

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4646703号
(P4646703)

(45) 発行日 平成23年3月9日(2011.3.9)

(24) 登録日 平成22年12月17日(2010.12.17)

(51) Int.Cl.

F I

B 4 1 J 5/30 (2006.01)
H 0 4 N 1/21 (2006.01)B 4 1 J 5/30 Z
H 0 4 N 1/21

請求項の数 8 (全 40 頁)

(21) 出願番号 特願2005-157609 (P2005-157609)
 (22) 出願日 平成17年5月30日(2005.5.30)
 (65) 公開番号 特開2006-327155 (P2006-327155A)
 (43) 公開日 平成18年12月7日(2006.12.7)
 審査請求日 平成20年5月28日(2008.5.28)

(73) 特許権者 000001007
 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 100076428
 弁理士 大塚 康徳
 (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
 (72) 発明者 保田 昌孝
 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

最終頁に続く

(54) 【発明の名称】 画像処理装置及びその制御方法、プログラム

(57) 【特許請求の範囲】

【請求項 1】

入力された画像データに対する処理を実行する画像処理装置であって、
 画像データを入力する入力手段と、
 画像データを出力する出力手段と、

ラスト画像データ中のオブジェクトを内包するブロックのブロックサイズを設定し、当該設定したブロックサイズのうち所定サイズより大きいブロックサイズに関しては前記所定サイズ以下のブロックサイズに分割して設定し、前記ラスト画像データを、当該設定された前記所定サイズ以下のブロックサイズのブロック毎に分割して、各ブロックにベクトル化処理を実行することで、ブロックベクタ画像データに変換する第1変換手段と、

ブロックベクタ画像データを記憶する第1記憶手段と、

ブロックベクタ画像データをラスト画像データに展開する展開手段と、

前記展開手段から出力されるデータを記憶する第2記憶手段と、

前記入力手段から入力されたラスト画像データを前記第1変換手段でブロックベクタ画像データに変換させ、当該変換されたブロックベクタ画像データを前記第1記憶手段に記憶させ、前記第1記憶手段に記憶されたブロックベクタ画像データを前記展開手段でラスト画像データに展開して前記第2記憶手段に記憶させ、前記第2記憶手段に記憶された前記ラスト画像データを前記出力手段から出力させるように、装置内での処理対象の画像データの転送を制御する画像データ転送制御手段と

を備えることを特徴とする画像処理装置。

10

20

【請求項 2】

ページ全体を示すページベクタ画像データ中のオブジェクトを内包する大きさのブロックを設定し、当該設定したブロックのうち上限サイズを超えている大きさのブロックに関しては該上限サイズ以下のブロックに分割して設定し、前記ページベクタ画像データを、当該設定された前記上限サイズ以下のブロック毎のブロックベクタ画像データへ変換する第 2 変換手段を更に備え、

前記画像データ転送制御手段は、前記入力手段から入力された画像データがページベクタ画像データである場合は、当該入力されたページベクタ画像データを前記第 2 変換手段でブロックベクタ画像データに変換させ、当該変換されたブロックベクタ画像データを前記第 1 記憶手段に記憶させる

10

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】

1 ページ分の前記ブロックベクタ画像データから、ページ全体を示すページベクタ画像データへ変換する第 3 変換手段を更に備え、

前記画像データ転送制御手段は、

前記出力手段から出力させるべき画像データの形式がラスタ画像データ形式であると判断した場合は、前記第 1 記憶手段に記憶されたブロックベクタ画像データを前記展開手段でラスタ画像データに展開して前記第 2 記憶手段に記憶させ、前記第 2 記憶手段に記憶された前記ラスタ画像データを前記出力手段から出力させ、

前記出力手段から出力させるべき画像データの形式がベクタ画像データ形式であると判断した場合は、前記第 1 記憶手段に記憶されたブロックベクタ画像データを前記第 3 変換手段でページベクタ画像データに展開して前記第 1 記憶手段に記憶させ、当該第 1 記憶手段に記憶された前記ページベクタ画像データを前記出力手段から出力させる

20

ように、装置内での処理対象の画像データの転送を制御する

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】

前記画像データ転送制御手段は、バスを介して、各手段に接続され、

処理対象の画像データの形式に応じて、当該処理対象の画像データの転送に用いる前記バスの調整制御を実行するバス調停手段を備える

ことを特徴とする請求項 1 に記載の画像処理装置。

30

【請求項 5】

前記第 1 変換手段あるいは前記第 2 変換手段は、処理対象の画像データを分割したブロック中にオブジェクトが含まれない場合には、対応するブロックベクタ画像データの生成を禁止する

ことを特徴とする請求項 2 に記載の画像処理装置。

【請求項 6】

前記第 1 変換手段は、少なくとも 2 つのオブジェクトそれぞれを内包するブロックの重複部の割合が所定値以上の場合は、前記少なくとも 2 つのオブジェクトを内包するブロックのブロックサイズを設定する一方、前記重複部の割合が前記所定値より小さい場合は、前記少なくとも 2 つのオブジェクトを複数のブロックに分割するブロックサイズを設定する

40

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 7】

入力された画像データに対する処理を実行する画像処理装置の制御方法であって、

前記画像処理装置の第 1 変換手段が、入力されたラスタ画像データ中のオブジェクトを内包するブロックのブロックサイズを設定し、当該設定したブロックサイズのうち所定サイズより大きいブロックサイズに関しては前記所定サイズ以下のブロックサイズに分割して設定し、前記ラスタ画像データを、当該設定された前記所定サイズ以下のブロックサイズのブロック毎に分割して、各ブロックにベクトル化処理を実行することで、ブロックベクタ画像データに変換し、当該変換されたブロックベクタ画像データを前記画像処理装置

50

の第1記憶手段に記憶させる第1変換工程と、

前記画像処理装置の展開手段が、前記第1記憶手段に記憶されたブロックベクタ画像データをラスタ画像データに展開して、当該展開したラスタ画像データを前記画像処理装置の第2記憶手段に記憶させる展開工程と、

前記画像処理装置の出力手段が、前記第2記憶手段に記憶されたラスタ画像データを出力する出力工程と、

を備えることを特徴とする画像処理装置の制御方法。

【請求項8】

コンピュータを、

入力されたラスタ画像データ中のオブジェクトを内包するブロックのブロックサイズを設定し、当該設定したブロックサイズのうち所定サイズより大きいブロックサイズに関しては前記所定サイズ以下のブロックサイズに分割して設定し、前記ラスタ画像データを、当該設定された前記所定サイズ以下のブロックサイズのブロック毎に分割して、各ブロックにベクトル化処理を実行することで、ブロックベクタ画像データに変換し、当該変換されたブロックベクタ画像データを第1記憶手段に記憶させる第1変換手段、

前記第1記憶手段に記憶されたブロックベクタ画像データをラスタ画像データに展開して、当該展開したラスタ画像データを第2記憶手段に記憶させる展開手段、

前記第2記憶手段に記憶されたラスタ画像データを出力する出力手段、

として機能させるためのプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、入力された画像データに対して画像処理を施して管理する画像処理装置及びその制御方法、プログラムに関するものである。

【背景技術】

【0002】

ネットワーク上に接続され、外部あるいは内部の画像データに対する画像データ処理を実行し、その処理した画像データを出力する画像データ入出力システムが知られている。

【0003】

この画像データ入出力システムとしては、MFP (Multi Function Peripheral) と呼ばれるものがある。

【0004】

ここで、従来のMFPを制御するコントローラ100を図28に示す。コントローラ100は、システムバスブリッジ (SBB) 101により、CPU102、メモリコントローラ (MC) 103、汎用バス105、画像処理部110、画像データ展開部 (RIP (Raster Image Processor)) 113が接続されている。

【0005】

汎用バス105には、画像データを蓄積するための：大容量記憶部 (HDD (ハードディスクドライブ)) 107を制御するハードディスクコントローラ (HDD Cont) 106、MFPが接続されているネットワーク108を介して、外部機器間との画像データの転送を行うインターフェースになるネットワークI/F109が接続されている。この画像データとしては、ページベクタ形式 (PDL (ページ記述言語)、PDF、SVG等) の画像データがある。

【0006】

HDD Cont 106には、HDD (ハードディスクドライブ) 107が接続されており、画像データの記憶媒体として使われる。同様に、MC 103には、システムメモリ (Memory) 104が接続されており、画像データを一時記憶するための媒体として使われる。システムメモリ104には、一般的には、DIMMが用いられる。

【0007】

画像処理部110には、スキャナ111及びプリンタ112が接続されている。スキャ

10

20

30

40

50

ナ１１１から入力された画像データは、画像処理部１１０によって所定の画像処理が実施された後、コントローラ１００へ入力される。また、コントローラ１００内に記憶されている画像データは、画像処理部１１０によって所定の画像処理が実施され、プリンタ１１２へ出力される。

【０００８】

コントローラ１００でハンドリングされる画像データは、ネットワーク経由での外部機器との入出力はページベクタ形式（ＰＤＬやＰＤＦ、ＳＶＧ等）、スキャナ１１１やプリンタ１１２との入出力はラスタデータ形式でインターフェースされる。外部機器から入力されたページベクタ形式の画像データは、ＣＰＵ１０２により、プリミティブなオブジェクトにインタプリットされ、ＤＬ（ＤｉｓｐｌａｙＬｉｓｔ）と呼ばれる中間データ（ＤＬデータ）に変換されてから、ＲＩＰ１１３に入力される。

10

【０００９】

これらの画像データはコントローラ１００の内部のシステムメモリ１０４に一旦記憶される。そのため、システムメモリ１０４上には、ラスタデータ、ページベクタデータ（ＰＤＬ等）、ＤＬデータ等の多種類のデータが存在する。

【００１０】

ＨＤＤ１０７には、画像データとしてスキャナ１１３から入力された画像データとＲＩＰ１１３でレンダリングされたラスタ画像データが記憶される。

【００１１】

また、ページラスタ／タイルラスタ変換部１１４は、システムメモリ１０４上に一旦記憶されたラスタデータを所定サイズの矩形ブロック（タイル）に分割して、タイルラスタデータに変換する。この変換したタイルラスタデータは、再度、システムメモリ１０４上に一旦記憶される。システムメモリ１０４上に、一旦記憶されたタイルラスタデータは、画像処理部１１０内部で、複数タイルを同時に処理することも可能である。これにより、画像処理の並列動作による高速化が実現される。

20

【特許文献１】特開２００４－１２０６３９号公報

【発明の開示】

【発明が解決しようとする課題】

【００１２】

上述のようなＭＦＰが扱う画像データの中でも、ラスタ画像データはデータサイズが大きい。そのため、システムメモリ１０４のメモリサイズや、汎用バス１０５及びＨＤＤＣｏｎｔ１０６－ＨＤＤ１０７間のバンド幅等のシステム・リソースを多く消費する。

30

【００１３】

これに加え、ＰＤＬデータ等のページベクタデータでは、システム中でインタプリットを行い、描画オブジェクトを生成するＤＬデータに展開する。その際、ＤＬデータをシステムメモリ１０４にスプールするため、それによるメモリリソースの消費量も莫大なものとなっている。

【００１４】

一方、最近ユーザの出力画像の画質要求がますます高くなり、その解決策の１つとして、画像データに対する高解像度化（高画質化）が促進されている。また、画質と並行してシステムの処理速度向上も要求されている。

40

【００１５】

そのため、上述の様々な要求仕様を満足するために必要なシステム・リソースが肥大化してしまう。従って、プロダクトのコストパフォーマンスでの折り合いをつけることが課題となっている。

【００１６】

また、このプロダクトのコストパフォーマンスに対する課題とは別に、複雑、多様化するシステムを開発するための人的リソースの問題も解決することが要求されている。そして、これを満足するためには、様々な要求仕様に対し、１つの基本システムをスケラブルな形で構成することにより、製品ラインナップを効率良く整備することが課題となって

50

いる。

【 0 0 1 7 】

例えば、図 2 8 における画像処理部 1 1 0 や R I P 1 1 3 等のモジュールを、ハイエンドな機種では複数個実装し、分散処理できるようなシステムが必要となってくる。

【 0 0 1 8 】

他方、オフィスにおいてはペーパーレス化が進行しており、紙の出力物と電子データをシームレスに扱えるような要求も生まれてきている。そのためには、紙と電子データとの I / F 機器である M F P においても、例えば、蓄積画像データの検索性の向上や、ラスト画像データをオブジェクト化して再利用可能なオブジェクトデータに変換する、P O D (P r i n t O n D e m a n d) 印刷に対応するために画像加工を高速化する、といったよりインテリジェントな機能を持つことが必要となっている。

10

【 0 0 1 9 】

本発明は上記の課題を解決するためになされたものであり、システム全体のシステム・リソースの制約を緩和し、かつトータルスループットを向上することができる画像処理装置及びその制御方法、プログラム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】

【 0 0 2 0 】

上記課題を解決するための本発明による画像処理装置は以下の構成を備える。即ち、
入力された画像データに対する処理を実行する画像処理装置であって、
画像データを入力する入力手段と、
画像データを出力する出力手段と、

20

ラスト画像データ中のオブジェクトを内包するブロックのブロックサイズを設定し、当該設定したブロックサイズのうち所定サイズより大きいブロックサイズに関しては前記所定サイズ以下のブロックサイズに分割して設定し、前記ラスト画像データを、当該設定された前記所定サイズ以下のブロックサイズのブロック毎に分割して、各ブロックにベクトル化処理を実行することで、ブロックベクタ画像データに変換する第 1 変換手段と、

ブロックベクタ画像データを記憶する第 1 記憶手段と、

ブロックベクタ画像データをラスト画像データに展開する展開手段と、

前記展開手段から出力されるデータを記憶する第 2 記憶手段と、

前記入力手段から入力されたラスト画像データを前記第 1 変換手段でブロックベクタ画像データに変換させ、当該変換されたブロックベクタ画像データを前記第 1 記憶手段に記憶させ、前記第 1 記憶手段に記憶されたブロックベクタ画像データを前記展開手段でラスト画像データに展開して前記第 2 記憶手段に記憶させ、前記第 2 記憶手段に記憶された前記ラスト画像データを前記出力手段から出力させるように、装置内での処理対象の画像データの転送を制御する画像データ転送制御手段と

30

を備える。

【 0 0 2 3 】

また、好ましくは、ページベクタ画像データ中のオブジェクトを内包する大きさのブロックを設定し、当該設定したブロックのうち上限サイズを超えている大きさのブロックに関しては該上限サイズ以下のブロックに分割して設定し、前記ページベクタ画像データを、当該設定された前記上限サイズ以下のブロック毎のブロックベクタ画像データへ変換する第 2 変換手段を更に備え、

40

前記画像データ転送制御手段は、前記入力手段から入力された画像データがページベクタ画像データである場合は、当該入力されたページベクタ画像データを前記第 2 変換手段でブロックベクタ画像データに変換させ、当該変換されたブロックベクタ画像データを前記第 1 記憶手段に記憶させる。

【 0 0 2 4 】

また、好ましくは、1 ページ分の前記ブロックベクタ画像データから、ページ全体を示すページベクタ画像データへ変換する第 3 変換手段を更に備え、

前記画像データ転送制御手段は、

50

前記出力手段から出力させるべき画像データの形式がラスタ画像データ形式であると判断した場合は、前記第1記憶手段に記憶されたブロックベクタ画像データを前記展開手段でラスタ画像データに展開して前記第2記憶手段に記憶させ、前記第2記憶手段に記憶された前記ラスタ画像データを前記出力手段から出力させ、

前記出力手段から出力させるべき画像データの形式がベクタ画像データ形式であると判断した場合は、前記第1記憶手段に記憶されたブロックベクタ画像データを前記第3変換手段でページベクタ画像データに展開して前記第1記憶手段に記憶させ、当該第1記憶手段に記憶された前記ページベクタ画像データを前記出力手段から出力させる。

【0026】

また、好ましくは、前記画像データ転送制御手段は、バスを介して、各手段に接続され、処理対象の画像データの形式に応じて、当該処理対象の画像データの転送に用いる前記バスの調整制御を実行するバス調停手段を備える。

【0027】

また、好ましくは、前記第1変換手段あるいは前記第2変換手段は、処理対象の画像データを分割したブロック中にオブジェクトが含まれない場合には、対応するブロックベクタ画像データの生成を禁止する。

【0028】

また、好ましくは、前記第1変換手段は、少なくとも2つのオブジェクトそれぞれを内包するブロックの重複部の割合が所定値以上の場合は、前記少なくとも2つのオブジェクトを内包するブロックのブロックサイズを設定する一方、前記重複部の割合が前記所定値より小さい場合は、前記少なくとも2つのオブジェクトを複数のブロックに分割するブロックサイズを設定する。

【0032】

上記課題を解決するための本発明による画像処理装置の制御方法は以下の構成を備える。即ち、

入力された画像データに対する処理を実行する画像処理装置の制御方法であって、

前記画像処理装置の第1変換手段が、入力されたラスタ画像データ中のオブジェクトを内包するブロックのブロックサイズを設定し、当該設定したブロックサイズのうち所定サイズより大きいブロックサイズに関しては前記所定サイズ以下のブロックサイズに分割して設定し、前記ラスタ画像データを、当該設定された前記所定サイズ以下のブロックサイズのブロック毎に分割して、各ブロックにベクトル化処理を実行することで、ブロックベクタ画像データに変換し、当該変換されたブロックベクタ画像データを前記画像処理装置の第1記憶手段に記憶させる第1変換工程と、

前記画像処理装置の展開手段が、前記第1記憶手段に記憶されたブロックベクタ画像データをラスタ画像データに展開して、当該展開したラスタ画像データを前記画像処理装置の第2記憶手段に記憶させる展開工程と、

前記画像処理装置の出力手段が、前記第2記憶手段に記憶されたラスタ画像データを出力する出力工程と、

を備える。

【0033】

上記課題を解決するための本発明によるプログラムは以下の構成を備える。即ち、コンピュータを、

入力されたラスタ画像データ中のオブジェクトを内包するブロックのブロックサイズを設定し、当該設定したブロックサイズのうち所定サイズより大きいブロックサイズに関しては前記所定サイズ以下のブロックサイズに分割して設定し、前記ラスタ画像データを、当該設定された前記所定サイズ以下のブロックサイズのブロック毎に分割して、各ブロックにベクトル化処理を実行することで、ブロックベクタ画像データに変換し、当該変換されたブロックベクタ画像データを第1記憶手段に記憶させる第1変換手段、

前記第1記憶手段に記憶されたブロックベクタ画像データをラスタ画像データに展開して、当該展開したラスタ画像データを第2記憶手段に記憶させる展開手段、

10

20

30

40

50

前記第２記憶手段に記憶されたラスタ画像データを出力する出力手段、
として機能させる。

【発明の効果】

【００３４】

本発明によれば、システム全体のシステム・リソースの制約を緩和し、かつトータルスループットを向上することができる画像処理装置及びその制御方法、プログラムを提供できる。

【発明を実施するための最良の形態】

【００３５】

以下、本発明の実施の形態について図面を用いて詳細に説明する。

10

【００３６】

[実施形態１]

[MFP装置概要]

図１は本発明の実施形態１の画像処理システムを構成するMFPのコントローラの詳細を示すブロック図である。

【００３７】

MFP１０００を制御するコントローラ１は、システムバスブリッジ（SBB）２により、CPU３、メモリコントローラ（MC）４、汎用バス６、タイル／ページベクタ変換部１３、ラスタ／ベクタ変換部１４、画像処理部１５、画像データ展開部（RIP）１８が接続されている。

20

【００３８】

ここで、このRIP１８は、タイルベクタデータを展開することが可能であり、内部には、複数の小画像データ展開部（μRIP）１８ａ～１８ｄが構成されている。

【００３９】

MC４には、システムメモリ（Memory）５が接続されており、画像データを一時記憶するための媒体として使われる。

【００４０】

汎用バス６には、画像データを蓄積するためのHDD８を制御するハードディスクコントローラ（HDDCont）７及び操作部（例えば、LCD等から構成されるタッチパネル）１０を制御する操作部コントローラ９及び、MFP１０００が接続されているネットワーク１２を介して、外部機器間との画像データの転送を行うインターフェースになるネットワークI/F１１が接続されている。

30

【００４１】

尚、操作部１０では、実施形態１及び後述する各実施形態の各種処理の実行指示の入力や、処理結果等を表示するための操作画面が表示され、ユーザは、この操作画面を介して各種操作を実現することができる。

【００４２】

ラスタ／タイルベクタ変換部１４には、画像処理部１５が接続されている。また、画像処理部１５には、スキャナ１６及びプリンタ１７が接続されている。

【００４３】

また、SBB２には、RIP１８が接続されている。また、RIP１８には、RIP１８から出力されるデータを記憶するローカルメモリ（Local Memory）１９が接続されている。

40

【００４４】

コントローラ１でハンドリングされる画像データは、外部機器との入出力はベクタ（PDL、PDF、SVGなど）形式の画像データ（以下、ベクタデータとも言う）、スキャナ１６やプリンタ１７との入出力はラスタ形式の画像データ（以下、ラスタデータとも言う）でインターフェースされる。

【００４５】

コントローラ１では、スキャンデータ（ラスタデータ）はラスタ／タイルベクタ変換部

50

14でタイルベクタデータに変換される。また、このタイルベクタデータからRIP18の処理によって得られるタイルDLデータは、RIP18に接続されたローカルメモリ19に記憶される。

【0046】

従って、システムメモリ5上には、ページベクタデータとタイルベクタデータの2種類の画像のみが記憶される。つまり、画像サイズの大きいラスタデータ及びDLデータをシステムメモリ5に記憶する必要がなくなる。そのため、システムメモリ5上で確保しなければならない画像データ領域を削減することができる。

【0047】

また、RIP18から出力されるDLデータはタイル単位に分割された、タイルDLデータで記憶される。そのため、従来のページ単位のページDLデータに比べ、非常に少ないメモリ容量で記憶できる。従って、ローカルメモリ19はオンチップ上に実装することが可能になり、メモリレイテンシを小さくできる。その結果、タイルデータ展開速度を高速化することが可能となる。

【0048】

また、タイルベクタデータのみを画像データとして、HDD8上に記憶すれば良いので、HDD8へのアクセス速度のボトルネックが緩和され、データ処理の高速化が図れる。同時に、タイル単位で処理することにより、RIP18のコストダウンも可能となる。

【0049】

より高い処理能力が要求される場合は、RIP18内に備えるμRIP18a~18dを並列に複数実装することで、処理能力を可変にすることができる。このようにすることで、コントローラ1の処理能力がシンプルに調整できることから、スケーラビリティの確保が容易なシステムを構築できる。

【0050】

尚、本発明では、ネットワークI/F11及びスキャナ16がコントローラ1内へ画像データを入力する画像入力部として機能する。また、ネットワークI/F11及びプリンタ17が画像データを出力する画像出力部として機能する。

【0051】

以下に、MFP1000が実現可能な各種処理のデータフローについて説明する。

【0052】

[コピー]

図2は本発明の実施形態1の画像処理システムでのコピー動作に係るデータフローを示す図である。

【0053】

尚、このデータフローは、CPU3の制御の下、MFP1000を構成する各種構成要素を協調動作させることで実現される。

【0054】

また、図2中に示す矢印は、各種データフローを示している。特に、実線矢印はラスタデータ(ラスタ画像データ)、破線矢印はタイルベクタデータ(タイルベクタ画像データ)、1点鎖線矢印はページベクタデータ(ページベクタ画像データ)のデータフローを示している。ページベクタデータ及びタイルベクタデータについては、後述のタイル/ページベクタ変換部13で詳細に説明する。

【0055】

(S21): 操作部10より、ユーザがコピー開始を指示すると、スキャナ16は原稿画像の読取動作を開始する。スキャナ14より画像処理部15へ入力された画像(R、G、B画像)は画像処理部15のクロック同期に周波数変換された後、例えば、以下の処理が実行される。

【0056】

1) スキャナ16内のCCDセンサのラインピッチや色収差等のスキャナ特性の補正処理

10

20

30

40

50

2) 色空間補正やシャープネス等の入力画像データの画質補正処理

3) 入力画像データの枠消やブック枠消等の画像加工処理

(S22): 画像処理部15による画像処理が終了し、画像処理部15から出力された画像データはラスタ/タイルベクタ変換部14に入力され、タイルベクタ変換処理が実行され。即ち、ラスタ/タイルベクタ変換部14は、画像データ中のオブジェクトを内包する大きさのブロック(タイル)を設定して、その大きさのブロックに画像データ中のオブジェクトを分割する。そして、各ブロック内のラスタデータに対して、ベクトル化処理を実行して、ブロック(タイル)単位のベクタデータ(タイルベクタデータ(ブロックベクタデータ))を生成する。

【0057】

10

生成されたタイルベクタデータは、SBB2によりバスの調停を受け、システムメモリ5へのバス権を取得し、MC4を介して、システムメモリ5に記憶される。(尚、SBB2経由でデータバスが接続される場合は、基本的にバスの調停を受け、バス権を取得する手続きを踏むが、以降のフロー説明では省略する。)

(S23): システムメモリ5に記憶されたタイルベクタデータは、HDDCont7とMC4を介して、SBB2経由でHDD8に記憶される。HDD8にタイルベクタデータを記憶することにより、複数部の原稿をコピーする時にソーティングをして、ページ順を変えて出力したり、MFP1000内に保存画像データとして記憶したりすることができる。

【0058】

20

(S24): HDD8に記憶されたタイルベクタデータは、プリンタ17内のプリンタCPU(不図示)から送られてくるプリンタレディのタイミングに合わせて、HDDCont7により読み出され、SBB2、MC4を経由してシステムメモリ5に一時的に記憶される。

【0059】

仮に、読み取ったタイルベクタデータをHDD8からダイレクトにプリンタ17へ出力する場合、HDD8のアクセススピードが律則したり、汎用バス6のバスの混雑度合によりプリンタ17に同期して出力することが保証できなくなる。そのため、プリンタ17に同期してデータ転送を行う前に、システムメモリ5にページベクタデータをスプールしておくことで、リアルタイムなスループットを保証する。

30

【0060】

(S25): システムメモリ5に記憶されたタイルベクタデータは、プリンタ17からコントローラ1に送られる起動信号に従って、MC4によって読み出され、SBB2を介してRIP18に転送される。

【0061】

RIP18では、まず、タイルベクタデータを解析し、タイル単位の描画オブジェクト(タイルDLデータ)の生成(インタプリット)を実行する。生成されたタイルDLデータはローカルメモリ19に一旦記憶される。

【0062】

RIP18はローカルメモリ19からタイルDLデータを読み出して、タイル単位のラスタデータ(タイルラスタデータ)へと展開し、出力する。

40

【0063】

実施形態1では、上述のように、RIP18内に4つの小画像データ展開部(μ RIP)18a~18dを備えている。コントローラ1は、 μ RIP18a~ μ RIP18dを並列に動作させることにより、タイルベクタデータの展開を高速に行わせることができる。

【0064】

ここで、画像処理システムの全体パフォーマンスは、ベクタデータ展開時間が支配的であり、この μ RIPの構成数を増やすことで、パフォーマンスアップが見込める。そのため、本発明のような構成を用いると、その構成数あるいは動作させる構成数を増減させる

50

ことで、容易にスケーラブルなシステムを構築することが可能となる。

【0065】

(S26) : R I P 1 8 によって生成されたタイルラスタデータは、画像処理部 1 5 に転送され、例えば、以下の処理が実行される。

【0066】

- 1) タイルラスタデータからページラスタデータへの変換処理
- 2) プリンタの特性に合わせた出力画像の色や濃度の補正処理
- 3) 画像データを量子化して出力画像の階調変換を行う中間調処理
- 4) プリンタ I / F クロックに同期して画像を出力するための周波数変換処理

そして、画像処理部 1 5 で、1) ~ 4) の画像処理が実行されて得られたラスタデータは、プリンタ 1 7 に転送され、記録媒体上に印刷され出力される。

10

【0067】

[プリント]

図 3 は本発明の実施形態 1 の画像処理システムでのプリント動作に係るデータフローを示す図である。

【0068】

尚、このデータフローは、C P U 3 の制御の下、M F P 1 0 0 0 を構成する各種構成要素を協調動作させることで実現される。

【0069】

(S31) : ネットワーク 1 2 に接続された外部機器より、汎用バス 6 に接続されたネットワーク I / F 1 1 がページベクタデータを受信する。そして、S B B 2 の先に接続された M C 4 を介してシステムメモリ 5 に転送する。

20

【0070】

(S32) : システムメモリ 5 に記憶されたページベクタデータは、タイル / ページベクタ変換部 1 3 より読み出され、タイルベクタ変換処理が実行される。即ち、タイル / ページベクタ変換部 1 3 は、ページベクタデータ内に存在するオブジェクトを内包する大きさのブロック (タイル) を設定して、その大きさのブロックにオブジェクトを分割する。そして、タイル単位のベクタデータ (タイルベクタデータ) を生成する。

【0071】

(S33) : 生成されたタイルベクタデータは、S B B 2 を介して再度システムメモリ 5 に記憶される。

30

【0072】

(S34) : システムメモリ 5 に記憶されたタイルベクタデータは、H D D C o n t 7 と M C 4 を介して、S B B 2 経由で H D D 8 に記憶される。H D D 8 にタイルベクタデータを記憶することにより、複数部の原稿をコピーする時にソーティングをして、ページ順を変えて出力したり、M F P 1 0 0 0 内に保存画像データとして記憶したりすることができる。

【0073】

(S35) : H D D 8 に記憶されたタイルベクタデータは、プリンタ 1 7 内の C P U (不図示) から送られてくるプリンタレディのタイミングに合わせて、H D D C o n t 7 より読み出され、S B B 2、M C 4 を経由してシステムメモリ 5 に一時的に記憶される。

40

【0074】

読み取ったタイルベクタデータを H D D 8 からダイレクトにプリンタ 1 7 へ出力する場合、H D D 8 のアクセススピードが律則したり、汎用バス 6 のバスの混雑度合によりプリンタ 1 7 に同期して出力することが保証できなくなる。そのため、プリンタ 1 7 に同期してデータ転送を行う前に、システムメモリ 5 に 1 ページ分のベクタ画像データをスプールすることにより、リアルタイムなスループットを保証する。

【0075】

(S36) : システムメモリ 5 に記憶されたタイルベクタデータは、プリンタ 1 7 からコントローラ 1 に送られる起動信号に従って、M C 4 によって読み出され、S B B 2 を介

50

してR I P 1 8 に転送される。

【 0 0 7 6 】

R I P 1 8 では、まず、タイルベクタデータを解析し、タイル単位の描画オブジェクト（タイルD L データ）の生成（インタプリット）を実行する。生成されたタイルD L データはローカルメモリ 1 9 に一旦記憶される。

【 0 0 7 7 】

R I P 1 8 はローカルメモリ 1 9 からタイルD L データを読み出して、タイル単位のラスタデータ（タイルラスタデータ）へと展開し、出力する。

【 0 0 7 8 】

（ S 3 7 ）： R I P 1 8 によって生成されたタイルラスタデータは、画像処理部 1 5 に転送され、例えば、以下の処理が実行される。

【 0 0 7 9 】

- 1) タイルラスタデータからページラスタデータへの変換処理
- 2) プリンタの特性に合わせた出力画像の色や濃度の補正処理
- 3) 画像データを量子化して出力画像の階調変換を実行する中間調処理
- 4) プリンタ I / F クロックに同期して画像を出力するための周波数変換処理

そして、画像処理部 1 5 で、 1) ~ 4) の画像処理が実行されて得られたラスタデータは、プリンタ 1 7 に転送され、記録媒体上に印刷され出力される。

【 0 0 8 0 】

[送信]

図 4 は本発明の実施形態 1 の画像処理システムでの送信動作に係るデータフローを示す図である。

【 0 0 8 1 】

尚、このデータフローは、C P U 3 の制御の下、M F P 1 0 0 0 を構成する各種構成要素を協調動作させることで実現される。

【 0 0 8 2 】

また、画像データをH D D 8 に格納するまでのデータフローについては、ラスタデータの場合は[コピー]と、ネットワーク 1 2 上の外部機器からの入力されたページベクタデータの場合は[プリント]と同一であるので、その説明は割愛する。

尚、画像データをH D D 8 に格納する処理は、ユーザからの格納指示によって実行されるようにしてもよいし、[コピー]や[プリント]の処理の際にH D D 8 に自動的に残しておくようにしてもよい。このようにして、H D D 8 に格納された画像データの中から、ユーザによって指定された画像データを送信するように指示された際に行われる送信処理について説明する。

【 0 0 8 3 】

（ S 4 1 ）： H D D 8 に記憶されたタイルベクタデータはS B B 2 を介して、汎用バス 6 に接続されたH D D C o n t 7 より読み出され、システムメモリ 5 に一時的に記憶される。

【 0 0 8 4 】

（ S 4 2 ）： システムメモリ 5 に記憶されたタイルベクタデータは、タイル / ページベクタ変換部 1 3 より読み出され、タイルベクタ変換処理を実行する。即ち、ブロック単位に分割されたオブジェクトを結合し、ページ全体でオブジェクトを記述したページベクタデータを生成する。つまり、1 ページ分のタイルベクタデータを、ページ全体のベクタデータを示すページベクタデータを生成する。

【 0 0 8 5 】

（ S 4 3 ）： 生成されたページベクタデータは、S B B 2 を介して再度システムメモリ 5 に記憶される。

【 0 0 8 6 】

（ S 4 4 ）： システムメモリ 5 中に記憶されたページベクタデータは、汎用バス 6 に接続されたネットワーク I / F 1 1 から読み出され、ネットワーク 1 2 に接続された外部機

10

20

30

40

50

器へと送信転送される。

【 0 0 8 7 】

本発明のように、外部機器に送信する際に、タイルベクタデータをページベクタデータに戻して、そのデータを構成するオブジェクト数を減らすことで、送信データ量を削減することができる。また、P D F や S V G 等の汎用フォーマットへ容易に変換することができる。

【 0 0 8 8 】

尚、本発明では、スキャナ 1 6 から入力したラスタデータを、外部機器に送信することも可能である。この場合は、そのラスタデータをページベクタに変換してから、外部機器へ送信することが好ましい。

10

【 0 0 8 9 】

[ラスタ / タイルベクタ変換部]

次に、ラスタ / タイルベクタ変換部 1 4 の処理の詳細について説明する。

【 0 0 9 0 】

図 5 は本発明の実施形態 1 のラスタ / タイルベクタ変換部が実行する処理を示すフローチャートである。

【 0 0 9 1 】

(ステップ S 5 1 : ブロックセレクション (領域分割 : B S) 処理)

画像処理部 1 5 より入力されたラスタデータ (イメージデータ) を、文字あるいは線画を含む文字・線画領域と、ハーフトーンの写真領域、不定形の画像領域その他に分割する。さらに、文字・線画領域については、主に文字を含む文字領域と、主に表、図形等を含む線画領域とに分離し、線画領域は表領域と図形領域に分割する。

20

【 0 0 9 2 】

尚、実施形態 1 では、処理対象中の画像の連結画素を検知し、その連結画素の外接矩形領域の形状・サイズ・画素密度等の特徴量を用いて、属性毎の領域に分離するものとするが、その他の領域分割手法を用いても構わない。

【 0 0 9 3 】

文字領域については、文字段落ごとの纏まった塊をブロックとして矩形ブロック (文字領域矩形ブロック) にセグメント化する。線画領域では、表、図形等の個々のオブジェクト (表領域矩形ブロック、線画領域矩形ブロック) ごとに矩形ブロックにセグメント化する。

30

【 0 0 9 4 】

ハーフトーンで表現される写真領域は、画像領域矩形ブロック、背景領域矩形ブロック等のオブジェクトごとに、矩形ブロックにセグメント化する。

【 0 0 9 5 】

(ステップ S 5 2 : タイルサイズ決定処理)

分離された各領域は、さらに各領域のオブジェクトの大きさを判定し、その判定結果に基づいて、各オブジェクトを内包可能なタイルサイズを決定 (設定) する。

【 0 0 9 6 】

具体例について、図 6 を用いて説明する。

40

【 0 0 9 7 】

図 6 は本発明の実施形態 1 のタイルサイズ決定処理の具体例を説明するための図である。

【 0 0 9 8 】

図 6 に示す、ラスタデータ 6 0 0 では、オブジェクト 6 0 1 及び 6 0 2 の 2 つのオブジェクトが存在している。このラスタデータ 6 0 0 に対して、ステップ S 5 1 のブロックセレクション処理を実行すると、オブジェクト 6 0 1 は文字領域として認識され、オブジェクト 6 0 2 は図形領域として認識される。

【 0 0 9 9 】

この処理結果に対して、タイルサイズ決定処理を実行すると、ラスタデータ 6 0 0 は、

50

例えば、図 7 に示すように、異なる大きさのタイル 701 ~ 709 に分割される。ここで、タイル 704 は、オブジェクト 601 を内包可能な大きさに設定される。また、タイル 707 は、オブジェクト 602 を内包可能な大きさに設定される。

【0100】

また、タイル 701、702、704、705、706、708、709 の各タイルは、オブジェクトが存在しない空白タイルとなる。

【0101】

(ステップ S53 : タイルサイズ判定処理)

次に、ステップ S52 で決定された各タイルに関して、そのサイズ(大きさ)の判定を実行する。

【0102】

例えば、図 7 の例の場合、作成されたタイル 701 ~ 709 に関して、その大きさの判定を実行する。この判定は、以下の理由によって実行するものである。

【0103】

つまり、ステップ S52 で一旦決定したタイルサイズが大きい場合、RIP18 内の μ RIPa18a ~ μ RIP18d で並列してタイルベクタデータを展開する際に、一つのタイルサイズが大きすぎると並列動作による処理の高速化が行いづらくなる。そこで、これを防ぐため、各タイルサイズを判定して、所定サイズよりも大きいタイルに関しては、より小さいサイズのタイルに分割する。

【0104】

仮に、オブジェクト 602 を内包する図 7 のタイル 707 が、所定サイズよりも大きい場合は、図 8 に示すように、タイル 707 を、そのサイズより小さいサイズの 4 つのタイル 801、802、803、804 に分割する。そして、全てのタイルが所定サイズよりも小さくなったら、以下のベクトル化処理により、個々のタイルサイズに基づいて、ページベクタデータをタイルベクタデータに変換する。

【0105】

(ステップ S54 : ベクトル化処理)

ベクトル化処理により、各属性の領域のイメージデータをベクトルデータに変換する(ベクトル化する)。ベクトル化の方法は、例えば、以下、(a) ~ (f) 等が存在する。

【0106】

(a) 属性領域が文字領域のとき、さらに OCR による文字画像のコード変換を行った、あるいは文字のサイズ、スタイル、字体を認識し、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する。

【0107】

(b) 属性領域が文字領域であり、かつ OCR による認識が不可能であったとき、文字の輪郭を追跡し、輪郭情報(アウトライン)を線分のつながりとして表現する形式に変換する。

【0108】

(c) 属性領域が図形領域のとき、図形オブジェクトの輪郭を追跡し、輪郭情報を線分のつながりとして表現する形式に変換する。

【0109】

(d) (b) 及び (c) の線分形式のアウトライン情報をベジエ関数等でフィッティングして、関数情報に変換する。

【0110】

(e) (c) の図形オブジェクトの輪郭情報から、図形の形状を認識し、円、矩形、多角形、等の図形定義情報に変換する。

【0111】

(f) 属性領域が図形領域であって、特定領域の表形式のオブジェクトの場合、罫線や枠線を認識し、所定のフォーマットの帳票フォーマット情報に変換する。

【0112】

10

20

30

40

50

(ステップS55: タイルベクタデータ生成処理)

ステップS54で、(a)~(f)のフォーマットコード情報、図形情報、関数情報等のコマンド定義形式情報にベクトル変換されたデータに対し、コントローラ1内でページベクタデータかタイルベクタデータかを判別するベクタタイプや、当該タイルのページ内の座標位置等の座標情報を判別するためのヘッダ情報を付加した、タイルベクタデータを生成する。このようにして、タイル単位に各種情報が付加されたタイルベクタデータをSBB2へ出力する。

【0113】

(ステップS56: 終了判定処理)

処理対象のラスタデータの有無を判定する。処理対象のラスタデータがある場合(ステップS56でNO)、ステップS51に戻る。一方、処理対象のラスタデータがない場合(ステップS56でYES)、処理を終了する。

【0114】

[タイル/ページベクタ変換部]

次に、タイル/ページベクタ変換部13の処理の詳細を説明するにあたり、処理対象となるドキュメントデータ(画像データ)について説明する。

【0115】

図9は本発明の実施形態1のネットワークから転送されてくるドキュメントデータの一例を示している。

【0116】

図9では、ドキュメントデータ1101の短手方向を「X」方向、長手方向を「Y」方向とするデバイス座標系を定義する。ドキュメントデータ1101としては、ページベクタデータ、タイルベクタデータ、あるいはラスタデータ表現を含むページベクタデータ(タイルベクタデータ)、あるいはラスタデータのいずれかで構成される可能性がある。

【0117】

ここで、ドキュメントデータ1101が、ページベクタデータである場合、その内容を構成する記述例について、図10を用いて説明する。

【0118】

図10は本発明の実施形態1のページベクタデータの記述例を示す図である。

【0119】

図10において、1201はドキュメントデータ全体の設定に関わるドキュメント設定命令部分、1202は文字の描画命令部分、1203は図形の描画命令部分を示している。

【0120】

各描画命令部分の詳細について説明する。

【0121】

ドキュメント設定命令部分1201において、C1~C5は、ドキュメント全体に関係するコマンドである。従って、これらのコマンドC1~C5は、ドキュメント1部について1ヶ所しか付いていない。

【0122】

これらドキュメントデータ全体に関係するコマンドには、例えば、キャラクタセットコマンド(フォント指定コマンド)、スケラブルフォントコマンド(スケラブルフォントを使用するか否かを指定するコマンド)、ハードリセットコマンド(以前のプリンタ使用環境をリセットするコマンド)等がある。

【0123】

ここで、C1はドキュメント設定開始コマンドである。C2はドキュメントデータの出力用紙サイズを示すコマンドであり、この場合には、A4の設定になっている。C3はドキュメントデータの方向を示すコマンドである。ここでは、ポートレートとランドスケープがあるが、この場合にはポートレート(PORT)の設定になっている。

【0124】

C 4 はドキュメントデータのタイプを示すコマンドであり、ページベクタで構成されるドキュメントデータであるかタイルベクタで構成されるドキュメントデータであるかを示している。この場合には、ページ (P A G E) に設定されている。C 5 はドキュメント設定終了コマンドである。

【 0 1 2 5 】

文字の描画命令部分 1 2 0 2 及び図形の描画命令部分 1 2 0 3 を構成する C 6 ~ C 2 2 は、ドキュメントデータを出力するための各種コマンドである。

【 0 1 2 6 】

C 6 はページの開始を示すコマンドである。C 7 は文字のフォントの種類を選択するためのコマンドであり、この場合には「 1 」という番号の付けられたフォントセットに設定されている。C 8 はフォントの大きさを設定するコマンドであり、この場合には「 1 0 ポイント」の大きさに設定されている。

【 0 1 2 7 】

C 9 は文字の色を設定するコマンドであり、順に R (赤) , G (緑) , B (青) 各色成分の輝度を示してある。この輝度は、例えば、0 から 2 5 5 の 2 5 6 段階で量子化されているものとする。この場合は、{ 0 , 0 , 0 } に設定されている。C 1 0 は文字を描画する開始位置の座標を示すコマンドである。座標位置 (X , Y) は、ページの左上を原点に指定する。この場合は、ページの { 1 0 , 5 } の位置から文字の描画を開始するように設定されている。C 1 1 は実際に描画する文字列 (A) を示すコマンドである。

【 0 1 2 8 】

C 1 2 は図形描画の際の面の塗りつぶしの色を示すコマンドである。色の指定は、文字の色と同様である。C 1 3 は図形描画の線の色を指定するコマンドである。C 1 4 は図形描画位置の座標を示すコマンドである。

【 0 1 2 9 】

C 1 5 は円弧を描画する際の半径を指定するコマンドであり、この場合、「 1 0 」座標単位を示している。C 1 6 は閉円弧の描画をするコマンドである。コマンド内の 2 個のパラメータは円弧を描画する際の描画開始角度と終了角度を示している。垂直情報を 0 度として、この場合には 0 度から 9 0 度の円弧を描画することを示している。

【 0 1 3 0 】

C 1 7 ~ C 2 1 は、C 1 2 ~ C 1 6 までのコマンドと同様に、図形描画の際の面、線の色指定、位置の指定等のコマンドである。C 2 2 はページの終了を示すコマンドである。

【 0 1 3 1 】

一方、ドキュメントデータ 1 1 0 1 がタイルベクタデータである場合について、図 1 1 を用いて説明する。

【 0 1 3 2 】

図 1 1 は本発明の実施形態 1 のタイルベクタデータの例を示す図である。

【 0 1 3 3 】

図 1 1 では、図 9 のドキュメントデータ 1 1 0 1 (ページベクタデータ) を、図 1 1 に示すようなブロック (タイル) 単位で分割したタイルベクタデータ (ドキュメントデータ 1 3 0 0) の一例を示している。

【 0 1 3 4 】

図 1 1 では、ドキュメントデータ 1 3 0 0 の短手方向を「 X 」方向、長手方向を「 Y 」とするデバイス座標系を定義する。

【 0 1 3 5 】

ここで、このドキュメントデータ 1 3 0 0 が、タイルベクタデータである場合、その内容を構成する記述例について、図 1 2 を用いて説明する。

【 0 1 3 6 】

図 1 2 は本発明の実施形態 1 のタイルベクタデータの記述例を示す図である。

【 0 1 3 7 】

図12において、1401はドキュメントデータ全体の設定に関わるドキュメント設定命令部分、1402は描画命令部分全体、1403～1406はそれぞれタイル1301, 1303, 1308, 1309の描画命令部分を示している。

【0138】

各描画命令の詳細について説明する。

【0139】

ドキュメント設定命令部分1401において、C1～C5は、ドキュメント全体に関するコマンドである。従って、これらのコマンドC1～C5は、ドキュメント1部について1ヶ所しか付いていない。

【0140】

これらドキュメントデータ全体に関係するコマンドには、例えば、キャラクタセットコマンド(フォント指定コマンド)、スケーラブルフォントコマンド(スケーラブルフォントを使用するか否かを指定するコマンド)、ハードリセットコマンド(以前のプリンタ使用環境をリセットするコマンド)等がある。

【0141】

ここで、C1はドキュメント設定開始コマンドである。C2はドキュメントデータの出力用紙サイズを示すコマンドであり、この場合には、A4の設定になっている。C3はドキュメントデータの方向を示すコマンドである。ここでは、ポートレートとランドスケープがあるが、この場合にはポートレート(PORT)の設定になっている。

【0142】

C4はドキュメントデータのタイプを示すコマンドであり、ページベクタで構成されるドキュメントデータあるのかタイルベクタで構成されるドキュメントデータあるかを示している。この場合には、タイル(TILE)に設定されている。C5はドキュメント設定終了コマンドである。

【0143】

描画命令部分1402を構成するC6～C500は、ドキュメントデータを出力するための各種コマンドである。

【0144】

C6はページの開始を示すコマンドである。C7は図11のタイル1301の描画コマンドの開始を示すコマンドである。ここで、TileStart(0,0)中の2個のパラメータは、ドキュメントデータ内におけるタイルの先頭位置(例えば、タイル左上角の位置)の情報を示している。C8はタイル1301の描画コマンドの終了を示すコマンドである。TileEnd(100,20)中の2個のパラメータは、ドキュメントデータ内におけるタイルの終端位置(例えば、タイル右下角の位置)の情報を示している。タイル1301のように、そのタイル内にオブジェクトが何も存在しない場合は、タイルの開始と終了を示すコマンドだけが記述される。

【0145】

C9は図11のタイル1303の描画コマンドの開始を示すコマンドである。C10は文字のフォントの種類を選択するためのコマンドであり、この場合には「1」という番号の付けられたフォントセットに設定されている。C11はフォントの大きさを設定するコマンドであり、この場合には「10ポイント」の大きさに設定されている。

【0146】

C12は文字の色を設定するコマンドであり、順にR(赤), G(緑), B(青)各色成分の輝度を示してある。この輝度は、例えば、0から255の256段階で量子化されているものとする。この場合は、{0,0,0}に設定されている。C13は文字を描画する開始位置の座標を示すコマンドである。座標位置(X,Y)は、タイルの左上を原点に指定する。この場合は、タイルの{0,5}の位置から文字の描画を開始するように設定されている。C14は実際に描画する文字列(A)を示すコマンドである。C15はタイル1303の描画コマンドの終了を示すコマンドである。

【0147】

10

20

30

40

50

C 1 0 0 は図 1 1 のタイル 1 3 0 8 の描画コマンドの開始を示すコマンドである。C 1 0 1 は図形描画の際の面の塗りつぶしの色を示すコマンドである。色の指定は、文字の色と同様である。C 1 0 2 は図形描画の線の色を指定するコマンドである。C 1 0 3 は図形を描画する位置の座標を示すコマンドである。

【 0 1 4 8 】

C 1 0 4 は円弧を描画する際の半径を指定するコマンドであり、この場合、「 1 0 」座標単位を表している。C 1 0 5 は閉円弧の描画をするコマンドである。コマンド内の 2 個のパラメータは円弧を描画する際の描画開始角度と終了角度を示している。垂直情報を 0 度として、この場合には 0 度から 9 0 度の円弧を描画することを示している。C 1 0 6 はタイル 1 3 0 8 の描画コマンドの終了を示すコマンドである。

10

【 0 1 4 9 】

C 1 2 0 は図 1 1 のタイル 1 3 0 9 の描画コマンドの開始を示すコマンドである。C 1 2 1 ~ C 1 2 5 は、C 1 0 0 ~ C 1 0 5 のコマンドと同様に図形の描画命令による図形の面、線の色、位置等を指定するコマンドである。C 1 2 6 はタイル 1 3 0 9 の描画コマンドの終了を示すコマンドである。

【 0 1 5 0 】

C 5 0 0 はページの終了を示すコマンドである。

【 0 1 5 1 】

次に、タイル / ページベクタ変換部 1 3 の処理の詳細について、図 1 3 を用いて説明する。

20

【 0 1 5 2 】

図 1 3 は本発明の実施形態 1 のタイル / ページベクタ変換部が実行する処理を示すフローチャートである。

【 0 1 5 3 】

尚、このタイル / ページベクタ変換部 1 3 では、ページベクタデータとタイルベクタデータ間での相互変換を行うことができる。あるいは、タイル / ページベクタ変換部 1 3 は、ページベクタデータからタイルベクタデータへ変換する変換部と、タイルベクタデータからページベクタデータへ変換する変換部で構成されていても良い。

【 0 1 5 4 】

(ステップ S 9 0 1)

30

まず、システムメモリ 5 中に記憶されたドキュメントデータ (ベクタデータ) から、ヘッダ部分に相当するコマンド列を読み込み、処理対象となるドキュメントデータ全体に関するコマンド部分を解析する。具体的には、図 1 0 または図 1 2 の C 1 ~ C 5 に相当する部分の内容を解析する。

【 0 1 5 5 】

(ステップ S 9 0 2)

解析の結果に基づいて、ドキュメントデータのタイプが、ページベクタデータであるかを判定する。ページベクタデータである場合 (ステップ S 9 0 2 で Y E S)、ステップ S 9 0 3 以降のステップに進み、ページベクタ タイルベクタ変換を実行する。一方、ページベクタデータでない場合、つまり、タイルベクタデータである場合 (ステップ S 9 0 2 で N O)、ステップ S 9 1 0 以降のステップに進み、タイルベクタ ページベクタ変換を実行する。

40

【 0 1 5 6 】

(ステップ S 9 0 3)

ページベクタデータから、オブジェクトを記述するコマンド列を読み込む。

【 0 1 5 7 】

(ステップ S 9 0 4)

ステップ S 9 0 3 で読み込んだコマンド列を解析し、記述されているオブジェクトの大きさがタイルの上限サイズ (所定サイズ) を超えているかを判定する。つまり、オブジェクトの更なる分割が必要であるかを判定する。

50

【 0 1 5 8 】

オブジェクトの大きさがタイルの上限サイズを超えていない場合（ステップ S 9 0 4 で N O）、ステップ S 9 0 5 をスキップして、ステップ S 9 0 6 に進む。一方、オブジェクトの大きさがタイルの上限サイズを超えている場合（ステップ S 9 0 4 で Y E S）、ステップ S 9 0 5 に進む。

【 0 1 5 9 】

（ステップ S 9 0 5）

ここでは、入力されたオブジェクトの分割処理を実行する。

【 0 1 6 0 】

例えば、図 1 0 のページベクタデータにおいて、図形の描画命令部分 1 2 0 3 のコマンド C 1 7 ~ C 2 1 で記述されている図形が、タイルの上限サイズを超えている場合は、タイルの上限サイズを超えないように、図 1 1 に示すようなタイル 1 3 0 7、1 3 0 8、1 3 0 9、1 3 1 0 に分割する。図形の分割は、まず、元の図形オブジェクトを内包するタイルが一旦設定され、次に、このタイルを 4 つの均等サイズで分割したタイル 1 3 0 7、1 3 0 8、1 3 0 9、1 3 1 0 に設定する。

10

【 0 1 6 1 】

（ステップ S 9 0 6）

入力されたオブジェクトのコマンドの記述に対し、タイルベクタデータ内での描画位置へ変更するために、座標位置の変換を行う。ページベクタデータでは、ページの左上からの位置を記述していたのに対し、タイルベクタデータではタイルの左上からの位置に記述し直す。描画位置をタイル内の座標で記述することにより、座標計算に要するデータ長を削減することが可能になる。

20

【 0 1 6 2 】

また、図 1 2 のコマンド C 9、C 1 5 等で示されるように、個々のタイルに関して、ページの左上からのタイルの先頭と、タイルの終端位置の座標が計算され、そのタイルがページ上のどこに配置されるかがコマンドから判定できるようになっている。

【 0 1 6 3 】

（ステップ S 9 0 7）

1 つのオブジェクトに対するコマンドの記述変換が終了したら、ページ内の全てのオブジェクトのコマンドの記述変換が終了したか否かを判定する。終了していない場合（ステップ S 9 0 7 で N O）、ステップ S 9 0 3 に戻り、次のコマンドに対して、ステップ S 9 0 3 ~ ステップ S 9 0 7 の処理を繰り返す。一方、終了した場合（ステップ S 9 0 7 で Y E S）、ステップ S 9 0 8 へ進む。

30

【 0 1 6 4 】

尚、オブジェクトが存在していない場所、例えば、図 1 1 におけるタイル 1 3 0 1、1 3 0 2、1 3 0 4、1 3 0 5、1 3 0 6、1 3 1 1、1 3 1 2 等の領域に対しては、個々の矩形領域に対して、図 1 2 のコマンド C 7、C 8 等で示されるタイルの先頭位置と終端位置の情報を有するコマンドを生成した上で、ステップ S 9 0 8 へ進む。

【 0 1 6 5 】

（ステップ S 9 0 8）

全てのコマンドの記述変換が終了したら、図 1 1 のように分割されたタイルベクタデータ中の各タイルに対し、ページの左上から順にタイルベクタデータとしてシステムメモリ 5 への書出を実行する。タイルベクタデータとしては、ステップ S 9 0 5 及びステップ S 9 0 6 で記述されたコマンドに対し、タイルの開始と終了を示すコマンドを追加するようなフォーマットで記述される。

40

【 0 1 6 6 】

次に、ステップ S 9 0 3 ~ ステップ S 9 0 7 で処理されたコマンドが存在する座標のタイルに対し、オブジェクトの記述を追加する。例えば、図 1 1 のタイル 1 3 0 3 の場合は、コマンド C 9 ~ C 1 5 から構成される描画命令部分 1 4 0 4（図 1 2）で記述される。

【 0 1 6 7 】

50

(ステップS909)

1つのオブジェクトのタイルベクタへの書出が終了すると、そのページのオブジェクトの記述が全て終了したか否かを判定する。終了していない場合(ステップS909でNO)、ステップS903に戻る。一方、終了した場合(ステップS909でYES)、処理を終了する。

【0168】

一方、ステップS902で、ドキュメントデータのタイプが、タイルベクタデータである場合について説明する。

【0169】

(ステップS910)

タイルベクタデータから、オブジェクトを記述するコマンド列を読み込む。

【0170】

(ステップS911)

ステップS910で読み込んだコマンド列を解析し、記述されているオブジェクトがそれより以前に読み込まれたタイルと結合可能であるか否かを判定する。結合可能でない場合(ステップS911でNO)、ステップS912をスキップし、ステップS913へ進む。一方、結合可能である場合(ステップS911でYES)、ステップS912へ進む。

【0171】

尚、オブジェクトが結合可能であるか否かの判定は、読み込んだコマンドの座標位置や、図形の種別等を基に判定する。また、文字列の場合は、フォントサイズやフォントの種類を基に判定する。

【0172】

(ステップS912)

オブジェクトの結合処理を実行する。この処理は、基本的には、ステップS905の処理手順を逆にすることで実現する。

【0173】

(ステップS913)

入力されたオブジェクトのコマンドの記述に対し、ページベクタデータ内での描画位置へ変更するために、座標位置の変換を行う。タイルベクタデータでは、タイルの左上からの位置を記述していたのに対し、ページベクタデータでは、ページの左上からの位置に記述し直す。

【0174】

(ステップS914)

1つのオブジェクトに対するコマンドの記述変換が終了したら、タイル内の全てのオブジェクトのコマンドの記述変換が終了したか否かを判定する。終了していない場合(ステップS914でNO)、ステップS910に戻り、次のコマンドに対して、ステップS910～ステップS913の処理を繰り返す。一方、終了した場合(ステップS914でYES)、ステップS915へ進む。

【0175】

(ステップS915)

全てのコマンドの記述変換が終了したら、ページベクタデータとしてシステムメモリ5への書出を実行する。ページベクタデータとしては、ステップS912及びステップS913で記述されたコマンドに対し、タイルの開始と終了を示すコマンドを削除したようなフォーマットで記述される。

【0176】

まず、ページ内の一番最初のタイルに記述されたコマンドの書出時点では、システムメモリ5内にオブジェクトがない状態のページベクタデータを生成しておく。これは、図10の記述を用いて説明すると、コマンドC1～C6、コマンドC22だけで記述されるページベクタデータである。

10

20

30

40

50

【 0 1 7 7 】

(ステップ S 9 1 6)

1 つコマンドのページベクタへの書出が終了すると、そのタイルのオブジェクトの記述が全て終了したか否かを判定する。終了していない場合 (ステップ S 9 1 6 で N O)、ステップ S 9 1 0 に戻る。一方、終了した場合 (ステップ S 9 1 6 で Y E S)、ステップ S 9 1 7 に進む。

【 0 1 7 8 】

(ステップ S 9 1 7)

1 つのタイルベクタデータの書出が終了すると、そのページのタイルベクタデータの記述に対する処理が全て終了したか否かを判定する。終了していない場合 (ステップ S 9 1 7 で N O)、ステップ S 9 1 0 に戻る。一方、終了した場合 (ステップ S 9 1 7 で Y E S)、処理を終了する。

10

【 0 1 7 9 】

[画像データ展開部 (R I P)]

次に、コントローラ 1 中の画像データ展開部 1 8 の詳細について説明する。

【 0 1 8 0 】

まず、コピーやプリント、送信等の画像データに対する処理を開始する前に、ローカルメモリ 1 9 の初期化と、作成するオブジェクトの解像度の設定を行っておく。実施形態 1 では、生成解像度は 6 0 0 d p i とし、ポイントサイズや m m 等の単位系で指定された印刷コマンドは、この値を用いてドット数に変換されることになる。

20

【 0 1 8 1 】

以下では、画像データ展開部 1 8 によって、タイルベクタデータ展開を実行する場合の処理について、図 1 4 を用いて説明する。

【 0 1 8 2 】

図 1 4 は本発明の実施形態 1 の画像データ展開部が実行する処理を示すフローチャートである。

【 0 1 8 3 】

(ステップ S 1 0 1)

システムメモリ 5 より、S B B 2 を経由して、R I P 1 8 に一定サイズ分入力されたタイルベクタデータは、ローカルメモリ 1 9 のタイルベクタ領域に一時的に格納する。

30

【 0 1 8 4 】

(ステップ S 1 0 2)

タイルベクタデータをローカルメモリ 1 9 に格納したら、R I P 1 8 内の μ R I P 1 8 a ~ 1 8 d のいずれかが、タイルベクタデータを展開可能 (処理可能) であるか否かを判定する。 μ R I P 1 8 a ~ 1 8 d のいずれもタイルベクタデータの展開 (処理) 中である場合 (ステップ S 1 0 2 で N O)、それらのいずれかが展開可能になるまで待機する。

【 0 1 8 5 】

(ステップ S 1 0 3)

μ R I P 1 8 a ~ 1 8 d のいずれかがタイルベクタデータを展開可能になれば、予め定められた文法に従って、ローカルメモリ 1 9 に格納されたタイルベクタデータのコマンド解析を実行する。

40

【 0 1 8 6 】

(ステップ S 1 0 4)

コマンド解析結果に基づいて、コマンドが描画命令であるか排紙命令であるかを判定する。描画命令である場合 (ステップ S 1 0 4 で Y E S)、ステップ S 1 0 5 に進む。一方、排紙命令である場合 (ステップ S 1 0 4 で N O)、ステップ S 1 0 6 に進む。

【 0 1 8 7 】

(ステップ S 1 0 5)

コマンドが描画命令である場合、描画オブジェクト (D L データ) の生成を実行する。タイルベクタデータ内のコマンドが文字の描画命令である場合、そのコマンドで指定され

50

るフォントの書体や、文字サイズ、文字コードを元にフォントオブジェクトを生成して、ローカルメモリ19のDLデータ領域に格納する。また、文字以外の描画命令、つまり、図形の描画命令である場合、そのコマンドで指定された図形（ラインや円、多角形等）の描画オブジェクトを生成し、ローカルメモリ19のDLデータ領域に格納する。

【0188】

また、コマンドが描画命令で指定されない印刷データである場合には、この印刷データに応じて印刷位置移動や印刷環境設定等の印刷制御処理を実行し、一単位分のコマンド解釈を終了する。

【0189】

これは、タイルベクタデータ内のコマンド全ての解釈が終了するまで、上記処理を繰り返す。

【0190】

（ステップS106）

コマンドが排紙命令である場合、μRIPは、ローカルメモリ19上のタイルラスト領域に空き領域があるか否かを判定する。空き領域がない場合（ステップS106でNO）、他のμRIPの処理が終了して、タイルラスト領域が解放されて、空き領域ができるまで待機する。一方、空き領域がある場合（ステップS106でYES）、ステップS107に進む。

【0191】

（ステップS107）

タイルラスト領域に空き領域がある場合、ステップS105で生成された描画オブジェクトを読み出し、タイルラスト領域に描画（ラスタライズ）する。この時、生成解像度が600dpiであれば、タイルラスト領域には600dpiの画像としてラスタライズされる。そして、描画が終了したタイルラスト画像は、SBB2を介して、画像処理部15に出力する。

【0192】

（ステップS108）

ステップS105またはステップS107で一つのタイルベクタデータに対する、コマンド解析または描画処理が終了したら、タイルベクタ領域に格納したタイルベクタデータのすべてに対して処理が終了したか否かを判定する。未処理のタイルベクタデータがある場合（ステップS108でNO）、ステップS102に戻り、次のタイルベクタデータの処理を続ける。一方、未処理のタイルベクタデータがない場合（ステップS108でYES）、ステップS109に進む。

【0193】

（ステップS109）

1ページ分のタイルベクタデータに対し、全ての処理が終了したか否かを判定する。未処理のタイルベクタデータがある場合（ステップS109でNO）、ステップS101に戻り、システムメモリ5から、タイルベクタデータを読み出して、処理を続ける。一方、未処理のタイルベクタデータがない場合（ステップS109でYES）、処理を終了する。

【0194】

以上説明したように、実施形態1によれば、システムメモリ上に、入力された画像データをページベクタデータとタイルベクタデータの2種類の形式の画像データのみを記憶する構成を採用する。これにより、画像サイズの大きいラスタデータ及びDLデータをシステムメモリに記憶する必要がなくなる。そのため、システムメモリ上で確保しなければならない画像データ領域を削減することができる。

【0195】

また、タイルベクタデータ生成時には、固定サイズのタイルではなく、オブジェクトのサイズに適合させた可変サイズのタイルベクタデータを生成する構成とすることで、より適切なサイズのタイルベクタデータを生成することができる。これにより、タイルベクタ

10

20

30

40

50

データ全体のデータ量を削減することができる。

【 0 1 9 6 】

更に、個々のタイルベクタデータのタイルサイズが大きくなりすぎた場合に、並列処理性が低下することを防ぐために、所定のタイルサイズよりも大きいタイルに関しては、そのタイルを再分割して、新たなタイルを設定する。これにより、並列処理の効率を向上させることが可能となる。

【 0 1 9 7 】

[実施形態 2]

実施形態 1 では、ラスタベクタデータからタイルベクタデータへ変換する場合、または、ページベクタデータをタイルベクタデータへ変換する場合は、オブジェクトが存在しないタイル（空白タイル）であっても、図 1 2 のコマンド C 7、C 8 で示されるタイルの先頭位置と終端位置の情報だけをもつコマンドを生成している。

【 0 1 9 8 】

これに対して、実施形態 2 では、図 1 5 に示すように、空白タイルに関しては、ベクタデータを生成しないで（つまり、ベクタデータの生成を禁止して）、オブジェクトが存在する場合だけ、タイルベクタデータを生成するようにしても良い。

【 0 1 9 9 】

つまり、図 1 5 では、図 1 2 の描画命令部分 1 4 0 3 に対応する描画命令部分が省略され、図 1 2 の描画命令部分 1 4 0 1、1 4 0 4、1 4 0 5、1 4 0 6 に対応する描画命令部分 1 1 0 1、1 1 0 4、1 1 0 5、1 1 0 6 のみが記述されている。

【 0 2 0 0 】

このように、タイルベクタデータ中に、空白タイルの情報を省略することにより、MFP 内部で扱うデータ量をより削減することが可能となる。

【 0 2 0 1 】

尚、実施形態 2 の構成の場合は、図 1 2 のフローチャートにおいて、ステップ 9 0 7 の処理において、空白タイルに関しての処理を省略することになる。

【 0 2 0 2 】

以上説明したように、実施形態 2 によれば、実施形態 1 で説明した効果に加えて、タイルベクタデータのデータ量をより削減することができる。

【 0 2 0 3 】

[実施形態 3]

実施形態 3 は、実施形態 1 の応用例である。

【 0 2 0 4 】

特に、実施形態 3 では、画像データ中で複数のオブジェクトそれぞれを内包する矩形領域（タイル）を一旦設定した後、その矩形領域同士の重複部の割合に基づいて、より適切なタイルを再設定（結合 / 分割）する構成について説明する。

【 0 2 0 5 】

まず、本発明の実施形態 3 の画像処理システムでのコピー動作に係るデータフローについて、図 2 を用いて説明する。

【 0 2 0 6 】

尚、ここでは、実施形態 1 のコピー動作における処理内容の内、異なる処理内容の部分だけ説明する。特に、実施形態 3 では、実施形態 1 における（S 2 2）の処理内容が異なる。

【 0 2 0 7 】

[コピー]

（S 2 2）：画像処理部 1 5 による画像処理が終了し、画像処理部 1 5 から出力された画像データはラスタ / タイルベクタ変換部 1 4 に入力され、タイルベクタ変換処理が実行される。

【 0 2 0 8 】

ここでは、まず、ラスタ / タイルベクタ変換部 1 4 は、画像データ中のオブジェクトを

10

20

30

40

50

内包する矩形領域を一旦設定する。そして、画像データ中で複数のオブジェクトそれぞれを内包する矩形領域が重複していない場合は、オブジェクト毎に、この矩形領域の大きさのブロック（タイル）を確定して、その大きさのブロックに画像データ中のオブジェクトを分割する。

【0209】

一方、複数のオブジェクトそれぞれを内包する矩形領域が重複している場合で、特に、その重複している割合（重複度）が所定値より大きい場合は、その複数のオブジェクトを内包する矩形領域の大きさのブロック（タイル）を設定して、その大きさのブロックに画像データ中の複数のオブジェクトを分割する。一方、重複度が所定値より小さい場合は、各オブジェクトを内包する矩形領域の大きさのブロック（タイル）を設定して、その大きさのブロックに画像データ中の複数のオブジェクトの各オブジェクトを分割する。

10

【0210】

そして、各ブロック内のラスタデータに対して、ベクトル化処理を実行して、ブロック（タイル）単位のベクタデータ（タイルベクタデータ（ブロックベクタデータ））を生成する。

【0211】

生成されたタイルベクタデータは、SBB2によりバスの調停を受け、システムメモリ5へのバス権を取得し、MC4を介して、システムメモリ5に記憶される。（尚、SBB2経由でデータバスが接続される場合は、基本的にバスの調停を受け、バス権を取得する手続きを踏むが、以降のフロー説明では省略する。）

20

次に、本発明の実施形態3の画像処理システムでのプリント動作に係るデータフローについて、図3を用いて説明する。

【0212】

尚、ここでは、実施形態1のコピー動作における処理内容の内、異なる処理内容の部分だけ説明する。特に、実施形態3では、実施形態1における（S32）の処理内容が異なる。

【0213】

〔プリント〕

（S32）：システムメモリ5に記憶されたページベクタデータは、タイル／ページベクタ変換部13より読み出され、タイルベクタ変換処理が実行される。

30

【0214】

ここでは、まず、タイル／ページベクタ変換部13は、ページベクタデータ内に存在するオブジェクト毎に、オブジェクトを内包する矩形領域を一旦設定する。そして、ページベクタデータ中で、複数のオブジェクトが重複していない場合は、オブジェクト毎に、この矩形領域の大きさのブロック（タイル）を確定して、その大きさのブロックにページベクタデータ中のオブジェクトを分割する。

【0215】

一方、複数のオブジェクトが重複している場合で、特に、その重複している割合（重複度）が所定値より大きい場合は、その複数のオブジェクトを内包する矩形領域の大きさのブロック（タイル）を設定して、その大きさのブロックにページベクタデータ中の複数のオブジェクトを分割する。一方、重複度が所定値より小さい場合は、各オブジェクトを内包する矩形領域の大きさのブロック（タイル）を設定して、その大きさのブロックにページベクタデータ中の複数のオブジェクトの各オブジェクトを分割する。

40

【0216】

〔ラスタ／タイルベクタ変換部〕

次に、ラスタ／タイルベクタ変換部14の処理の詳細について説明する。

【0217】

図16は本発明の実施形態3のラスタ／タイルベクタ変換部が実行する処理を示すフローチャートである。

【0218】

50

尚、図 16 において、実施形態 1 の図 5 のフローチャートと同一の処理については、同一のステップ番号を付加して、その詳細については省略する。

【0219】

(ステップ S52a : タイル重複部算出処理)

分離された各領域は、次に、領域間の重複部の割合(重複度)を算出する。

【0220】

具体例について、図 17 を用いて説明する。

【0221】

図 17 は本発明の実施形態 3 のタイル重複部算出処理の具体例を説明するための図である。

10

【0222】

図 17 に示す、ラスタデータ 1600 では、オブジェクト 1601 及び 1602 の 2 つのオブジェクトが存在している。このラスタデータ 1600 に対して、ステップ S51 のブロックセレクションを実行すると、オブジェクト 1601 は文字領域として認識され、オブジェクト 1602 は図形領域として認識される。

【0223】

この処理結果に対して、タイル重複部算出処理を実行すると、ラスタデータ 1600 中のオブジェクト 1601 及び 1602 は、例えば、図 18 に示すように、それぞれのオブジェクトを含む矩形領域 1701 及び 1702 が設定される。

【0224】

20

次に、矩形領域 1701 及び 1702 の重複部 1703 の面積を算出する。このとき、矩形領域 1701 及び 1702 それぞれの面積に対する、重複部 1703 の面積の割合を算出する。

【0225】

(ステップ S52b : タイル重複部割合判定処理)

次に、ステップ S52a で算出した重複部 1703 の面積の割合が所定値より大きいか否かを判定する。所定値未満である場合(ステップ S52b で NO)、ステップ S52c へ進む。一方、所定値未満より大きい場合(ステップ S52b で YES)、ステップ S52d へ進む。

【0226】

30

(ステップ S52c : タイル分割処理)

重複部 1703 の面積の割合が所定値未満である場合、その重複部を形成する矩形領域(タイル) 1701 及び 1702 を含む領域(タイル)を分割し直すためのタイル分割処理を実行する。

【0227】

このタイル分割処理によって、矩形領域(タイル) 1701 及び 1702 を含むページは、例えば、図 19 の領域(タイル) 1801 ~ 1807 に分割する。この場合、オブジェクト 1602 をすべて包含する領域 1805 を生成し、その中にはオブジェクト 1601 の一部が含まれることとなる。一方、領域 1804 では、オブジェクト 1601 の一部のみが含まれる。

40

【0228】

このように、オブジェクトを内包する矩形領域同士の重複部の割合が所定値未満である場合、タイル分割処理を実行して、より適切なサイズのタイルを設定して、タイルの並列処理性の向上を図る。

【0229】

(ステップ S52d : タイル結合処理)

重複部 1703 の面積の割合が所定値より大きい場合、その重複部を形成する矩形領域(タイル) 1701 及び 1702 を含む領域に結合するタイル結合処理を実行する。

【0230】

例えば、図 20 に示すように、オブジェクトを包含する矩形領域(タイル) 1701 及

50

び 1702 がほぼ重複している場合（重複部の割合が比較的大きい場合）、つまり、両領域の重複部 1901 の割合が所定値 より大きい場合、矩形領域（タイル）1701 及び 1702 を含むページ上では、図 21 の領域（タイル）1001～1005 を設定する。この場合、オブジェクト 1601 及び 1602 の両方を包含する領域（タイル）1003 が設定される。

【0231】

このように、オブジェクトの内包する矩形領域同士の重複部の割合が所定値 より大きい場合、タイルの並列処理による高速化が見込めない。そのため、このような場合には、ページを構成するタイルの量を削減し、ページ全体のデータ量を減らすために、複数のオブジェクトを包含するタイルを生成（設定）する。

10

【0232】

[タイル/ページベクタ変換部]

次に、タイル/ページベクタ変換部 13 の処理の詳細を説明するにあたり、処理対象となるドキュメントデータ（画像データ）について説明する。

【0233】

図 22 は本発明の実施形態 3 のネットワークから転送されてくるドキュメントデータの一例を示している。

【0234】

図 22 では、ドキュメントデータ 2201 の短手方向を「X」方向、長手方向を「Y」方向とするデバイス座標系を定義する。ドキュメントデータ 2201 としては、ページベクタデータ、タイルベクタデータ、あるいはラスタデータ表現を含むページベクタデータ（タイルベクタデータ）、あるいはラスタデータのいずれかで構成される可能性がある。

20

ここで、ドキュメントデータ 2201 が、ページベクタデータである場合、その内容を構成する記述例について、図 23 を用いて説明する。

【0235】

図 23 は本発明の実施形態 3 のページベクタデータの記述例を示す図である。

【0236】

図 23 において、2301 はドキュメントデータ全体の設定に関わるドキュメント設定命令部分、2302 は文字の描画命令部分、2303 は図形の描画命令部分を示している。

30

【0237】

各描画命令部分の詳細について説明する。

【0238】

ドキュメント設定命令部分 2301 において、C1～C5 は、ドキュメント全体に関するコマンドである。従って、これらのコマンド C1～C5 は、ドキュメント 1 部について 1 ケ所しか付いていない。

【0239】

これらドキュメントデータ全体に関係するコマンドには、実施形態 1 の図 10 で詳述した内容と同様である。

【0240】

文字の描画命令部分 2302 及び図形の描画命令部分 2303 を構成する C6～C22 は、ドキュメントデータを出力するための各種コマンドである。

40

【0241】

尚、文字の描画命令部分 2302 及び図形の描画命令部分 2303 を構成するコマンドについては、それぞれ文字及び図形の描画位置が異なる以外は、実施形態 1 の図 10 の描画命令部分 1202 及び図形の描画命令部分 1203 で詳述した内容と同様である。

【0242】

一方、ドキュメントデータ 2201 がタイルベクタデータである場合について、図 24 を用いて説明する。

【0243】

50

図 2 4 は本発明の実施形態 3 のタイルベクタデータの例を示す図である。

【 0 2 4 4 】

図 2 4 では、図 2 2 のドキュメントデータ 2 2 0 1 (ページベクタデータ) を、図 2 4 に示すようなブロック (タイル) 単位で分割したタイルベクタデータ (ドキュメントデータ 2 4 0 0) の一例を示している。

【 0 2 4 5 】

図 2 4 では、ドキュメントデータ 2 4 0 0 の短手方向を「 X 」方向、長手方向を「 Y 」とするデバイス座標系を定義する。

【 0 2 4 6 】

ここで、このドキュメントデータ 2 4 0 0 が、タイルベクタデータである場合、その内容を構成する記述例について、図 2 5 を用いて説明する。

10

【 0 2 4 7 】

図 2 5 は本発明の実施形態 3 のタイルベクタデータの記述例を示す図である。

【 0 2 4 8 】

図 2 5 において、2 5 0 1 はドキュメントデータ全体の設定に関わるドキュメント設定命令部分、2 5 0 2 は描画命令部分全体、2 5 0 3 ~ 2 5 0 5 はそれぞれタイル 2 4 0 1 , 2 4 0 4 , 2 4 0 5 の描画命令部分を示している。

【 0 2 4 9 】

各描画命令の詳細について説明する。

【 0 2 5 0 】

20

ドキュメント設定命令部分 2 5 0 1 において、C 1 ~ C 5 は、ドキュメント全体に関するコマンドである。従って、これらのコマンド C 1 ~ C 5 は、ドキュメント 1 部について 1 ケ所しか付いていない。

【 0 2 5 1 】

これらドキュメントデータ全体に関係するコマンドには、実施形態 1 の図 1 2 で詳述した内容と同様である。

【 0 2 5 2 】

描画命令部分 2 5 0 2 を構成する C 6 ~ C 5 0 0 は、ドキュメントデータを出力するための各種コマンドである。

【 0 2 5 3 】

30

ここで、描画命令部分 2 5 0 3 は、実施形態 1 の図 1 2 の描画命令部分 1 4 0 3 で詳述した内容と同様である。

【 0 2 5 4 】

C 9 は図 2 4 のタイル 2 4 0 4 の描画コマンドの開始を示すコマンドである。C 1 0 は文字のフォントの種類を選択するためのコマンドであり、この場合には「 1 」という番号の付けられたフォントセットに設定されている。C 1 1 はフォントの大きさを設定するコマンドであり、この場合には「 1 0 ポイント」の大きさに設定されている。

【 0 2 5 5 】

C 1 2 は文字の色を設定するコマンドであり、順に R (赤) , G (緑) , B (青) 各色成分の輝度を示してある。この輝度は、例えば、0 から 2 5 5 の 2 5 6 段階で量子化されているものとする。この場合は、{ 0 , 0 , 0 } に設定されている。C 1 3 は文字を描画する開始位置の座標を示すコマンドである。座標位置 (X , Y) は、ページの左上を原点に指定する。この場合は、ページの { 4 0 , 1 5 0 } の位置から文字の描画を開始するように設定されている。C 1 4 は実際に描画する文字列 (A) を示すコマンドである。C 1 5 はタイル 2 4 0 4 の描画コマンドの終了を示すコマンドである。

40

【 0 2 5 6 】

尚、実施形態 3 においては、このタイル 2 4 0 4 内に存在する文字を描画する開始位置も、タイル 2 4 0 4 を描画する開始位置と同様に、ページの左上を基準にしている。これは、文字列 (A) がタイル 2 4 0 4 とタイル 2 4 0 5 に跨っていて、各タイル内では、自身に含まれる文字列の部分のみを描画する必要があるからである。

50

【 0 2 5 7 】

つまり、タイル 2 4 0 4 では、描画コマンド C 9 と C 1 5 で示される矩形領域の位置情報と、文字列のページにおける位置情報とから、タイル 2 4 0 4 内部で描画されるべき範囲内のみの文字列 (A) を描画する必要がある。そのため、実施形態 3 では、タイル内に存在する文字を描画する開始位置も、ページの左上を基準にする座標系を採用している。

【 0 2 5 8 】

C 1 0 0 は図 2 4 のタイル 2 4 0 5 の描画コマンドの開始を示すコマンドである。C 1 0 1 は図形描画の際の面の塗りつぶしの色を示すコマンドである。色の指定は、文字の色と同様である。C 1 0 2 は図形描画の線の色を指定するコマンドである。C 1 0 3 は図形を描画する位置の座標を示すコマンドである。

10

【 0 2 5 9 】

C 1 0 4 は円弧を描画する際の半径を指定するコマンドであり、この場合、「 1 0 」座標単位を表している。C 1 0 5 は閉円弧の描画をするコマンドである。コマンド内の 2 個のパラメータは円弧を描画する際の描画開始角度と終了角度を示している。垂直情報を 0 度として、この場合には 0 度から 9 0 度の円弧を描画することを示している。

【 0 2 6 0 】

C 1 0 6 ~ C 1 1 0 は、C 1 0 1 ~ C 1 0 5 のコマンドと同様に図形の描画命令による図形の面、線の色指定、位置等を指定するコマンドである。

【 0 2 6 1 】

C 1 1 1 ~ C 1 1 4 は、タイル 2 4 0 4 で記述されている文字列 (A) の一部を、タイル 2 4 0 5 で描画するためのコマンドである。ここでは、特に、コマンド C 1 1 3 で設定される、文字列 (A) の一部を描画する開始位置の座標と、C 1 0 0 と C 1 1 5 で設定されるタイルの領域情報とから、文字列 (A) の一部を描画する。

20

【 0 2 6 2 】

C 1 1 5 はタイル 2 4 0 5 の描画コマンドの終了を示すコマンドである。また、C 5 0 0 はページの終了を示すコマンドである。

【 0 2 6 3 】

次に、タイル / ページベクタ変換部 1 3 の処理の詳細について、図 2 6 を用いて説明する。

【 0 2 6 4 】

図 2 6 は本発明の実施形態 3 のタイル / ページベクタ変換部が実行する処理を示すフローチャートである。

30

【 0 2 6 5 】

(ステップ S 1 1 0 1)

まず、システムメモリ 5 中に記憶されたドキュメントデータ (ベクタデータ) から、ヘッダ部分に相当するコマンド列を読み込み、処理対象となるドキュメントデータ全体に関するコマンド部分を解析する。具体的には、図 2 3 または図 2 5 の C 1 ~ C 5 に相当する部分の内容を解析する。

【 0 2 6 6 】

(ステップ S 1 1 0 2)

これは、図 1 3 のステップ S 9 0 2 に対応するので、その説明は省略する。

40

【 0 2 6 7 】

(ステップ S 1 1 0 3)

これは、図 1 3 のステップ S 9 0 3 に対応するので、その説明は省略する。

【 0 2 6 8 】

(ステップ S 1 1 0 4)

ステップ S 1 1 0 3 で読み込んだコマンド列を解析し、記述されているオブジェクトを内包する矩形領域を算出し、複数オブジェクト間で算出された矩形領域が重複しているか否かを判定する。つまり、オブジェクトを内包する矩形領域 (タイル) の結合を実行するか、あるいは更なる分割を実行するかを判定する。

50

【 0 2 6 9 】

矩形領域が重複している割合が所定値より大きい場合（ステップ S 1 1 0 4 で Y E S ）、ステップ S 1 1 0 5 へ進み、タイル結合処理を実行する。一方、矩形領域が重複している割合が所定値未満である場合（ステップ S 1 1 0 4 で N O ）、ステップ S 1 1 0 6 へ進み、タイル分割処理を実行する。

【 0 2 7 0 】

（ステップ S 1 1 0 5 ）

このステップでは、オブジェクトを内包するタイルの結合処理を行う。

【 0 2 7 1 】

例えば、図 2 3 のページベクタデータ中の文字の描画命令部分 2 3 0 2、図形の描画命令部分 2 3 0 3 において、両者によって描画される文字と図形それぞれを包含する領域 1 7 0 1 及び 1 7 0 2 の重複部 1 9 0 1 の割合が所定値より大きい場合（図 2 0 の状態の場合）、その領域（タイル）を再分割して並列処理したとしても、文字及び図形それぞれを包含する領域の重複部は分散処理しづらい。

10

【 0 2 7 2 】

つまり、このような場合には、分散処理による高速化が望めないので、ベクタデータ全体のデータ量を減らすために、図 2 1 で説明したように、文字及び図形それぞれを包含する領域 1 7 0 1 及び 1 7 0 2（タイル）を結合し、かつその結合領域を含み、文字及び図形の両方を包含する領域を生成（設定）する。

【 0 2 7 3 】

20

（ステップ S 1 1 0 6 ）

このステップでは、オブジェクトを内包するタイルの分割処理を行う。

【 0 2 7 4 】

例えば、図 2 3 のページベクタデータ中の文字の描画命令部分 2 3 0 2、図形の描画命令部分 2 3 0 3 において、両者によって描画される文字と図形それぞれを包含する領域 1 7 0 1 及び 1 7 0 2 の重複部 1 7 0 3 の割合が所定値未満である場合（図 1 8 の状態の場合）、その領域（タイル）を分割して、文字、図形を並列に処理したほうが高速化が行えると判定する。

【 0 2 7 5 】

つまり、このような場合には、図 2 4 に示すように、文字と図形それぞれを包含する領域 1 7 0 1 及び 1 7 0 2 をタイル 2 4 0 4 とタイル 2 4 0 5 の 2 つのタイルに分割する。この分割によって、タイル 2 4 0 5 では、片方のオブジェクト（図形オブジェクト）は全て内包し、もう一方のオブジェクト（文字オブジェクト）の一部が含まれる。また、タイル 2 4 0 4 のように、オブジェクトが複数のタイルに跨がる場合には、そのオブジェクトの一部がそのタイルに含まれるようになる。

30

【 0 2 7 6 】

（ステップ S 1 1 0 7 ）

これは、図 1 3 のステップ S 9 0 7 に対応するので、その説明は省略する。

【 0 2 7 7 】

尚、オブジェクトが存在していない場所、例えば、図 2 4 におけるタイル 2 4 0 1、2 4 0 2、2 4 0 3、2 4 0 6、2 4 0 7 等の領域に対しては、個々の矩形領域に対して、図 2 5 のコマンド C 7、C 8 等で示されるタイルの先頭位置と終端位置の情報を有するコマンドを生成した上で、ステップ S 1 1 0 8 へ進む。

40

【 0 2 7 8 】

（ステップ S 1 1 0 8 ）

これは、図 1 3 のステップ S 9 0 8 に対応するので、その説明は省略する。

【 0 2 7 9 】

次に、ステップ S 1 1 0 3 ～ステップ S 1 1 0 7 で処理されたコマンドが存在する座標のタイルに対し、オブジェクトの記述を追加する。例えば、図 2 4 のタイル 2 4 0 4 の場合は、コマンド C 9 ～ C 1 5 から構成される描画命令部分 2 5 0 4（図 2 5）で記述され

50

る。

【0280】

(ステップS1109)

これは、図13のステップS909に対応するので、その説明は省略する。

【0281】

一方、ステップS1102で、ドキュメントデータのタイプが、タイルベクタデータである場合について説明する。

【0282】

(ステップS1110)

これは、図13のステップS910に対応するので、その説明は省略する。

10

【0283】

(ステップS1111)

ステップS1110で読み込まれたコマンド列を解析し、記述されているオブジェクトがそれより以前に読み込まれたタイルと重複したデータを有するか否かを判定する。そして、この重複したデータの有無に応じて、そのタイルとオブジェクトが結合可能であるか否かを判定する。結合可能でない場合(ステップS1111でNO)、ステップS1112をスキップし、ステップS1114へ進む。一方、結合可能である場合(ステップS1111でYES)、ステップS1112へ進む。

【0284】

尚、オブジェクトが結合可能であるか否かの判定は、読み込んだコマンドの座標位置や、図形の種別等を基に判定する。また、文字列の場合は、フォントサイズやフォントの種類を基に判定する。例えば、図25の描画命令部分2505においては、コマンドC111~C114までの情報は、描画命令部分2504で規定されるタイルの情報にすでに含まれている。そこで、このような場合には、この重複したデータをなくすように動作する。

20

【0285】

(ステップS1112)

これは、図13のステップS912に対応するので、その説明は省略する。

【0286】

(ステップS1114)

これは、図13のステップS914に対応するので、その説明は省略する。

30

【0287】

(ステップS1115)

全てのコマンドの記述変換が終了したら、ページベクタとしてシステムメモリ5への書出を実行する。ページベクタデータとしては、ステップS1105及びステップS1106で記述されたコマンドに対し、タイルの開始と終了を示すコマンドを削除したようなフォーマットで記述される。

【0288】

まず、ページ内の一番最初のタイルに記述されたコマンドの書出時点では、システムメモリ5内にオブジェクトがない状態のページベクタデータを生成しておく。これは、図23の記述を用いて説明すると、コマンドC1~C6、C22だけで記述されるページである。次に、ステップS1110~ステップS1112で処理されたオブジェクトの記述を追加する。図23の記述を用いて説明すると、描画命令部分2302のコマンドC7~C11がこれにあたる。

40

【0289】

(ステップS1116)

これは、図13のステップS916に対応するので、その説明は省略する。

【0290】

(ステップS1117)

これは、図13のステップS917に対応するので、その説明は省略する。

50

【0291】

尚、コントローラ 1 中の画像データ展開部 18 の詳細については、実施形態 1 と同様な
ので、その説明は省略する。

【0292】

以上説明したように、実施形態 3 によれば、実施形態 1 で説明した効果に加えて、オブ
ジェクトを内包する矩形領域同士の重複部の割合に基づいて、一旦設定した矩形領域（タ
イル）を再設定（結合 / 分割）する。

【0293】

これにより、より適切なサイズの矩形領域からなるタイルベクタデータを生成すること
ができ、タイルベクタデータの並列処理の効率性を向上させることができる。

10

【0294】

[実施形態 4]

実施形態 4 は、実施形態 3 の応用例である。実施形態 4 が実施形態 3 と異なる点は、タ
イルベクタデータを生成する際に、オブジェクトを内包する矩形領域において、オブジェ
クト毎の矩形領域の重複部が所定値未満である場合、タイルの分割をより細かく行い、分
散処理による高速化をより行い易くした点である。

【0295】

例えば、図 16 のフローチャートにおいて、図 17 に示すラスタデータをタイルベクタ
データに変換する際、ステップ S52c の処理において、図 18 に示すように、それぞれの
オブジェクトを内包する矩形領域（タイル）1701 及び 1702 の重複部 1703 の
面積の割合が所定値未満である場合、タイル 1701 及び 1702 を分割し直すためのタ
イル分割処理を実行する。このタイル分割処理によって、矩形領域（タイル）1701 及
び 1702 を含むページは、例えば、図 27 の領域（タイル）1801 から 1808 に分
割する。

20

【0296】

ここで、図 19 のタイル分割処理による分割方法との差異は、タイル 1805a であり
、複数のオブジェクトを内包する矩形領域の重複部を独立したタイルとして生成している
。また、それ以外のタイルは、タイル 1804、またはタイル 1805b のようタイル内
に、単一オブジェクトが内包するようにタイルを分割している。

【0297】

このように、オブジェクトを内包する矩形領域の重複部の割合が所定値未満である場合
は、並列処理による高速化が見込める。そのため、例えば、2 つのオブジェクトを内包す
る矩形領域（タイル）が重複している割合が小さい場合は、2 つの単一オブジェクトを内
包するタイルと、両者の部分オブジェクトを複数の内包するタイルの計 3 つに分割するこ
とにより、タイルを細分化し、並列処理しやすくする。

30

【0298】

このように、タイルをより細分化すること以外は、実施形態 3 と同様の処理フローでデ
ータを処理することが可能である。

【0299】

以上説明したように、実施形態 4 によれば、実施形態 3 で説明した効果に加えて、より
タイルベクタデータの並列処理の効率性を向上させることができる。

40

【0300】

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。

【0301】

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

50

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。

【0302】

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。

【0303】

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。

【0304】

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리카ード、ROM、DVD（DVD-ROM、DVD-R）などがある。

【0305】

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。

【0306】

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。

【0307】

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。

【0308】

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。

【図面の簡単な説明】

【0309】

【図1】本発明の実施形態1の画像処理システムを構成するMFPのコントローラの詳細を示すブロック図である。

【図2】本発明の実施形態1の画像処理システムでのコピー動作に係るデータフローを示す図である。

【図3】本発明の実施形態1の画像処理システムでのプリント動作に係るデータフローを示す図である。

【図4】本発明の実施形態1の画像処理システムでの送信動作に係るデータフローを示す図である。

【図5】本発明の実施形態1のラスタ/タイルベクタ変換部が実行する処理を示すフローチャートである。

10

20

30

40

50

【図 6】本発明の実施形態 1 のタイルサイズ決定処理の具体例を説明するための図である。

【図 7】本発明の実施形態 1 のタイルサイズ決定処理の具体例を説明するための図である。

【図 8】本発明の実施形態 1 のタイルサイズ決定処理の具体例を説明するための図である。

【図 9】本発明の実施形態 1 のネットワークから転送されてくるドキュメントデータの一例を示している。

【図 10】本発明の実施形態 1 のページベクタデータの記述例を示す図である。

【図 11】本発明の実施形態 1 のタイルベクタデータの例を示す図である。

【図 12】本発明の実施形態 1 のタイルベクタデータの記述例を示す図である。

【図 13】本発明の実施形態 1 のタイル / ページベクタ変換部が実行する処理を示すフローチャートである。

【図 14】本発明の実施形態 1 の画像データ展開部が実行する処理を示すフローチャートである。

【図 15】本発明の実施形態 2 のタイルベクタデータの記述例を示す図である。

【図 16】本発明の実施形態 3 のラスタ / タイルベクタ変換部が実行する処理を示すフローチャートである。

【図 17】本発明の実施形態 3 のタイル重複部算出処理の具体例を説明するための図である。

【図 18】本発明の実施形態 3 のタイル重複部算出処理の具体例を説明するための図である。

【図 19】本発明の実施形態 3 のタイル分割処理の具体例を説明するための図である。

【図 20】本発明の実施形態 3 のタイル結合処理の具体例を説明するための図である。

【図 21】本発明の実施形態 3 のタイル結合処理の具体例を説明するための図である。

【図 22】本発明の実施形態 3 のネットワークから転送されてくるドキュメントデータの一例を示している。

【図 23】本発明の実施形態 3 のページベクタデータの記述例を示す図である。

【図 24】本発明の実施形態 3 のタイルベクタデータの例を示す図である。

【図 25】本発明の実施形態 3 のタイルベクタデータの記述例を示す図である。

【図 26】本発明の実施形態 3 のタイル / ページベクタ変換部が実行する処理を示すフローチャートである。

【図 27】本発明の実施形態 4 のタイル分割処理の具体例を説明するための図である。

【図 28】従来の画像処理システムの構成を示す図である。

【符号の説明】

【 0 3 1 0 】

- 1 コントローラ
- 2 システムバスブリッジ
- 3 CPU
- 4 メモリコントローラ
- 5 システムメモリ
- 6 汎用バス
- 7 ハードディスクコントローラ
- 8 ハードディスクドライブ
- 9 操作部コントローラ
- 10 操作部
- 11 ネットワーク I / F
- 12 ネットワーク
- 13 タイル / ページベクタ変換部
- 14 ラスタ / タイルベクタ変換部

10

20

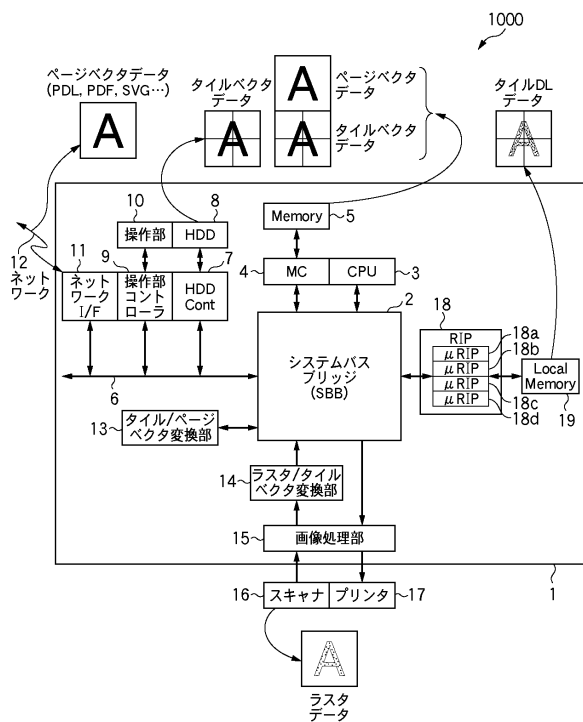
30

40

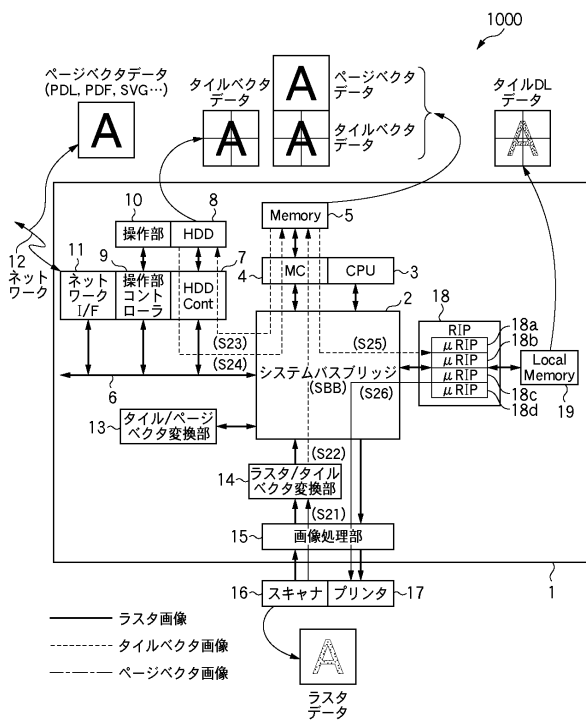
50

- 15 画像処理部
- 16 スキャナ
- 17 プリンタ
- 18 画像データ展開部
- 18 a ~ 18 d 小画像データ展開部
- 19 ローカルメモリ

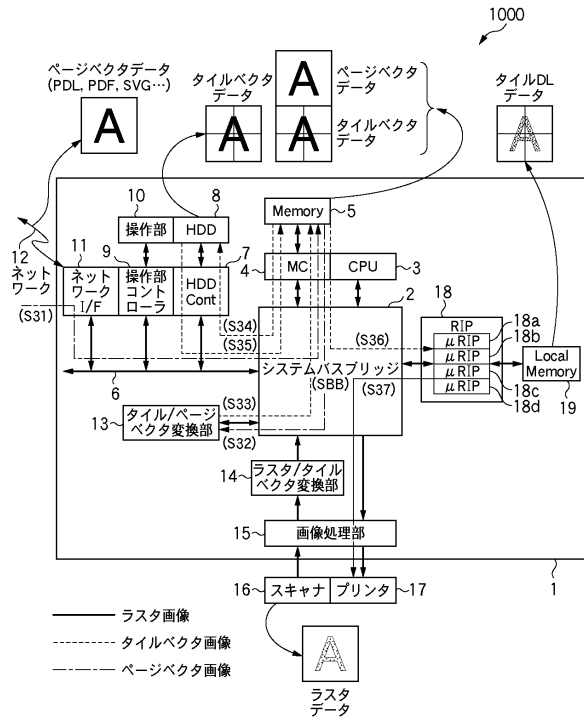
【図 1】



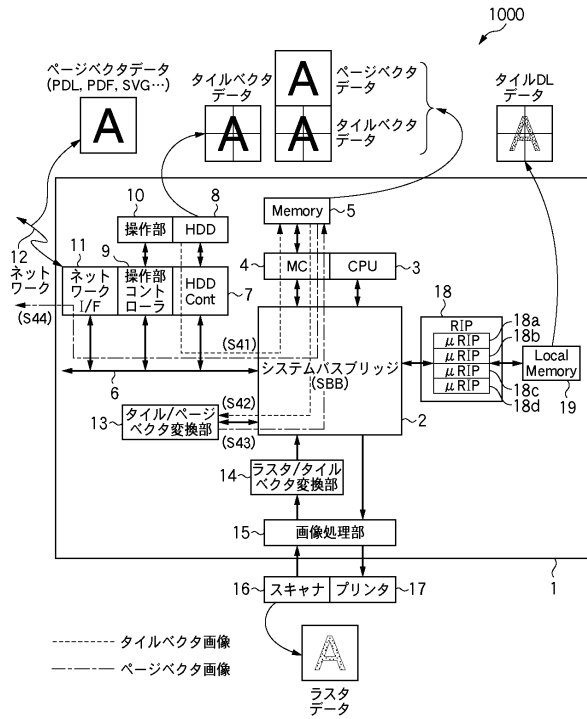
【図 2】



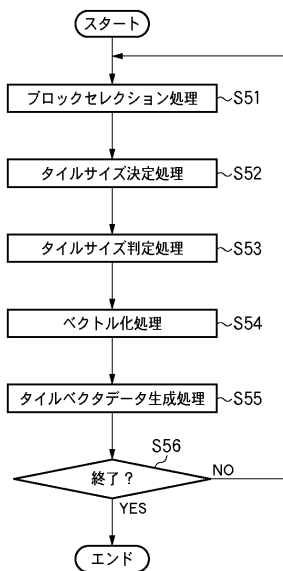
【図 3】



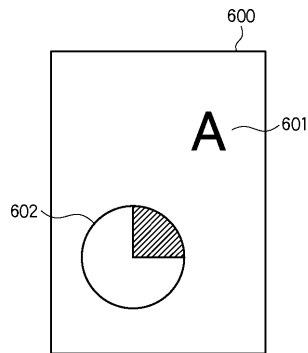
【図 4】



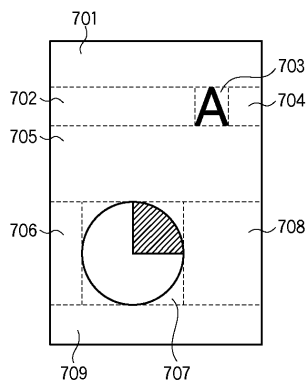
【図 5】



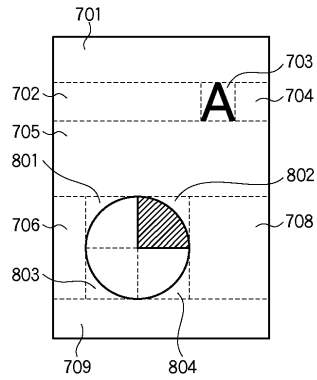
【図 6】



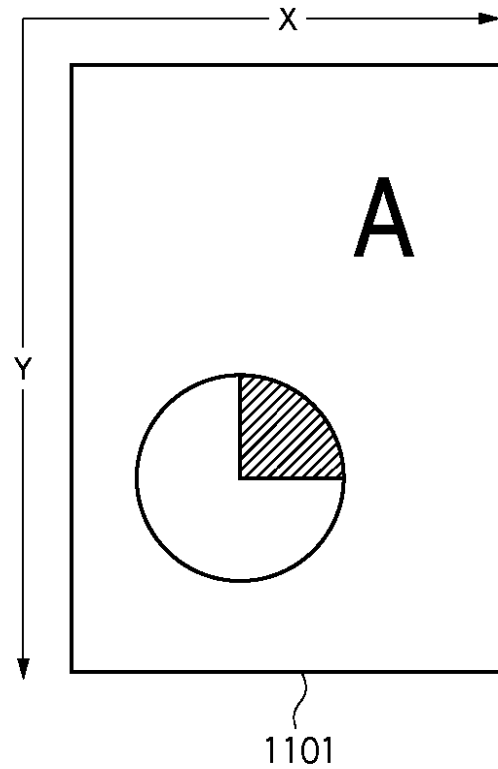
【図 7】



【図 8】



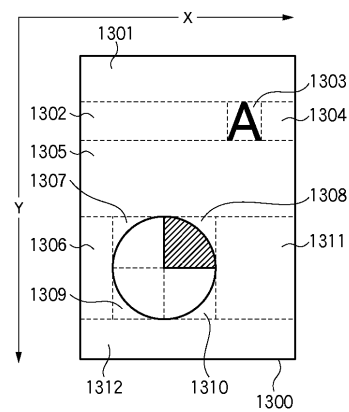
【図 9】



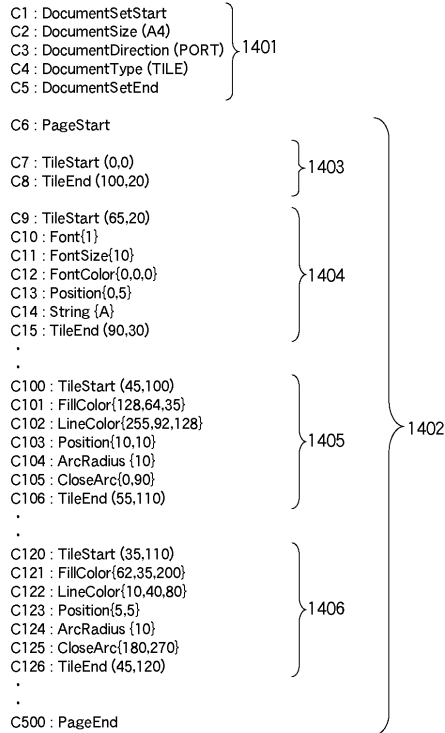
【図 10】

C1 : DocumentSetStart	}	1201
C2 : DocumentSize (A4)		
C3 : DocumentDirection (PORT)		
C4 : DocumentType (Page)		
C5 : DocumentSetEnd		
C6 : PageStart	}	1202
C7 : Font{1}		
C8 : FontSize{10}		
C9 : FontColor{0,0,0}		
C10 : Position{10,5}		
C11 : String {A}		
C12 : FillColor{128,64,35}	}	1203
C13 : LineColor{255,92,128}		
C14 : Position{40,300}		
C15 : ArcRadius {10}		
C16 : CloseArc{0,90}		
C17 : FillColor{62,35,200}		
C18 : LineColor{10,40,80}		
C19 : Position{40,300}		
C20 : ArcRadius {10}		
C21 : CloseArc{90,360}		
C22 : PageEnd		

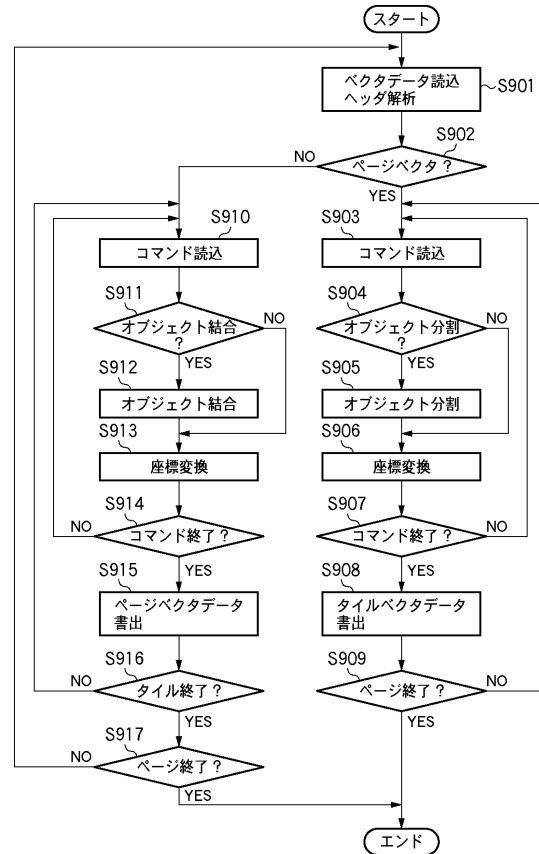
【図 11】



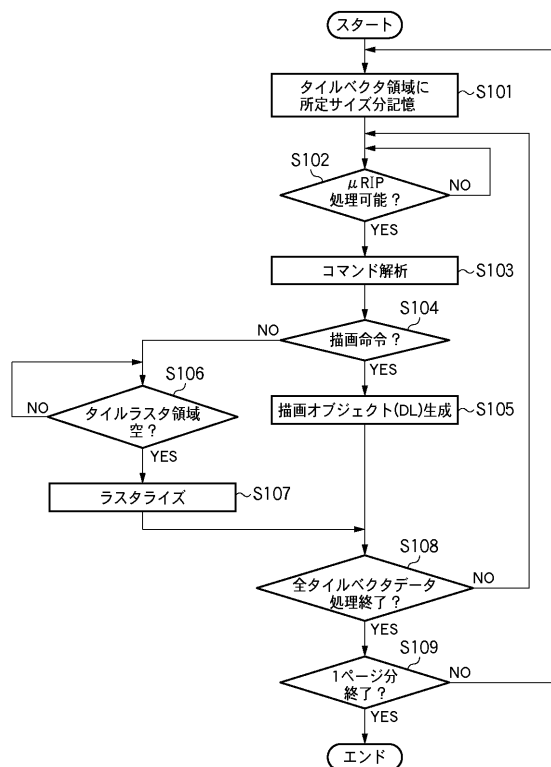
【図 12】



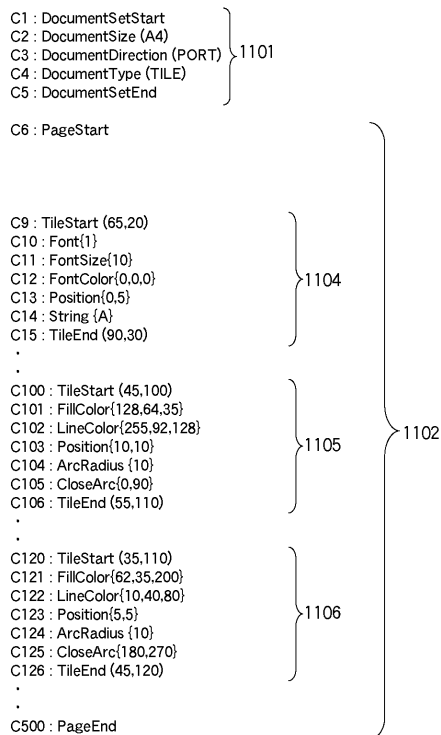
【図 13】



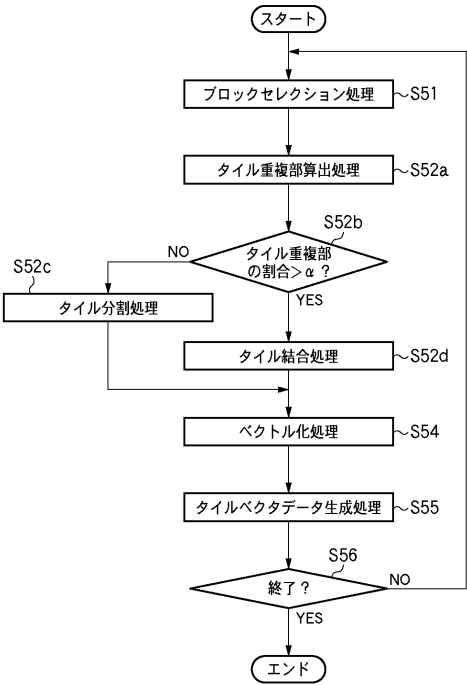
【図 14】



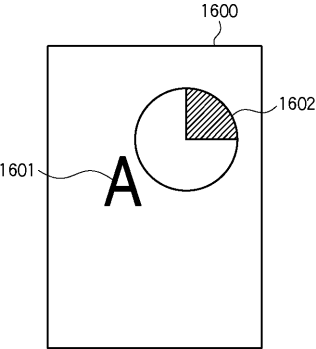
【図 15】



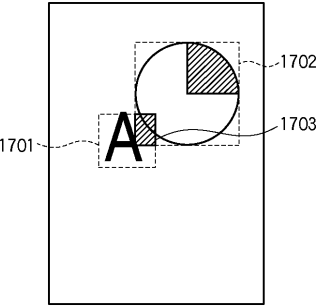
【図 16】



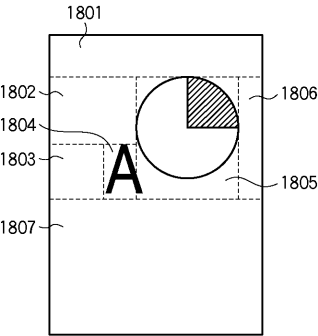
【図 17】



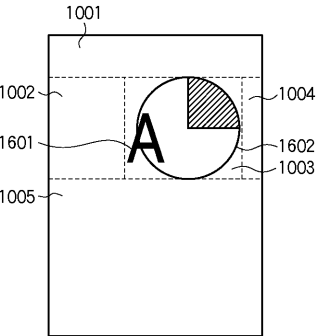
【図 18】



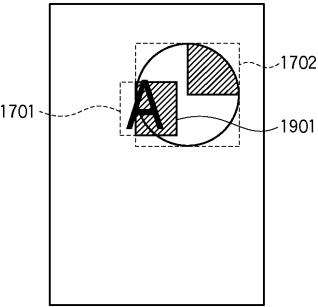
【図 19】



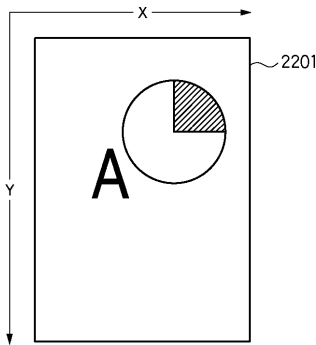
【図 21】



【図 20】



【図 22】



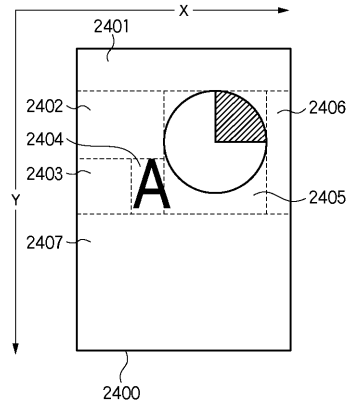
【図 23】

C1 : DocumentSetStart
 C2 : DocumentSize (A4)
 C3 : DocumentDirection (PORT)
 C4 : DocumentType (Page)
 C5 : DocumentSetEnd
 } 2301

 C6 : PageStart
 C7 : Font{1}
 C8 : FontSize{10}
 C9 : FontColor{0,0,0}
 C10 : Position{40,150}
 C11 : String {A}
 } 2302

 C12 : FillColor{128,64,35}
 C13 : LineColor{255,92,128}
 C14 : Position{80,100}
 C15 : ArcRadius {10}
 C16 : CloseArc{0,90}
 C17 : FillColor{62,35,200}
 C18 : LineColor{10,40,80}
 C19 : Position{80,100}
 C20 : ArcRadius {10}
 C21 : CloseArc{90,360}
 C22 : PageEnd
 } 2303

【図 24】



【図 25】

C1 : DocumentSetStart
 C2 : DocumentSize (A4)
 C3 : DocumentDirection (PORT)
 C4 : DocumentType (TILE)
 C5 : DocumentSetEnd
 } 2501

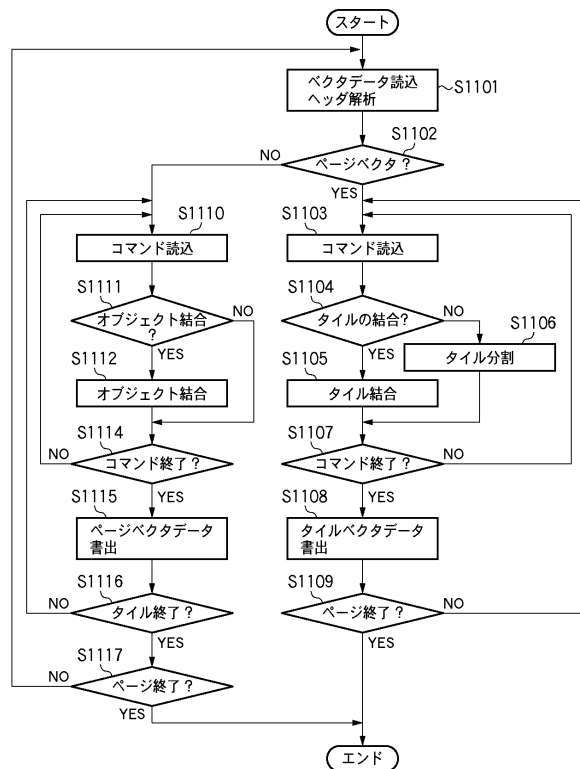
 C6 : PageStart
 } 2503

 C7 : TileStart (0,0)
 C8 : TileEnd (100,20)
 } 2504

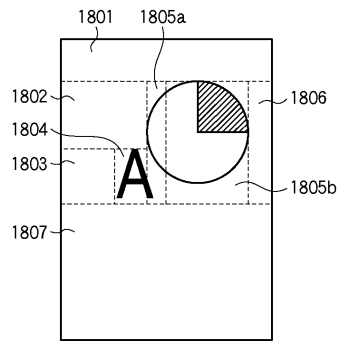
 C9 : TileStart (35,150)
 C10 : Font{1}
 C11 : FontSize{10}
 C12 : FontColor{0,0,0}
 C13 : Position{40,150}
 C14 : String {A}
 C15 : TileEnd (45,160)
 .
 C100 : TileStart (45,100)
 C101 : FillColor{128,64,35}
 C102 : LineColor{255,92,128}
 C103 : Position{55,120}
 C104 : ArcRadius {10}
 C105 : CloseArc{0,90}
 C106 : FillColor{62,35,200}
 C107 : LineColor{10,40,80}
 C108 : Position{50,105}
 C109 : ArcRadius {10}
 C110 : CloseArc{180,270}
 C111 : Font{1}
 C112 : FontSize{10}
 C112 : FontColor{0,0,0}
 C113 : Position{40,150}
 C114 : String {A}
 C115 : TileEnd (60,160)
 .
 C500 : PageEnd
 } 2502

 } 2505

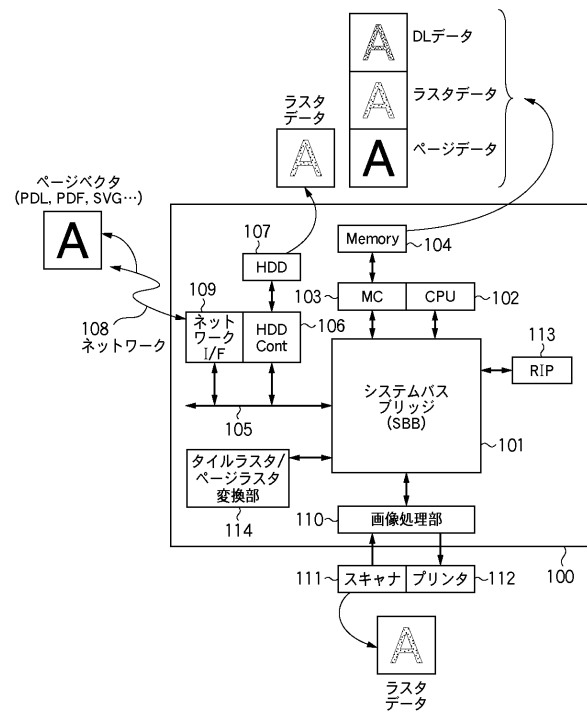
【図 26】



【図 27】



【図 28】



フロントページの続き

審査官 名取 乾治

(56)参考文献 特開平 1 1 - 1 9 8 4 6 5 (J P , A)
特開平 1 1 - 2 0 3 0 6 1 (J P , A)
特開平 1 0 - 2 0 7 6 5 0 (J P , A)
特開平 1 0 - 3 0 7 9 2 4 (J P , A)
特開 2 0 0 4 - 1 2 0 6 3 9 (J P , A)
特開 2 0 0 3 - 0 4 6 7 8 9 (J P , A)
特開 2 0 0 2 - 0 1 4 7 8 0 (J P , A)

(58)調査した分野(Int.Cl. , D B 名)

B 4 1 J	5 / 3 0
B 4 1 J	2 1 / 0 0
H 0 4 N	1 / 2 1
G 0 6 F	3 / 1 2
H 0 4 N	1 / 3 8 - 1 / 3 9 3