使用して説明する。ステップS127の処理は、前述したように、ステップS125で入力原稿（入力ファイル）にポインタ情報が存在しなかった場合、ポインタ情報は存在するが電子ファイルが見つからなかった場合、或いは電子ファイルがイメージファイルであった場合に行われる。

【0093】

ここでは、ステップS123のOCR/OMR処理の結果、抽出された各ブロック及び入力ファイルが、図5に示す情報（ブロック情報、入力ファイル情報）を備えるものとする。本実施形態では、情報内容として、図5に示すように属性、座標位置、幅及び高さのサイズ、OCR情報の有無を用いる。

【0094】

属性は、さらに、文字、線、写真、絵、表等に分類される。尚、図5では説明を簡単にするため、ブロックは座標Xの小さい順（例えば、 $X_1 < X_2 < X_3 < X_4 < X_5 < X_6$ ）に、ブロック1、ブロック2、ブロック3、ブロック4、ブロック5、ブロック6としている。また、ブロック総数は、入力ファイル中の全ブロック数であり、図5に示す例におけるブロック総数は6である。以下、これらの情報を使用して、データベース内から入力イメージファイルに類似した電子ファイルのレイアウト検索を行う手順について説明する。図10は、データベース内から入力イメージファイルに類似した電子ファイルのレイアウト検索を行う手順について説明するためのフローチャートである。尚、データベースファイルも、図5に示す情報と同様の情報を備えることを前提とする。図10のフローチャートの流れは、入力された原稿から読み取られた電子ファイルとデータベース中の電子ファイルを順次比較するものである。

【0095】

まず、後述する類似率等の初期化を行って初期値を設定する（ステップS510）。次に、ブロック総数の比較を行い（ステップS511）、真の場合（Yes）は、さらにファイル内のブロックの情報を順次比較する（ステップS512）。すなわち、データベースのファイルのブロック数nが入力ファイルのブロック数Nの誤差N範囲内かどうかを調べ、誤差範囲内であれば真（Yes）、範囲外であれば偽（No）とする。また、ステップS512では、入力ファイルとデータベースファイルのブロック属性を比較して一致すればステップS513以降の比較処理へ進み、不一致であればステップS521に進む。

【0096】

ブロックの情報比較では、ステップS513、S515、S518において、それぞれ属性類似率、サイズ類似率、OCR類似率をそれぞれ算出し、ステップS522においてそれらに基づいて総合類似率を算出する。各類似率の算出方法については、公知の技術が用いられるので説明を省略する。

【0097】

ステップS523においては、総合類似率が予め設定された閾値Thより高いかどうかを判定し、高い場合（Yes）は、その電子ファイルを類似候補として挙げて保存する（ステップS524）。尚、図中のN、W、Hは、入力ファイルのブロック総数、各ブロック幅、各ブロック高さ、N、W、Hは、入力ファイルのブロック情報を基準として誤差を考慮したものである。また、n、w、hは、データベースファイルのブロック総数、各ブロック幅、各ブロック高さとする。尚、ステップS514におけるサイズ比較時に、位置情報（X、Y）の比較等を行ってもよい。

【0098】

以上、検索の結果、総合類似率が閾値Thより高いもので候補として保存されたデータベースファイルをサムネール等に表示する（ステップS128）。これにより、複数の中

10

20

30

40

50

から操作者の選択が必要な場合は、操作者の入力操作によってファイルの特定を行う。

【 0 0 9 9 】

[ ベクトル化処理 ]

ファイルサーバにオリジナルの電子ファイルが存在しない場合、又は、前述したようにステップ S 1 2 0 の入力操作において検索処理の O F F を選択した場合は、図 4 に示すイメージデータを各ブロック毎にベクトル化する。以下では、図 3 のベクトル化処理（ステップ S 1 3 0 ）について詳細に説明する。

【 0 1 0 0 】

《文字認識》

前述したように、ステップ S 1 2 0 の入力操作によって O C R 処理を行うと指定した場合、文字ブロックに対しては各文字に対して文字認識処理が行われる。本実施例の文字認識処理では、文字単位で切り出された画像に対し、パターンマッチングの一手法を用いて認識を行い、対応する文字コードを得る。この認識処理は、文字画像から得られる特徴を数十次元の数値列に変換した観測特徴ベクトルと、あらかじめ字種毎に求められている辞書特徴ベクトルと比較し、最も距離の近い字種を認識結果とする処理である。尚、特徴ベクトルの抽出には種々の公知手法があり、例えば、文字をメッシュ状に分割し、各メッシュ内の文字線を方向別に線素としてカウントしたメッシュ次元ベクトルを特徴とすることができる。

10

【 0 1 0 1 】

ブロックセレクション処理（ステップ S 1 2 2 ）で抽出された文字領域に対して文字認識を行う場合、まず該当領域に対して横書き、縦書きの判定を行い、各々対応する方向に行を切り出し、その後文字を切り出して文字画像を得る。横書き、縦書きの判定は、該当領域内で画素値に対する水平 / 垂直の射影を取り、水平射影の分散が大きい場合は横書き領域、垂直射影の分散が大きい場合は縦書き領域と判断すればよい。

20

【 0 1 0 2 】

また、文字列及び文字への分解は、横書きの場合は水平方向の射影を利用して行を切り出し、さらに切り出された行に対する垂直方向の射影から、文字を切り出す。一方、縦書きの文字領域に対しては、水平と垂直を逆にすればよい。尚、文字のサイズは切り出した大きさに基づいて検出することができる。

【 0 1 0 3 】

《フォント認識》

文字認識の際に用いられる字種数分の辞書特徴ベクトルを、文字形状種、すなわちフォント種に対して複数用意し、マッチングの際に文字コードとともにフォント種を出力することで、文字のフォントを認識することができる。

30

【 0 1 0 4 】

《文字のベクトル化》

前述した文字認識及びフォント認識によって得られた、文字コード及びフォント情報を用いて、アウトライン化処理によるアウトラインデータを用いて、文字部分の情報をベクトルデータに変換する。尚、元原稿がカラーの場合は、カラー画像から各文字の色を抽出してベクトルデータとともに記録する。以上の処理により、文字ブロックに属するイメージ情報をほぼ形状、大きさ、色が忠実なベクトルデータに変換することができる。

40

【 0 1 0 5 】

《文字以外の部分のベクトル化》

ステップ S 1 2 2 のブロックセレクション処理で、線画或いは線、表領域とされた領域を対象として、それぞれ抽出された画素塊の輪郭をベクトルデータに変換する。具体的には、輪郭を成す画素の点列を角とみなされる点で区切って、各区間を部分的な直線或いは曲線で近似する。ここで、「角」とは、曲率が極大となる点である。

【 0 1 0 6 】

図 1 1 は、曲率が極大となる点を説明するための図である。図 1 1 に示すように、任意点  $P_i$  に対して左右  $k$  個の離れた点  $P_{i-k} \sim P_{i+k}$  の間に弦を引いたとき、この弦と  $P_i$  の距

50

離が極大となる点として求められる。さらに、 $P_{i-k} \sim P_{i+k}$ 間の弦の長さ/弧の長さを  $R$  とし、 $R$ の値が閾値以下である点を角とみなすことができる。角によって分割された後の各区間は、直線は点列に対する最小二乗法等を用いて、曲線は3次スプライン関数等を用いてベクトル化することができる。

【0107】

また、対象が内輪郭を持つ場合、ブロックセレクション処理で抽出した白画素輪郭の点列を用いて、同様に部分的直線或いは曲線で近似する。

【0108】

以上のように、輪郭の区分線近似を用いることによって、任意形状の図形のアウトラインをベクトル化することができる。尚、入力される原稿がカラーの場合は、カラー画像から図形の色を抽出してベクトルデータとともに記録する。

10

【0109】

図12は、外輪郭が内輪郭又は別の外輪郭と近接している場合に太さを持った線として表現する例について説明するための図である。図12に示すように、ある区間で外輪郭が、内輪郭又は別の外輪郭が近接している場合、2つの輪郭線を一まとめにし、太さを持った線として表現することができる。具体的には、ある輪郭の各点  $P_i$  から別輪郭上で最短距離となる点  $Q_i$  まで線を引き、各距離  $PQ_i$  が平均的に一定長以下の場合、注目区間は  $PQ_i$  中点を点列として直線又は曲線で近似し、その太さは  $PQ_i$  の平均値とする。線や線の集合体である表罫線は、前記したような太さを持つ線の集合として、効率よくベクトル表現することができる。

20

【0110】

尚、文字ブロックに対する文字認識処理を用いたベクトル化については前述したように、当該文字認識処理の結果、辞書からの距離が最も近い文字を認識結果として用いる。ここで、この距離が所定値以上の場合、必ずしも本来の文字に一致するとは限らず、形状が類似する文字に誤認識するような場合が多い。従って、本発明では、このような文字に対しては上記したように、一般的な線画と同様に扱って当該文字をアウトライン化する。すなわち、従来は文字認識処理で誤認識を起こしていたような文字でも、誤った文字にベクトル化されることなく、可視的にイメージデータに忠実なアウトライン化によるベクトル化を行うことができる。また、写真と判定されたブロックに対しては、本発明ではベクトル化せずに、イメージデータのままとするか、ステップS120の入力操作で指定された圧縮率に基づいて圧縮する。

30

【0111】

《図形認識》

ここでは、上述したように任意形状の図形のアウトラインをベクトル化した後、これらのベクトル化された区分線を図形オブジェクト毎にグループ化する処理について説明する。

【0112】

図13は、ベクトルデータを図形オブジェクト毎にグループ化するまでの処理手順を説明するためのフローチャートである。まず、各ベクトルデータの始点、終点を算出する(ステップS700)。次に、各ベクトルの始点、終点情報を用いて、図形要素を検出する(ステップS701)。ここで、図形要素の検出とは、区分線が構成している閉図形を検出することである。検出に際しては、閉形状を構成する各ベクトルはその両端にそれぞれ連結するベクトルを有しているという原理を応用して検出を行う。

40

【0113】

次に、図形要素内に存在する他の図形要素又は区分線をグループ化し、一つの図形オブジェクトとする(ステップS702)。尚、図形要素内に他の図形要素又は区分線が存在しない場合は、図形要素を図形オブジェクトとする。

【0114】

図14は、図形要素を検出する処理手順を説明するためのフローチャートである。まず、ベクトルデータから両端に連結していない不要なベクトルを除去し、閉図形構成ベクトル

50

ルを抽出する（ステップS710）。次に、閉図形構成ベクトルの中から当該ベクトルの始点を開始点とし、時計回りに順にベクトルを追っていく。そして、開始点に戻るまで追跡を行い、通過したベクトルを全て一つの図形要素を構成する閉図形としてグループ化する（ステップS711）。尚、この際に、閉図形内部にある閉図形構成ベクトルも全てグループ化する。さらに、まだグループ化されていないベクトルの始点を開始点とし、同様の処理を繰り返す。最後に、ステップS710で除去された不要ベクトルのうち、ステップS711で閉図形としてグループ化されたベクトルに接合しているものを検出し、一つの図形要素としてグループ化する（ステップS712）。

#### 【0115】

以上の処理によって、図形ブロックを個別に再利用可能な個別の図形オブジェクトとして扱うことが可能になる。

10

#### 【0116】

ここで、一例として、svgフォーマットの記述方法を用いて直線近似、曲線近似、及び図形近似を行った場合の例を図21～図23を用いて説明する。図21は、svgフォーマットの記述方法を用いて直線近似を行った場合の一例を示す図である。図21において、ラスタイメージ2101に対してアウトライン化した座標列データが2103で示される。これらの座標列2103に対し、直線近似を行うと、2104で示すように、わずかなデータで表現することが可能となる。また、直線として認識されているため、ビューアでの表示も高画質である。

#### 【0117】

20

同様に、図22は、svgフォーマットの記述方法を用いて曲線近似を行った場合の一例を示す図である。図22において、2203は曲線のラスタイメージ2201のアウトラインデータ（アウトライン化した座標列データ）、2204は曲線近似データを示す。さらに、図23は、svgフォーマットの記述方法を用いて図形近似を行った場合の一例を示す図である。図23において、2303は図形（例えば、円）のラスタイメージ2301のアウトラインデータ、2304は図形近似データである。

#### 【0118】

以上、説明したように、各ベクトル化処理（関数近似処理）は、ステップS120の入力操作処理による属性設定、又は、ステップS122のブロックセレクション処理の結果に応じて、切り替えることが可能となる。ここで、前述したように、ステップS120の入力操作では、関数近似を行うか否かが指定可能である。そこで、関数近似を行うと指定した場合の各属性ごとの関数近似処理の切り替え例を下記に示す。下記のように、各属性に応じて関数近似方法を切り替えることにより、処理の高速化及び高画質化、小容量化を図ることができる。

30

- ・文字属性の場合は、直線近似と曲線近似を行う。
- ・写真属性の場合は、関数近似は行わない。
- ・表属性の場合は、直線近似を行う。
- ・線画属性の場合は、直線近似と曲線近似と図形近似を行う。

#### 【0119】

[ アプリデータへの変換処理 ]

40

図15は、一頁分のイメージデータをブロックセレクション処理（ステップS123）及びベクトル化処理（ステップS131）によって変換された結果として得られる中間データ形式のファイルのデータ構造を示す図である。図15に示すようなデータ形式は、ドキュメント・アナリシス・アウトプット・フォーマット（DAOF）と呼ばれる。すなわち、図15は、DAOFのデータ構造を示す図である。

#### 【0120】

図15において、791はHeader（ヘッダ）であり、処理対象の文書画像データに関する情報が保持される。792はレイアウト記述データ部であり、文書画像データ中のTEXT（文字）、TITLE（タイトル）、CAPTION（キャプション）、LINEART（線画）、PICTURE（自然画）、FRAME（枠）、TABLE（表）等の属性毎に認識された各ブロックの属性情報とそ

50



の矩形アドレス情報を保持する。

【 0 1 2 1 】

7 9 3 は文字認識記述データ部であり、TEXT、TITLE、CAPTION等のTEXTブロックを文字認識して得られる文字認識結果を保持する。7 9 4 は表記述データ部であり、TABLEブロックの構造の詳細を格納する。7 9 5 は画像記述データ部であり、PICTUREやLINEART等のブロックのイメージデータを文書画像データから切り出して保持する。

【 0 1 2 2 】

このようなD A O Fは、中間データとしてのみならず、それ自体がファイル化されて保存される場合もあるが、このファイルの状態では、一般の文書作成アプリケーションで個々のオブジェクトを再利用・再編集等することはできない。そこで、次に、D A O Fからアプリデータに変換する処理（ステップS 1 3 2）について詳説する。

【 0 1 2 3 】

図1 6は、アプリデータへの変換処理全体の概略手順を説明するためのフローチャートである。まず、D A O Fデータを入力する（ステップS 8 0 0）。次いで、アプリデータの元となる文書構造ツリー生成を行う（ステップS 8 0 2）。そして、生成した文書構造ツリーに基づいて、D A O F内の実データを流し込み、実際のアプリデータを生成する（ステップS 8 0 4）。

【 0 1 2 4 】

図1 7は、文書構造ツリー生成処理（ステップS 8 0 2）の詳細な処理手順を説明するためのフローチャートである。また、図1 8は、文書構造ツリーの概要を説明するための図である。尚、全体制御の基本ルールとして、処理の流れはマイクロブロック（単一ブロック）からマクロブロック（ブロックの集合体）へ移行するものとする。また、以後の説明では、ブロックとは、マイクロブロック及びマクロブロック全体を指す。

【 0 1 2 5 】

まず、ブロック単位で縦方向の関連性を元に再グループ化する（ステップS 8 0 2 a）。尚、スタート直後はマイクロブロック単位での判定となる。ここで、関連性とは、距離が近く、ブロック幅（横方向の場合は高さ）がほぼ同一であること等で定義することができる。また、距離、幅、高さ等の情報はD A O Fを参照して抽出する。

【 0 1 2 6 】

図1 8において、（ a ）は実際のページ構成、（ b ）はその文書構造ツリーを示している。ステップS 8 0 2 aのグループ化の結果、T 3、T 4、T 5が一つのグループV 1として、T 6、T 7が一つのグループV 2として、それぞれ同じ階層のグループとして生成される。

【 0 1 2 7 】

次に、縦方向のセパレータの有無をチェックする（ステップS 8 0 2 b）。セパレータは、例えば、物理的にはD A O F中でライン属性を持つオブジェクトである。また、論理的な意味としては、アプリ中で明示的にブロックを分割する要素である。ここでセパレータを検出した場合は、同じ階層で再分割する。

【 0 1 2 8 】

次いで、分割がこれ以上存在し得ないか否かをグループ長を利用して判定する（ステップS 8 0 2 c）。例えば、縦方向のグループ長がページ高さか否かを判定する。その結果、縦方向のグループ長がページ高さとなっている場合は（ Y e s ）、文書構造ツリー生成は終了する。例えば、図1 8に示すような構造の場合は、セパレータもなく、グループ高さはページ高さではないので、N oと判定され、ステップS 8 0 2 dに進む。

【 0 1 2 9 】

ステップS 8 0 2 dでは、ブロック単位で横方向の関連性を元に再グループ化する。但し、この再グループ化においてもスタート直後の第一回目は、マイクロブロック単位で判定を行うことになる。また、関連性及びその判定情報の定義は、縦方向の場合と同じである。例えば、図1 8の構造の場合は、T 1とT 2でH 1、V 1とV 2でH 2が生成され、H 1はT 1、T 2の一つ上、H 2はV 1、V 2の一つ上の同じ階層のグループとして生成さ

10

20

30

40

50

れる。

#### 【0130】

次いで、横方向セパレータの有無をチェックする(ステップS802e)。図18では、S1があるので、これをツリーに登録し、H1、S1、H2という階層が生成される。そして、分割がこれ以上存在し得ないか否かをグループ長を利用して判定する(ステップS802f)。例えば、横方向のグルーピング長がページ幅か否かを判定する。その結果、横方向のグループ長がページ幅となっている場合(Yes)、文書構造ツリー生成は終了する。一方、ページ幅となっていない場合(No)は、ステップS802bに戻り、再度もう一段上の階層で、縦方向の関連性チェックから繰り返す。例えば、図18の構造の場合は、分割幅がページ幅になっているので、ここで終了し、最後にページ全体を表す最上位階層のV0が文書構造ツリーに付加される。

10

#### 【0131】

文書構造ツリーが完成した後、その情報に基づいて、ステップS804においてアプリデータの生成を行う。図18の構造の場合は、具体的に以下になる。

#### 【0132】

すなわち、H1は横方向に2つのブロックT1、T2があるので、2カラムとし、T1の内部情報(DAOFを参照した文字認識結果の文章や画像等)を出力後、カラムを変えて、T2の内部情報出力し、その後S1を出力する。また、H2は横方向に2つのブロックV1、V2があるので、2カラムとして出力し、V1はT3、T4、T5の順にその内部情報を出力し、その後カラムを変えて、V2のT6、T7の内部情報を出力する。以上により、アプリデータへの変換処理を行うことができる。

20

#### 【0133】

##### [ポインタ情報の付加]

次に、ステップS136で示されるポインタ情報付加処理について詳細に説明する。処理すべき文書が検索処理で特定された場合、或いはベクトル化によって元ファイルが再生できた場合において、該文書を記録処理する場合においては、紙への記録の際にポインタ情報を付与することで、この文書を用いて再度各種処理を行う場合に簡単に元ファイルデータを取得できる。

#### 【0134】

図19は、ポインタ情報としてのデータ文字列を2次元バーコード(QRコードシンボル: JIS X0510)311を用いて符号化して画像中に付加する手順を説明するためのフローチャートである。

30

#### 【0135】

2次元バーコード内に組み込むデータは、対応するファイルが格納されるサーバーアドレス情報を表しており、例えばファイルサーバ名からなるパス情報で構成される。或いは、対応するサーバのURLや、対応するファイルが格納されているデータベース105a、b内或いはMFP100自体が有する記憶装置111を管理するためのID等で構成される。

#### 【0136】

まず、符号化する種類の異なる文字を識別するため、入力データ列を分析する。また、誤り検出及び誤り訂正レベルを選択し、入力データが収容できる最小型番を選択する(ステップS900)。次に、入力データ列を所定のビット列に変換し、必要に応じてデータのモード(数字、英数字、8ビットバイト、漢字等)を表す指示子や、終端パターンを付加する。さらに、所定のビットコード語に変換することによってデータの符号化を行う(ステップS901)。

40

#### 【0137】

この時、誤り訂正を行うため、コード語列を型番及び誤り訂正レベルに応じて所定のブロック数に分割し、各ブロック毎に誤り訂正コード語を生成し、データコード語列の後に付加する(ステップS902)。さらに、ステップS902で得られた各ブロックのデータコード語を接続し、各ブロックの誤り訂正コード語、また必要に応じて剰余コード語を

50

接続して、メッセージの構築を行う（ステップS903）。

【0138】

次に、マトリクスに位置検出パターン、分離パターン、タイミングパターン及び位置合わせパターン等とともにコード語モジュールを配置する（ステップS904）。さらに、シンボルの符号化領域に対して最適なマスクパターンを選択して、マスク処理パターンをステップS904で得られたモジュールにXOR演算により変換する（ステップS905）。最後に、ステップS905で得られたモジュールに形式情報及び型番情報を生成して、2次元コードシンボルを完成させる（ステップS906）。

【0139】

上述したサーバアドレス情報の組み込まれた2次元バーコードは、例えば、クライアントPC102から電子ファイルをプリントデータとして形成装置112で紙上に記録画像として形成する場合に、データ処理装置115内で記録可能なラスタデータに変換された後にラスタデータ上の所定の個所に付加されて画像形成される。ここで、画像形成された紙を配布されたユーザは、画像読み取り部110で読み取ることにより、前述したステップS124においてポインタ情報からオリジナルの電子ファイルが格納されているサーバの場所を適切に検出することができる。

【0140】

尚、同様の目的で付加情報を付与する手段は、本実施例で説明した2次元バーコードの他に、例えば、ポインタ情報を直接文字列で文書に付加する方法、文書内の文字列、特に文字と文字の間隔を変調して情報を埋め込む方法、文書中の中間調画像中に埋め込む方法等の一般に電子透かしと呼ばれる方法を適用してもよい。

【0141】

上述したように本実施例によれば、原稿を読み取ったイメージデータからユーザが所望している（クライアント側で指定された）アプリケーションフォーマットの電子ファイルを効率良く取得することができる。

【0142】

尚、原稿の属性は上述のものに限るものではなく、例えば地図モードなど種々の変更が可能であるものとする。

【0143】

いずれにしても、属性の指定や変換内容の指定に応じて属性判定のためのパラメータを変更したり、データ変換方法や実行させる変換処理を変更したりすることによって効率的な処理を行えるようにしたり、所望の形態のファイルを得ることができるようになるものである。

【実施例2】

【0144】

次に、ファイルアクセス権について考慮した実施例について説明する。通常扱われる文書ファイルの中には、第三者による再利用を制限することが望ましい文書がある。前述した実施例では、ファイルサーバに蓄積された電子ファイルは全て自由にアクセスすることが可能であり、ファイル全体、或いはその一部のオブジェクトは全て再利用が可能なることを前提に説明した。以下では、ポインタ情報から電子ファイルを検索した際に、検索の結果から特定された電子ファイルにアクセス権の制限が有る場合について説明する。

【0145】

図9は、ファイルアクセス権を含むポインタ情報から電子ファイルが格納されているサーバを検索する処理手順について説明するためのフローチャートである。図9において、ステップS400～S403までは、図8におけるステップと同様であるため、説明は省略する。ステップS403で電子ファイルが特定された場合（Yes）、ファイルサーバはそのファイルのアクセス権情報を調べる（ステップS404）。その結果、アクセス制限がある場合（Yes）は、MFPP100に対してパスワードの送信を要求する（ステップS405）。

【0146】

10

20

30

40

50

そして、MFP100は、操作者に対してパスワードの入力を促し、入力されたパスワードをファイルサーバに送信する(ステップS406)。ファイルサーバは送信されたパスワードを照合し(ステップS407)、一致した場合(Yes)は、図3を用いて説明したように、電子ファイルのアドレスを通知すると共に、ユーザの希望する処理がイメージデータの取得であれば、MFP100に対して電子ファイルを転送する(ステップS408)。

【0147】

尚、アクセス権の制御を行うための認証の方法は、ステップS405、S406に示したパスワードによる方法に限定されず、例えば、指紋認証等の一般に広く用いられている生体認証、カードによる認証等のあらゆる認証情報を用いた認証手段を用いることができる。

10

【0148】

また、本実施例では、紙文書に付加的に付与されたポインタ情報によりファイルを特定した場合の例について説明したが、図3のステップS127～S129で示す検索処理によってファイルを特定した場合においても同様の制御が可能である。

【0149】

一方、ステップS403でファイルサーバ内からファイルを特定することができなかった場合は、図3のステップS130で説明したベクトル化処理に対しても、制限を加えることができる。すなわち、紙文書を走査して得られたイメージデータからオリジナルの電子ファイルに対してのアクセス権の制限の存在を検出した場合には、認証確認が取れた場合のみベクトル化処理を行うことで、機密性の高い文書の使用に制限をかけることができる。

20

【実施例3】

【0150】

前述した実施例では、原稿画像を走査して得られるイメージ情報からサーバが保存するオリジナルのファイルデータを特定する手段としては、図3に示すように、文書中に付与されたポインタ情報に従って特定する手段か、文書中に記載された各オブジェクト情報に従って検索して特定する手段のいずれかであった。これに対し、より正確にオリジナルの電子ファイルを特定するには、上述した両手段を併用すれば良い。

【0151】

30

すなわち、原稿画像中から得られるポインタ情報から元ファイルの存在が検出できた場合であっても、さらに当該文書中のオブジェクト情報を使って、例えば、レイアウト情報に従うレイアウト検索や文字認識されたキーワードによる全文検索を検出されたファイルに対して行う。そして、高い一致度が得られた場合に当該検出された電子ファイルを、正式にオリジナルのファイルであると特定するようにする。このようにすることにより、例えば、ポインタ情報の下位の部分が曖昧であったり、誤り訂正でも訂正できなかった場合に対して、検索の範囲を絞り込んでファイルを特定することができるので、より高速で確実な電子ファイルの特定を行うことが可能となる。

【実施例4】

【0152】

40

前述した実施例では、検索処理において元ファイルの特定ができない場合、イメージ情報全体に対してベクトル化処理を行う。これに対し、例えば、一般の文書の場合、文書中のオブジェクト全て新規に作成された物でなく、一部のオブジェクトは他のファイルから流用して作成されるような場合がある。例えば、背景オブジェクト(壁紙)は文書作成アプリケーションで幾つかのパターンを予め用意してあり、その中から選択して用いるのが通常である。従って、このようなオブジェクトは、文書ファイルデータベースの中の他の文書ファイル中に存在している可能性が高く、又、再利用可能なベクトルデータとして存在する可能性が高い。

【0153】

そこで、図3のベクトル化処理(ステップS130)の別の実施例として、ブロックセ

50

レクション処理で個別のオブジェクトに分割された各オブジェクトに対して、それぞれのオブジェクト単位でデータベース中から一致するオブジェクトを一部を含むファイルを検索する。そして、一致したオブジェクトに対して、個別に当該ファイルからオブジェクト単位でベクトルデータを取得する。これにより、文書全体をベクトル化する必要がなくなり、より高速にベクトル化処理を行うことができ、さらにベクトル化処理が不必要な部分のベクトル化処理による画質劣化を防止することができる。

【0154】

一方、図3における検索処理(ステップS127)で、元ファイルがPDFとして特定できた場合、当該PDFがその文書の文字オブジェクトに対して既に文字認識された文字コードを付加ファイルとして有している場合がある。このようなPDFファイルをベクトル化処理の際には、文字コードファイルを用いることで、ステップS130以降のベクトル化処理の中の文字認識処理を省略することが可能となる。従って、ベクトル化処理全体をより高速に行うことが可能となる。

【実施例5】

【0155】

以上、実施形態例を詳述したが、本発明は、例えば、システム、装置、方法、プログラムもしくは記憶媒体等としての実施態様をとることが可能であり、具体的には、複数の機器から構成されるシステムに適用しても良いし、また、一つの機器からなる装置に適用しても良い。

【0156】

尚、本発明は、前述した実施形態の機能を実現するソフトウェアのプログラム(実施形態では図に示すフローチャートに対応したプログラム)を、システムあるいは装置に直接あるいは遠隔から供給し、そのシステムあるいは装置のコンピュータが該供給されたプログラムコードを読み出して実行することによっても達成される場合を含む。

【0157】

従って、本発明の機能処理をコンピュータで実現するために、該コンピュータにインストールされるプログラムコード自体も本発明を実現するものである。つまり、本発明は、本発明の機能処理を実現するためのコンピュータプログラム自体も含まれる。

【0158】

その場合、プログラムの機能を有していれば、オブジェクトコード、インタプリタにより実行されるプログラム、OSに供給するスクリプトデータ等の形態であっても良い。

【0159】

プログラムを供給するための記録媒体としては、例えば、フロッピー(登録商標)ディスク、ハードディスク、光ディスク、光磁気ディスク、MO、CD-ROM、CD-R、CD-RW、磁気テープ、不揮発性のメモリカード、ROM、DVD(DVD-ROM, DVD-R)などがある。

【0160】

その他、プログラムの供給方法としては、クライアントコンピュータのブラウザを用いてインターネットのホームページに接続し、該ホームページから本発明のコンピュータプログラムそのもの、もしくは圧縮され自動インストール機能を含むファイルをハードディスク等の記録媒体にダウンロードすることによっても供給できる。また、本発明のプログラムを構成するプログラムコードを複数のファイルに分割し、それぞれのファイルを異なるホームページからダウンロードすることによっても実現可能である。つまり、本発明の機能処理をコンピュータで実現するためのプログラムファイルを複数のユーザに対してダウンロードさせるWWWサーバも、本発明に含まれるものである。

【0161】

また、本発明のプログラムを暗号化してCD-ROM等の記憶媒体に格納してユーザに配布し、所定の条件をクリアしたユーザに対し、インターネットを介してホームページから暗号化を解く鍵情報をダウンロードさせ、その鍵情報を使用することにより暗号化されたプログラムを実行してコンピュータにインストールさせて実現することも可能である。

10

20

30

40

50

## 【 0 1 6 2 】

また、コンピュータが、読み出したプログラムを実行することによって、前述した実施形態の機能が実現される他、そのプログラムの指示に基づき、コンピュータ上で稼動しているOSなどが、実際の処理の一部または全部を行ない、その処理によっても前述した実施形態の機能が実現され得る。

## 【 0 1 6 3 】

さらに、記録媒体から読み出されたプログラムが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラムの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行ない、その処理によっても前述した実施形態の機能が実現される。

10

## 【 図面の簡単な説明 】

## 【 0 1 6 4 】

【 図 1 】 本発明の第 1 の実施例に係る画像処理システムの構成を示すブロック図である。

【 図 2 】 本発明の第 1 の実施例に係る M F P 1 0 0 の構成を示すブロック図である。

【 図 3 】 本発明の第 1 の実施例に係る画像処理システムによる画像処理の手順について説明するためのフローチャートである。

【 図 4 】 ブロックセレクション処理によって読み取った 1 枚のイメージデータを属性を判定し複数のブロックに分割する様子を示す図である。

【 図 5 】 ブロックセレクション処理で得られた各ブロックに対するブロック情報の一例について示す図である。

20

【 図 6 】 原稿画像中に付加された 2 次元バーコード ( Q R コードシンボル ) を復号してデータ文字列を出力する手順を説明するためのフローチャートである。

【 図 7 】 2 次元バーコードが付加された原稿 3 1 0 の一例を示す図である。

【 図 8 】 検出されたポインタ情報から電子ファイルが格納されているサーバを検索する処理手順について説明するためのフローチャートである。

【 図 9 】 ファイルアクセス権を含むポインタ情報から電子ファイルが格納されているサーバを検索する処理手順について説明するためのフローチャートである。

【 図 1 0 】 データベース内から入力イメージファイルに類似した電子ファイルのレイアウト検索を行う手順について説明するためのフローチャートである。

30

【 図 1 1 】 曲率が極大となる点を説明するための図である。

【 図 1 2 】 外輪郭が内輪郭又は別の外輪郭と近接している場合に太さを持った線として表現する例について説明するための図である。

【 図 1 3 】 ベクトルデータを図形オブジェクト毎にグループ化するまでの処理手順を説明するためのフローチャートである。

【 図 1 4 】 図形要素を検出する処理手順を説明するためのフローチャートである。

【 図 1 5 】 一頁分のイメージデータをブロックセレクション処理及びベクトル化処理によって変換された結果として得られる中間データ形式のファイルのデータ構造を示す図である。

【 図 1 6 】 アプリデータへの変換処理全体の概略手順を説明するためのフローチャートである。

40

【 図 1 7 】 文書構造ツリー生成処理の詳細な処理手順を説明するためのフローチャートである。

【 図 1 8 】 文書構造ツリーの概要を説明するための図である。

【 図 1 9 】 ポインタ情報としてのデータ文字列を 2 次元バーコードを用いて符号化して画像中に付加する手順を説明するためのフローチャートである。

【 図 2 0 】 ステップ S 1 2 0 の入力操作の手順を詳細に説明するためのフローチャートである。

【 図 2 1 】 s v g フォーマットの記述方法を用いて直線近似を行った場合の一例を示す図である。

50

【図 2 2】 s v g フォーマットの記述方法を用いて曲線近似を行った場合の一例を示す図である。

【図 2 3】 s v g フォーマットの記述方法を用いて図形近似を行った場合の一例を示す図である。

【符号の説明】

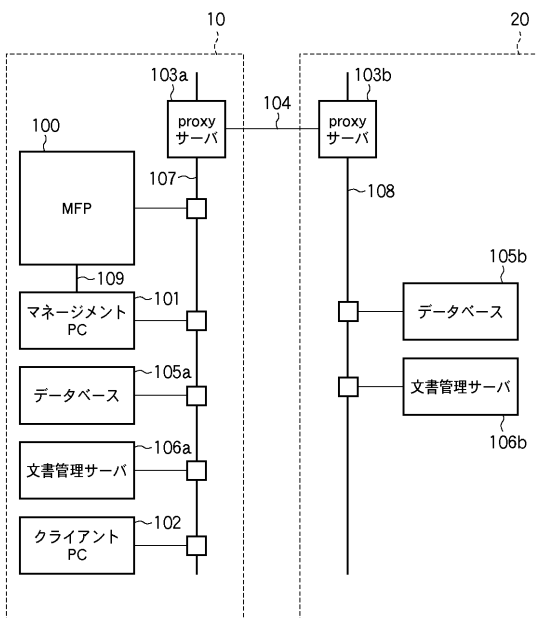
【 0 1 6 5 】

- 1 0 0 デジタル複合機 ( M F P )
- 1 0 1 マネージメント P C
- 1 0 2 クライアント P C
- 1 0 3 a、1 0 3 b プロキシサーバ
- 1 0 4 ネットワーク
- 1 0 5 a、1 0 5 b データベース
- 1 0 6 a、1 0 6 b 文書管理サーバ
- 1 0 7、1 0 8、1 0 9 L A N
- 1 1 0 画像読み取り部
- 1 1 1 記憶装置
- 1 1 2 形成装置
- 1 1 3 入力装置
- 1 1 4、1 1 7 ネットワーク I / F
- 1 1 5 データ処理装置
- 1 1 6 表示装置

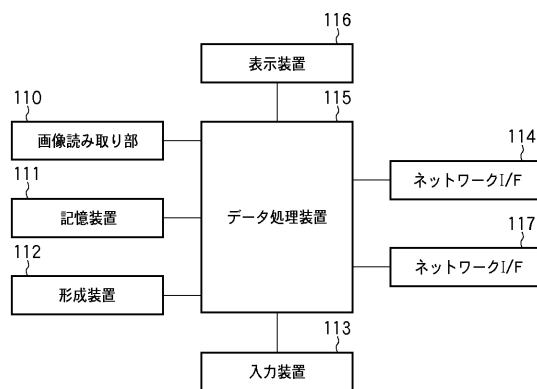
10

20

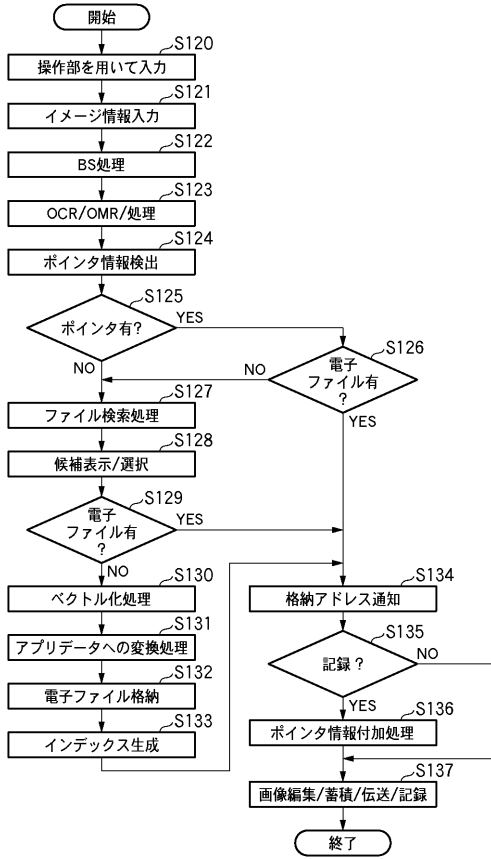
【 図 1 】



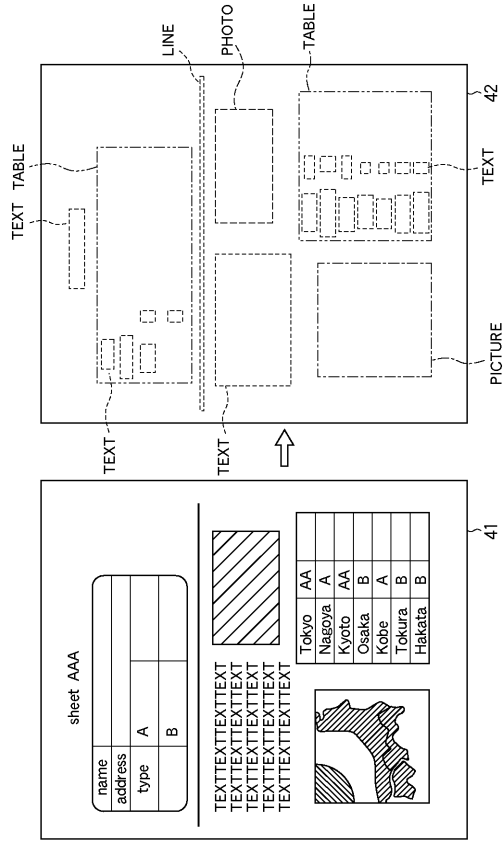
【 図 2 】



【図3】



【図4】



【図5】

ブロック情報

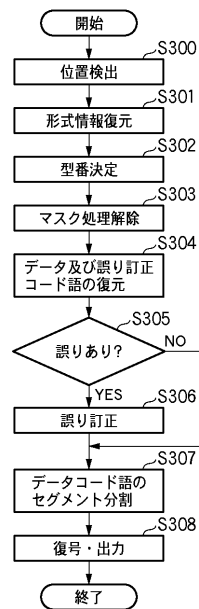
ブロック	属性	座標X	座標Y	幅W	高さH	OCR情報
ブロック1	1	X1	Y1	W1	H1	有
ブロック2	3	X2	Y2	W2	H2	有
ブロック3	2	X3	Y3	W3	H3	無
ブロック4	1	X4	Y4	W4	H4	有
ブロック5	3	X5	Y5	W5	H5	有
ブロック6	5	X6	Y6	W6	H6	無

属性 1: text 2: picture 3: table 4: line 5: photo

入力ファイル情報

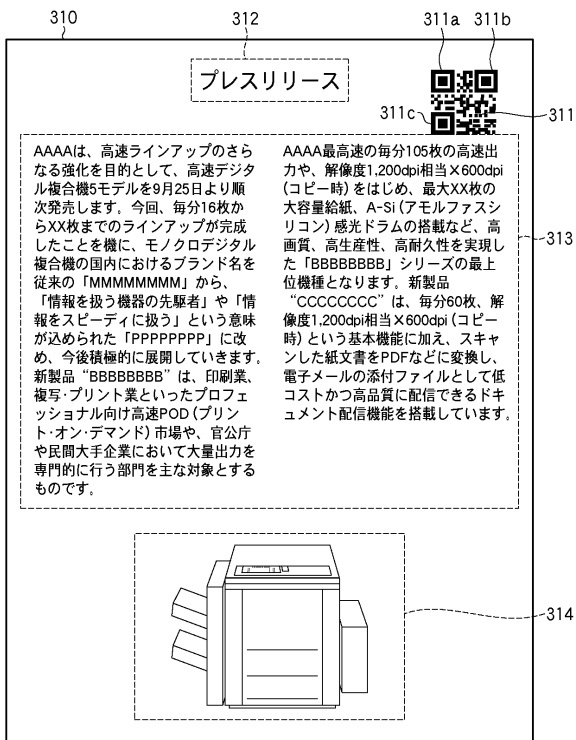
ブロック総数	N (=6)
--------	--------

【図6】

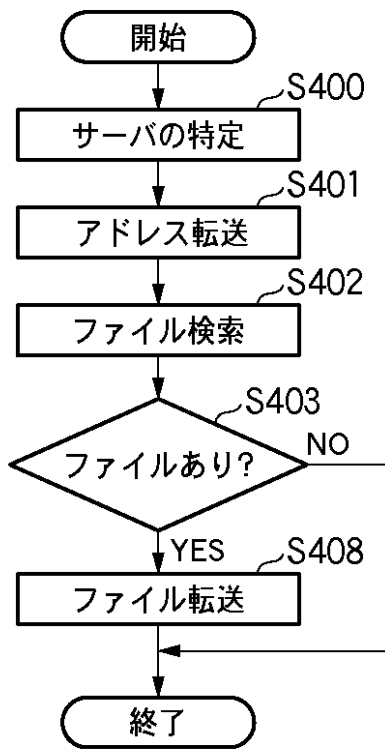




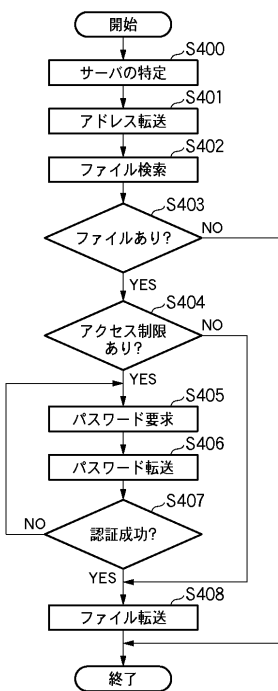
【図7】



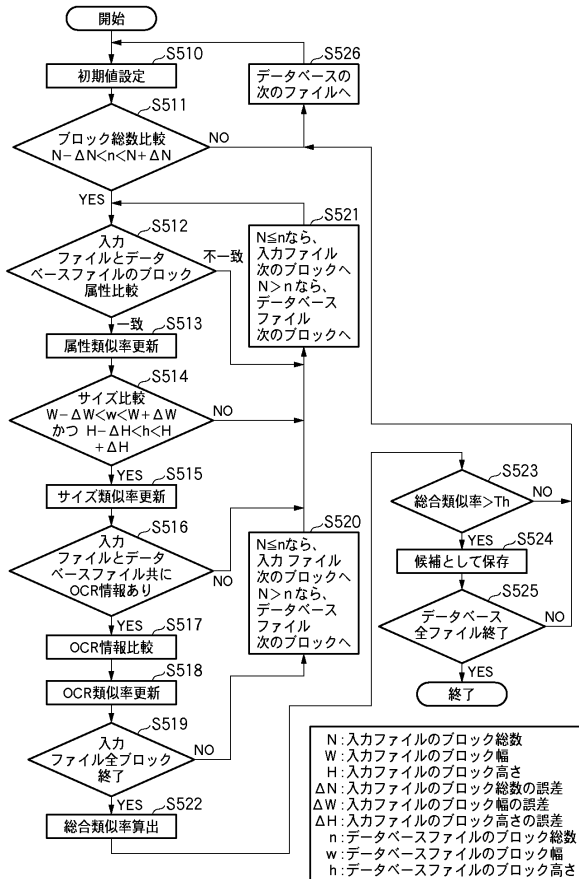
【図8】



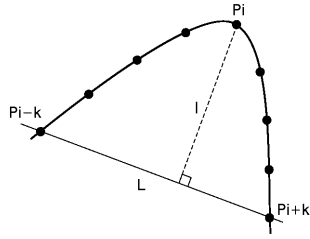
【図9】



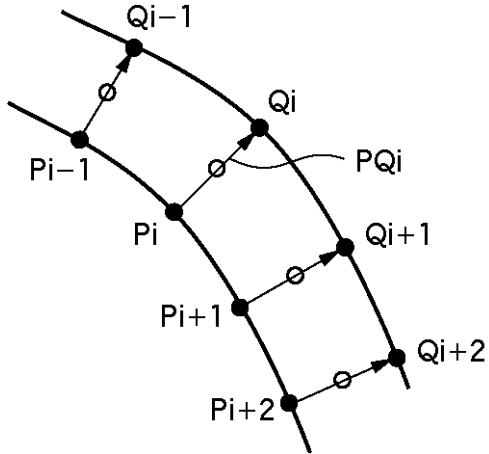
【図10】



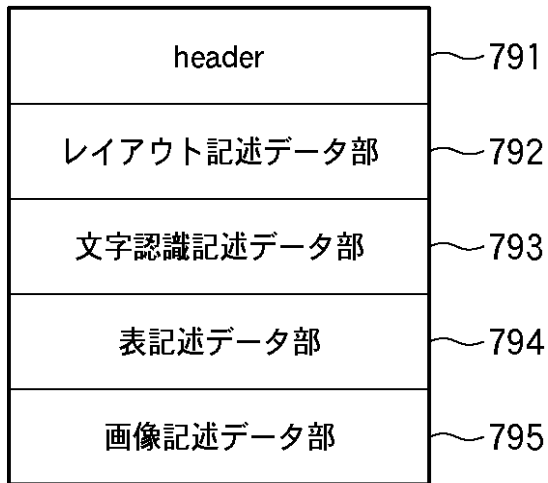
【図11】



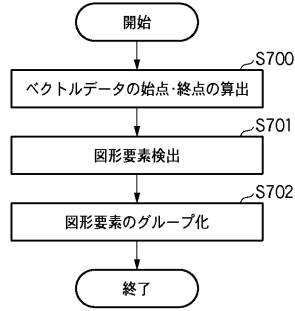
【図12】



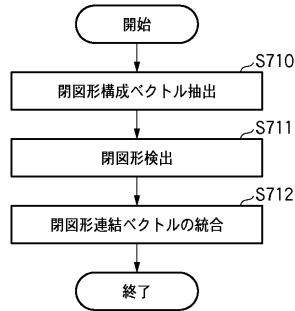
【図15】



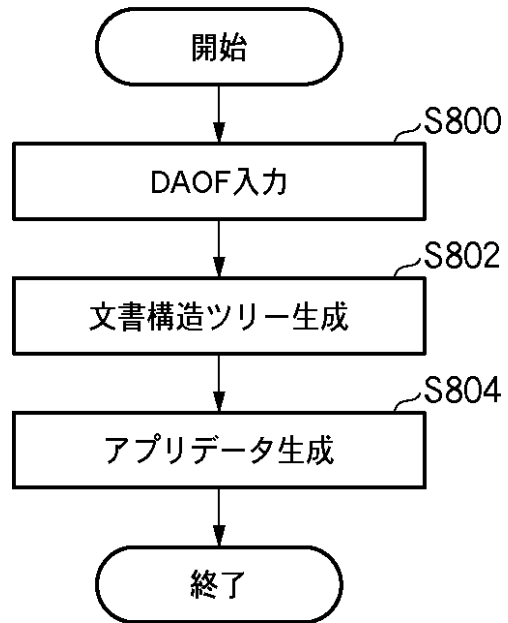
【図13】



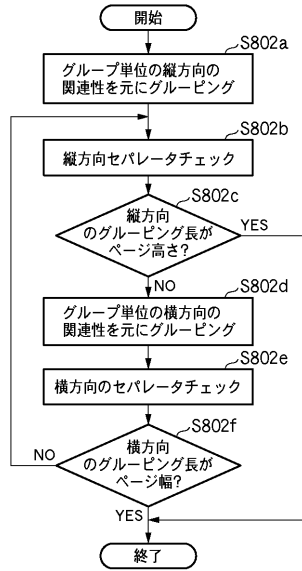
【図14】



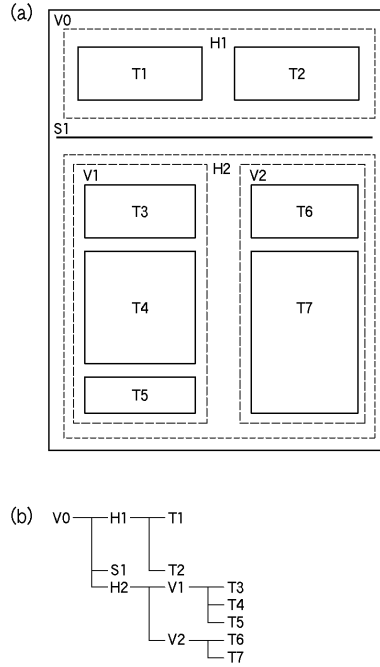
【図16】



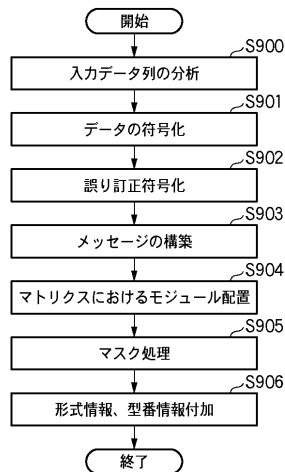
【図 17】



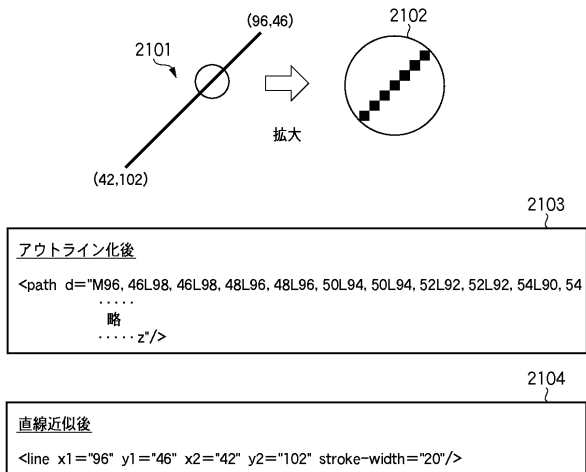
【図 18】



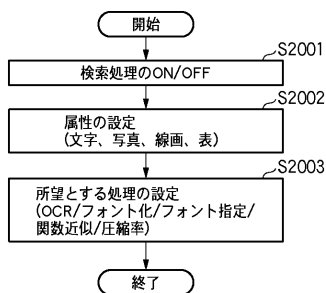
【図 19】



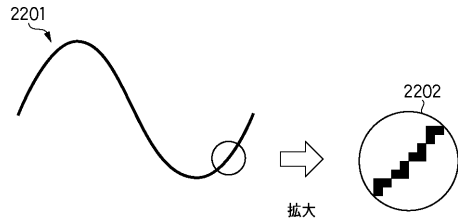
【図 21】



【図 20】



【図 2 2】



2203

アウトライン化後

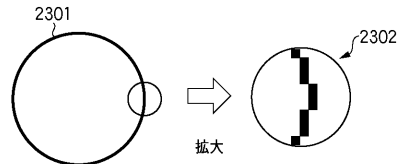
```
<path d="M44, 32L54, 32L54, 34L56, 34L56, 36L58, 36L58, 38L60, 38L60, 40L62, 40
.....
略
.....z"/>
```

2204

曲線近似後

```
<path d="M100,200 C100,100 250,100 250,200 S400,300 400,200"/>
```

【図 2 3】



2303

アウトライン化後

```
<path d="M58, 24L78, 24L78, 26L84, 26L84, 28L90, 28L90, 30L92, 30L92, 32L96,
.....
略
.....z"/>
```

2304

図形近似後

```
<circle cx="70" cy="70" r="30" fill="none" stroke="black" stroke-width="5"/>
```

---

フロントページの続き

(72)発明者 三沢 玲司  
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

審査官 千葉 久博

(56)参考文献 特開2003-125198(JP,A)  
特開2002-319025(JP,A)  
特開平10-143414(JP,A)  
特開平06-014201(JP,A)  
特開平06-004070(JP,A)  
特開平05-342408(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06T 11/60, 11/80  
G06T 1/00 - 1/40  
G06K 9/00 - 9/03, 9/46 - 9/52  
G06K 9/62 - 9/82  
H04N 1/00  
H04N 1/38 - 1/393