

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2011-165010

(P2011-165010A)

(43) 公開日 平成23年8月25日(2011.8.25)

(51) Int.Cl.

G 06 F 3/12 (2006.01)
B 41 J 29/38 (2006.01)

F 1

G 06 F 3/12
B 41 J 29/38

テーマコード(参考)

2 C O 6 1
Z

審査請求 未請求 請求項の数 9 O L (全 16 頁)

(21) 出願番号

特願2010-28212(P2010-28212)

(22) 出願日

平成22年2月10日(2010.2.10)

(71) 出願人 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

(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

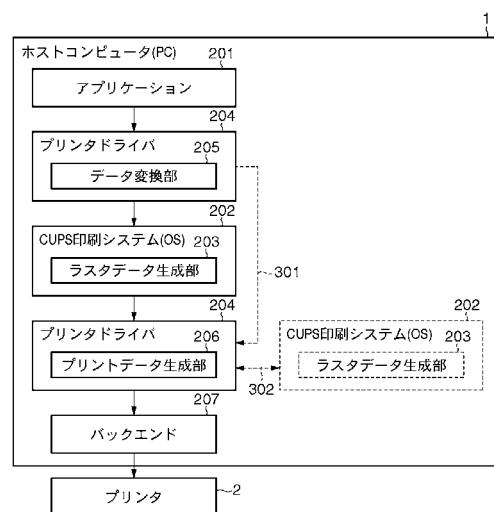
(54) 【発明の名称】情報処理方法および情報処理装置、プログラム

(57) 【要約】

【課題】プリンタドライバが、ラスタデータに含まれるオブジェクトの領域を判定するには、複雑な座標計算を要したり、OSの印刷処理を詳細に把握したりする必要があった。

【解決手段】プリンタドライバのデータ変換部は、PDF形式のスプールファイルを複製して外部記憶装置にPDFデータ1として保存する。プリンタドライバのラスタデータ生成部は、データ変換部が保存したPDFデータ1を取得し、PDFデータ1に含まれるオブジェクトに従って、対象のオブジェクトを描画した新たなPDFデータ2を作成し、PDFデータ2をOSのラスタデータ生成部にてラスタデータ2に変換させる。ラスタデータ生成部は、生成したラスタデータ2を参照して、スプールファイルから生成されたラスタデータのうち対象のオブジェクトを特定し、必要な処理を施してプリントデータを生成する。

【選択図】図3



【特許請求の範囲】**【請求項 1】**

ページ記述言語で記述された第1の形式のデータを印刷設定に従ってラスタデータに変換するラスタデータ生成手段を備えた情報処理装置であって、

前記第1の形式の処理対象データに基づいて、該処理対象データに含まれた特定のオブジェクトの領域を示す領域データを含む前記第1の形式のオブジェクト領域データを生成するオブジェクト領域データ作成手段と、

前記ラスタデータ生成手段によりラスタデータに変換された前記処理対象データと、前記ラスタデータ生成手段により前記処理対象データと同じ印刷設定に従ってラスタデータに変換された前記オブジェクト領域データとを対比し、ラスタデータに変換された前記処理対象データにおける前記特定のオブジェクトを特定する特定手段と
を備えることを特徴とする情報処理装置。
10

【請求項 2】

前記特定のオブジェクトは、特定の種類のオブジェクトであることを特徴とする請求項1に記載の情報処理装置。

【請求項 3】

前記特定の種類のオブジェクトは、ビットマップオブジェクトであることを特徴とする請求項2に記載の情報処理装置。

【請求項 4】

色インクで画像を形成する機能に加え、光沢インクで画像形成する機能を有するラスタプリンタに接続するためのインターフェース手段を更に備え、
20

前記特定手段により特定された、ラスタデータに変換された前記特定のオブジェクトを、光沢インクに対応する成分として、ラスタデータに変換された前記処理対象データに附加する処理手段を更に備えることを特徴とする請求項1乃至3のいずれか一項に記載の情報処理装置。

【請求項 5】

ラスタデータに変換された前記処理対象データに含まれた、前記特定手段により特定された前記特定のオブジェクトを、他のオブジェクトで置換する手段を更に備えることを特徴とする請求項1乃至3のいずれか一項に記載の情報処理装置。

【請求項 6】

前記第1の形式は、PDF (Portable Document Format) 形式またはPostScript形式であることを特徴とする請求項1乃至5のいずれか一項に記載の情報処理方法。
30

【請求項 7】

ページ記述言語で記述された第1の形式のデータを印刷設定に従ってラスタデータに変換するラスタデータ生成手段をオペレーティングシステムの一部として提供するコンピュータを、

前記第1の形式の処理対象データに基づいて、該処理対象データに含まれた特定のオブジェクトの領域を示す領域データを含む前記第1の形式のオブジェクト領域データを生成するオブジェクト領域データ作成手段と、
40

前記ラスタデータ生成手段によりラスタデータに変換された前記処理対象データと、前記ラスタデータ生成手段により前記処理対象データと同じ印刷設定に従ってラスタデータに変換された前記オブジェクト領域データとを対比し、ラスタデータに変換された前記処理対象データにおける前記特定のオブジェクトを特定する特定手段として機能させるためのプログラム。

【請求項 8】

ページ記述言語で記述された第1の形式のデータを印刷設定に従ってラスタデータに変換するラスタデータ生成手段を備えた情報処理装置における情報処理方法であって、

オブジェクト領域データ作成手段が、前記第1の形式の処理対象データに基づいて、該処理対象データに含まれた特定のオブジェクトの領域を示す領域データを含む前記第1の
50

形式のオブジェクト領域データを生成するオブジェクト領域データ作成工程と、

前記ラスタデータ生成手段が、前記処理対象データと前記オブジェクト領域データと同じ印刷設定にしたがってラスタデータに変換する変換工程と、

特定手段が、前記ラスタデータ生成手段によりラスタデータに変換された前記処理対象データと、前記ラスタデータ生成手段により前記処理対象データと同じ印刷設定に従ってラスタデータに変換された前記オブジェクト領域データとを対比し、ラスタデータに変換された前記処理対象データにおける前記特定のオブジェクトを特定する特定工程とを有することを特徴とする情報処理方法。

【請求項 9】

ページ記述言語で記述された第1の形式のデータを印刷設定に従ってラスタデータに変換するラスタデータ生成手段と、

前記第1の形式の処理対象データに基づいて、該処理対象データに含まれた特定のオブジェクトの領域を示す領域データを含む前記第1の形式のオブジェクト領域データを生成するオブジェクト領域データ作成手段と、

前記オブジェクト領域データ作成手段により生成されたオブジェクト領域データを前記処理対象データと同じ印刷設定に従ってラスタデータに変換する変換手段と、

前記変換手段により変換されたラスタデータは、光沢インクにより印刷され、前記ラスタデータ生成手段により変換されたラスタデータは通常インクで印刷される印刷データを生成する生成手段と

してコンピュータを機能させることを特徴とするプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、情報処理方法および情報処理装置およびプログラムに関し、例えば印刷データに含まれるオブジェクトの領域を判定するための処理方法に関する。

【背景技術】

【0002】

一般に、ホストコンピュータ（以下、PCと称す）に搭載されるオペレーティングシステム（以下、OSと称す）には、印刷要求を行なうPCにプリンタが接続された環境での印刷をサポートするために印刷システムが備えられている。このような印刷システムとしては、米国アップル社のMac OS X（登録商標）で動作するCUPS（Common UNIX（登録商標）Printing System）印刷システムが知られている。また、Mac OS Xでは、アプリケーションからの印刷要求に応じてOSが生成するスプールファイルの標準データ形式としてPDF（Portable Document Format（登録商標））を採用している。

【0003】

プリンタは一般にスプールファイルを直接解釈する事ができないため、印刷システムおよびプリンタドライバにより、プリンタが解釈可能なデータ形式に変換する。例えば、インクジェットプリンタのようなラスタプリンタで印刷する際には、以下のようないくつかの処理部が連携して順に処理を行い、データ形式を変換していくことで印刷を実現している。また、これら処理部の順序はシステムによって決められるようになっている。

【0004】

ここで、プリンタドライバによっては、プリンタでより高画質な印刷を行なうために、印刷データに含まれるビットマップやテキストなどのオブジェクトに応じて画像処理を行なうものがある。このようなプリンタドライバは、OSからページ記述言語（以下PDL

10

20

30

40

50

と称す)形式の印刷データを受け取り、その描画コマンドを解析することで、印刷データに含まれるビットマップやテキストなどのオブジェクトを認識している。そして、その認識結果に応じて、各々のオブジェクトに応じた画像処理を行い印刷を行なっている(例えば特許文献1等参照)。

【先行技術文献】

【特許文献】

【0005】

【特許文献1】特開2002-247367号公報

【発明の概要】

【発明が解決しようとする課題】

【0006】

ここで、上述のようなラスタプリンタに対するCUPS印刷システムにおいて、印刷データに含まれるオブジェクトに応じて、プリンタで印刷可能なプリントデータに画像処理を施すことを考える。このような場合、プリンタドライバのプリントデータ生成部は、印刷データに含まれるオブジェクトを識別し、その領域を判定する必要がある。しかしながら、プリントデータ生成部に渡されるデータは、OSのラスタデータ生成部により生成されたラスタデータであるため、印刷データに含まれるオブジェクトの領域を判定するのは一般に困難である。ラスタデータを解析し、オブジェクトの領域を判定するには、複雑な計算を要するが、正しく判定できるとは限らない。また、印刷データに含まれるビットマップオブジェクトの背景色とページの色が同色の場合、領域を正しく判定できない可能性もある。

【0007】

一方、上述したプリンタドライバのデータ変換部によるオブジェクトの領域判定の可能性について考えてみる。このデータ変換部は、Mac OS X Leopard 10.5におけるCUPS 1.3より追加されたcupsPrefilterに対応するものである。データ変換部は、OSのラスタデータ生成部によってラスタデータに変換される前の印刷データである、PDF形式のスプールファイルを入力として受け取り、PDF形式のデータを出力するものである。この出力データがプリンタドライバのプリントデータ生成部に入力される。したがって、PDFデータの内部コマンドを解析することにより、印刷データに含まれるオブジェクトの領域を判定する事は一見可能なように思われる。

【0008】

しかしながら、ここで一つ問題がある。Mac OSの仕様では、印刷を行なう際に設定可能なページの割付、拡縮、送り順等のページに関わる操作は、OSが提供するラスタデータ生成部により行われる。このため、プリンタドライバのデータ変換部でPDF形式のスプールファイルを解析して得られるオブジェクトの領域はそのまま利用する事はできない。ラスタデータ生成部で実行されるページの割付、拡縮、送り順等の操作がスプールファイルには反映されておらず、プリンタドライバのプリントデータ生成部に渡されるラスタデータとは一般に一致しないためである。両者を一致させるにはOSのラスタデータ生成部の詳細な仕様を把握し、ラスタデータ生成部と同様な処理を行ってから座標計算を行なう必要がある。しかしながら、ラスタデータ生成部の仕様は公開されていないため、これも一般に困難である。

【0009】

本発明は上記の問題点を鑑みて成されたもので、例えばラスタプリンタによる印刷等のために印刷データをラスタデータへ変換する機能をOSが担う印刷システムにおいて、プリンタドライバが複雑な計算をすることなく、当該印刷システムから出力されるラスタデータでのオブジェクトの領域を判定可能な情報処理装置および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】

【0010】

上記目的を達成するために、本発明は以下のような構成を備える。すなわち、

10

20

30

40

50

ページ記述言語で記述された第1の形式のデータを印刷設定に従ってラスタデータに変換するラスタデータ生成手段を備えた情報処理装置であって、

前記第1の形式の処理対象データに基づいて、該処理対象データに含まれた特定のオブジェクトの領域を示す領域データを含む前記第1の形式のオブジェクト領域データを生成するオブジェクト領域データ作成手段と、

前記ラスタデータ生成手段によりラスタデータに変換された前記処理対象データと、前記ラスタデータ生成手段により前記処理対象データと同じ印刷設定に従ってラスタデータに変換された前記オブジェクト領域データとを対比し、ラスタデータに変換された前記処理対象データにおける前記特定のオブジェクトを特定する特定手段とを備える。

【発明の効果】

10

【0011】

本発明によれば、ラスタ化処理前後のデータに含まれるオブジェクトを高精度に対応付けることができる。これにより、ラスタデータ中のオブジェクトの領域を高精度に決定することができる。例えばオペレーティングシステム等により提供されるラスタ化処理において画像データ中のオブジェクトの変倍や移動等が行われる場合、ラスタ化処理の過程でオブジェクトの配置や大きさを変更するものであっても、ラスタ処理前の例えばページ記述言語形式のデータ中のオブジェクトと、ラスタデータに含まれるオブジェクトとを高精度に対応付け、ラスタデータ中のオブジェクトの領域を判定することができる。

【図面の簡単な説明】

20

【0012】

【図1】本発明の実施形態におけるハードウェア構成を示すブロック図

【図2】情報処理装置のソフトウェア構成を示すブロック図

【図3】本発明の第1の実施形態を示し、印刷データの処理順序を説明するブロック図

【図4】OSおよびプリンタドライバが提供するユーザインターフェースを表す図

【図5】プリンタドライバの処理の流れを説明するフローチャート

【図6】実際の印刷データとビットマップ領域判定のために生成されるデータの一例を示す図

【図7】本発明の第2の実施形態を示し、印刷データの処理順序を説明するブロック図

【図8】実際の印刷データとビットマップ領域およびテキスト領域判定のために生成されるデータの一例を示す図

30

【発明を実施するための形態】

【0013】

以下、添付図面を参照して本発明の好適な実施の形態を詳しく説明する。尚、以下の実施の形態は特許請求の範囲に係る本発明を限定するものでなく、また本実施の形態で説明されている特徴の組み合わせの全てが本発明の解決手段に必須のものとは限らない。

【0014】

[実施形態1]

<情報処理システムのハードウェア構成>

図1は本発明の実施形態における情報処理装置及び周辺装置からなる情報処理システムのハードウェア構成を示すブロック図である。同図は、PC1とEthernet(登録商標)やUniversal Serial Bus(以下、USBと称す)などの通信バス111を介してプリンタA2、プリンタB3とが通信可能な印刷システムの例である。PC1は情報処理装置の一例である。PC1には、Mac OS X(登録商標)と同等のOSがインストールされているものとし、印刷システムとしてはMac OS X(登録商標)に導入されているCUPS印刷システムを想定している。図1では、プリンタは2つのみ図示したが、通信バス111を介して任意の数のプリンタが接続可能である。

【0015】

PC1において、CPU101はRAM103及び外部記憶装置104に格納されたプログラムに従って各部を制御する。CPU101は、プリンタA2及びプリンタB3に対

40

50

応したプリンタドライバを実行し、アプリケーションプログラムに基づいて作成された出力情報をプリンタに対応する出力コマンドに変換し、プリンタA2またはプリンタB3に所定のプロトコルで出力する。図5のフローチャートの手順は、CPU101により実行される各処理部（データ変換部やプリントデータ生成部）により実現される。ROM102は、CPU101が実行するBIOSプログラムやフォントデータを格納している。RAM103は、必要に応じてCPU101がデータやプログラムの書き込みや読み出しを行なうために使用され、プリンタドライバ起動時には、プリンタドライバのプログラムコードがこのRAM103にロードされる。外部記憶装置104はファイルストレージであり、CPU101が読み出すプログラムコードやスプールファイル、生成されたラスタデータ等が格納される。外部記憶装置104は本実施形態の機能の一部を実現するデバイスで、例えばFD、CD-RW、HDDなどである。入力デバイス制御部105は、キーボードやマウス等の入力デバイス107を制御する。表示装置制御部106は、CRT等の表示装置108を制御する。内部バス109は、各構成要素間のデータ通信を行なうデータバス、システムバスで構成される。通信I/F110は、通信バス111を介してプリンタA2及びプリンタB3との通信を司り、双方向通信可能である。なお、シリアルインターフェースとしてはUSBやIEEE1394など、パラレルインターフェースとしてはセントロニクスやSCSIなど複数の種類が利用可能であるが、双方向の通信を実現するものであればどのようなインターフェースでも構わない。

10

【0016】

プリンタA2はラスタデータを受信して印刷を行うラスタプリンタである。また後述するように、色インクで画像を形成する機能に加え、光沢インクで画像形成する機能を有している。プリンタA2において、プリントコントローラ113は、通信I/F112より得られたプリンタコマンドをもとにプリンタエンジン114を制御し印刷を行なう。プリンタエンジン114は、プリンタヘッドの操作や記録媒体の搬送を行なう機構である。

20

【0017】

<情報処理システムのソフトウェア構成>

図2は、PC1のソフトウェア構成を示すブロック図である。PC1は、CUPS印刷システム202、プリンタドライバ204を備え、アプリケーション201の印刷データをCUPS印刷システム202及びプリンタドライバ204を通じてプリンタA2が処理可能（印刷可能）なプリントデータに変換して出力する。アプリケーション201は、CUPS印刷システム202に対して印刷要求を行い、印刷される描画データと、用紙サイズや用紙種類、またページの割付や拡縮などが設定された印刷設定情報とをCUPS印刷システム202に受け渡す。CUPS印刷システム(OS)202は、アプリケーション201から送られた描画データのスプール処理や、プリンタドライバ204のロード／実行などの様々な印刷制御を行なう。また、OS202はラスタデータ生成部203を持ち、印刷データをラスタデータへ変換する機能を持つ。プリンタドライバ204は、データ変換部205とプリントデータ生成部206とを備えており、CUPS印刷システム202から印刷データと印刷設定情報を受け取り印刷動作に必要な処理を行なう。データ変換部205は、ラスタデータ生成部203に先だって、受け取った印刷データに対して印刷処理に必要な前処理を行なう。データ変換部205で処理された印刷データは前述のラスタデータ生成部203を経由してプリントデータ生成部206に渡される。プリントデータ生成部206は、ラスタデータ生成部203で生成されたラスタデータを受け取り、印刷設定情報に基づいてプリントデータを生成する。バックエンド207は、プリントデータ生成部206によって生成されたプリントデータをプリンタA2へ出力する。

30

【0018】

図3は、本発明の第1の実施形態について印刷データの処理順序を説明するブロック図である。本ブロック図は、その特徴的な構成として、本実施形態に特有のオブジェクト領域判定用の新たなデータフロー（破線で示す）を有する。すなわち、データ変換部205からプリントデータ生成部206へのデータフロー301、およびプリントデータ生成部206とラスタデータ生成部203間のデータフロー302である。以下、OS202が

40

50

生成するスプールファイルの形式は、OS標準のPDF（Portable Document Format）形式とし、アプリケーション201によって印刷が行なわれた際の処理を順を追って説明する。なおOS202が生成するスプールファイルは、PDF等、オブジェクトを識別可能に記述するページ記述言語の形式を有し、本実施形態ではこの形式を第1の形式と呼ぶ。

【0019】

ここで、本実施形態では、発明の効果をわかりやすく説明するために、OS202は印刷設定として1枚あたりのページ数である割付設定を選択できる機能を持つものとする。また、この印刷設定は、OS202が提供するユーザインターフェース（以下、UIと称す）を通して、アプリケーション201からの印刷時に設定できるものとする。図4(a)がそのUIの一例を表すものであり、UI401はページのレイアウトを設定するため表示される。UI401は、プリンタ選択メニュー部402と印刷設定パネル選択メニュー部403と印刷設定パネル表示部404から構成される。プリンタ選択メニュー部402は、OS202にプリンタが複数登録されている場合にその一覧をメニュー項目として表示し、プリンタを選択可能とするための選択欄である。印刷設定パネル選択メニュー部403は、印刷処理に必要な各種設定項目を関連する項目群で分類し、その項目群の一覧をメニュー項目として表示し、設定項目を選択可能とするための選択欄である。印刷設定パネル選択メニュー部403で選択可能な項目としては、OS202が提供するものとプリンタドライバ204が提供するものが存在し、同図で選択されている「レイアウト」はOS202が提供する機能の一例である。印刷設定パネル表示部404は、印刷設定パネル選択メニュー部403で選択された設定項目に基づいて、更にその詳細が表示される。割付設定部405は、1枚あたりのページ数を設定するための設定欄であり、ポップアップメニューによって割付の設定を行なう事ができる。なお、本実施形態では、割付設定部405で「2ページ/枚」と設定されたものとして話を進める。なお図4のUIで選択された印刷設定は、印刷設定情報の一部として保存される。

10

20

30

40

【0020】

さらに、本実施形態で印刷に用いるプリンタA2は、光沢をもつ特殊なインク（以下、光沢インクと称す）を用紙に塗布して光沢インクで画像形成する機能を備える。プリンタA2へは通常のプリントデータとともに光沢インクを塗布する領域データを送る事により光沢インクによる印刷が実行される。アプリケーション201で印刷を実行する際には、プリンタドライバ204が提供するUIを通して光沢インクに関するオプションを選択できるものとする。図4(b)がその一例を表すものであり、光沢インクを塗布する領域を選択させるために表示するものである。プリンタ選択メニュー部402、印刷設定パネル選択メニュー部403、印刷設定パネル表示部404は、図4(a)と同等の役割を果たすものである。ラジオボタン406がユーザに選択させるためのラジオボタンであり、「ページ全面に塗布する」か「ビットマップ領域のみに塗布する」のいずれかを選択できる。「ページ全面に塗布する」が選択されている場合は、光沢インクをページ全面に塗布し、「ビットマップ領域のみに塗布する」が選択されている場合は、ビットマップ領域以外の部分には光沢インクを塗布せず、ビットマップ領域のみに光沢インクを塗布する。OS202のラスタデータ生成部203およびプリンタドライバ204のデータ変換部205とプリントデータ生成部206は、これらOS202が提供する機能、プリンタドライバ204が提供する機能の設定値を、印刷設定情報として印刷データとともに受け取ることができ、それに応じて処理を行なう。

30

40

【0021】

<画像データ処理>

図3に戻って具体的な処理の説明を行なう。ユーザによる操作等に応じてアプリケーション201から印刷要求が行われると、OS202は印刷される描画データと印刷設定情報をアプリケーション201から受け取り、PDF形式のスプールファイル（以下、PDF印刷データと称す）を生成する。前述の通りこれが第1の形式のデータに相当する。図6(a)はPDF印刷データの一例を表す。PDF印刷データ601は印刷されるPD

50

F形式のデータ1ページを表し、オブジェクト602はビットマップオブジェクト、オブジェクト603はテキストオブジェクトを表す。本例ではビットマップオブジェクト602が領域判定の対象、すなわち特定すべきオブジェクト（特定のオブジェクト）となる。PDF印刷データでは、これらのオブジェクトが特定可能である。

【0022】

生成されたPDF印刷データはまず、プリンタドライバ204のデータ変換部205へ入力される。データ変換部205では、PDF印刷データと印刷設定情報を受け取り、PDF印刷データに対して処理を行なった後、PDF形式の印刷データを出力する。

【0023】

図5(a)は、データ変換部205の処理を説明するフローチャートである。データ変換部205は、まずOS202からPDF印刷データ（すなわち処理対象データ）を受け取る(S501)。そして次に、データ変換部205は、S502でビットマップ領域の判定を行なうかどうかを印刷設定情報から判断する。本実施形態では、図4(b)のUIにて、「ビットマップ領域のみに塗布する」が選択されている場合は、S503に進む。また、「ページ全面に塗布する」が選択されている場合には、S503をスキップし、S504に進む。S503では、データ変換部205が、PDF印刷データを外部記憶装置104にコピーする。この作成したコピーをPDFデータ1とする。この処理は、後ほど処理を行なうプリントデータ生成部206が、CUPS印刷システムにおける通常のフローでは利用できないPDF印刷データを利用できるようにするためのものである。

10

【0024】

最後に、データ変換部205は、S504にてPDF印刷データを出力して終了する。ここで、本実施例におけるデータ変換部205は、受け取ったPDF印刷データそのものに変更を加える処理は行なわないため、入力されるPDF印刷データ、保存するPDFデータ1、出力するPDF印刷データは全て等しいものである。例えば、前述した図6の印刷データの例であれば、PDFデータ1と出力するPDF印刷データは、共に図6(a)と同じものである。もちろん、必要に応じてPDF印刷データに変更を加えることも可能であり、その場合には保存するPDFデータ1と出力するPDF印刷データが等しいものとなるようすればよい。

20

【0025】

図3に戻って説明を続ける。プリンタドライバ204のデータ変換部205により出力されたPDF印刷データは、次にOS202のラスタデータ生成部203に入力される。ラスタデータ生成部203は、OS202が提供するものであり、入力されたPDF印刷データを印刷設定情報に従ってラスタデータへ変換する処理を行なう。以後、ここで生成されるラスタデータをラスタ印刷データと称する。さらに、ラスタデータ生成部203は、その重要な役割として、印刷設定情報に設定されたページの割付、拡縮といったページに関わる操作も行なう。例えば、ラスタデータ生成部203が、データ変換部205の出力である図6(a)のPDF印刷データ601を、先の図4(a)で示したUIのように「2ページ/枚」の設定でラスタライズした場合、図6(d)のようなラスタ印刷データ607が生成される。図6(d)のラスタ印刷データ607は、ラスタデータ形式の印刷データ1ページを表し、この状態ではページ内に含まれるビットマップをオブジェクトとしては認識できない。ラスタデータ生成部203は、最後に、生成したラスタ印刷データを出力し、終了する。

30

【0026】

さらに図3に戻って説明を続ける。ラスタデータ生成部203により出力されたラスタ印刷データは、次にプリンタドライバ204のプリントデータ生成部206に入力される。プリントデータ生成部206は、その基本的な役割として、受け取ったラスタ印刷データに対し、印刷設定情報に従ってプリンタA2で解釈可能なデータ形式へ変換する処理を担うものである。さらに、本実施形態では、ラスタ印刷データに含まれるビットマップ領域を判定し、その領域に応じた処理を行なう。

40

【0027】

50

図5(b)は、プリントデータ生成部206の処理を説明するフローチャートである。プリントデータ生成部206は、まず、S505にてラスタデータ生成部203が出力したラスタ印刷データを受け取る。このラスタ印刷データが、実際に印刷されるデータである。次に、プリントデータ生成部206は、S506にてビットマップ領域の判定を行なうかどうかを印刷設定情報から判断する。プリントデータ生成部206は、データ変換部205のS502と同様に、図4(b)のU1で「ビットマップ領域のみに塗布する」が選択されている場合は、S507に進む。また、「ページ全面に塗布する」が選択されている場合には、S513に進み、通常のプリントデータ生成処理を行なう。S507からS512が、実際にビットマップ領域の判定を行なう処理である。

【0028】

10

プリントデータ生成部206は、まずS507にて、データ変換部205が保存したPDFデータ1を取得する。次に、プリントデータ生成部206は、S508にて、このPDFデータ1の内部コマンドを解析し、ラスタ印刷データにおけるビットマップ領域を判定するための新たなPDFデータ2を作成する。すなわちオブジェクト領域データ作成の機能を遂行する。PDFデータ2は、PDFデータ1の内部コマンドを解析することにより、ビットマップオブジェクトが描画される領域を取得することで、対象となる領域に白色以外の色(例えば黒色)で内部を塗りつぶした矩形を描画したものである。このPDFデータ2は、対象となるオブジェクトの領域を特定するためのデータであり、オブジェクト領域データと呼ぶこともある。例えば、図6(a)のPDFデータ1から生成されるPDFデータ2は、図6(b)のような結果となる。図6(b)にて、オブジェクト領域データ604は図6(a)のPDF印刷データ601と同じページ構成をもつPDF形式のページであり、オブジェクト605はビットマップオブジェクトの領域を表す、描画した黒色の矩形である。黒色としているのは本例では黒色成分を用いてオブジェクト領域を描画しているためであり、色には特別な意味はなく、どの色成分を用いてもよい。しかし特定されるべきオブジェクト(ビットマップオブジェクト)に対応する画素を本実施形態では「黒」と呼ぶことにする。

20

【0029】

30

次に、プリントデータ生成部206は、ラスタデータ形式でのビットマップ領域を判定するためにS509の処理を行なう。すなわち、プリントデータ生成部206は、OS202のラスタデータ生成部203を内部的に起動して、生成したPDFデータ2をラスタライズし、ラスタデータ1に変換する処理を行なう。ここで注意として、ラスタデータ生成部203を動作させる際には、PDF形式の入力データと共に印刷設定情報を渡す必要がある。その印刷設定情報はプリントデータ生成部206に渡される印刷設定情報と少なくともページに関わる設定について同じ設定値を設定したものを渡すようにする。こうすることにより、プリントデータ生成部206が内部的に起動したラスタデータ生成部203より得るラスタデータ1は、ラスタ印刷データと同じページ設定を適用されたラスタデータとなる。そのため、ラスタデータ1とラスタ印刷データの各ピクセルは一対一に対応する。そして、ラスタデータ1のうち、矩形を描画された部分、すなわち黒色で色がついた部分は、ラスタ印刷データに含まれるビットマップ領域を表す。プリントデータ生成部206はラスタ印刷データとラスタ化されたラスタデータ1(オブジェクト領域データ)とを対比し、ラスタデータ1の各ピクセルが黒色であればビットマップ領域、黒色でなければそれ以外の領域と判定する事ができる。こうしてラスタ印刷データ上における、特定すべきオブジェクトを特定することができる。例えば、プリントデータ生成部206が、内部的に起動したラスタデータ生成部203によって先の図6(b)のPDFデータ2を「2ページ/枚」の設定でラスタライズすると、図6(c)のようなラスタデータ1を取得することができる。図6(c)のラスタ化されたオブジェクト領域データ606は、図6(d)のラスタ印刷データ607と同じページ構成を持つラスタデータである。

40

【0030】

プリントデータ生成部206は、次にS510、S511にて、得られたビットマップ領域判定用のラスタデータ1と通常の印刷データであるラスタ印刷データに対し、領域判

50

定に応じた処理を行なう。ここで領域判定に応じた処理は、先ほど述べたような光沢インクをビットマップ領域のみに塗布する処理とすると、領域判定用のラスタデータ1は、黒色のピクセルがビットマップ領域を表し、そのまま塗布する領域データとして用いることができる。そのため、ラスタデータ1を光沢インク用の成分データとしてラスタ印刷データに統合する処理だけを行なうものとする。この光沢インク用の成分データに応じた画像が、プリンタA2の持つ光沢インク用ヘッドにより記録されることになる。また、ラスタ印刷データ607に応じた画像が、C(シアン)M(マゼンタ)Y(イエロー)K(ブラック)の通常インク用ヘッドにより記録される。そして、プリントデータ生成部206は、S512にて、不要になったPDFデータ1を削除する。最後に、プリントデータ生成部206は、S513にて光沢インクの塗布する領域データが統合されたラスタ印刷データをプリントデータへ変換し、S514にて出力して処理を終了する。こうして作成されたプリントデータは、最終的にバックエンド207を通してプリンタへ送られ、印刷が行われる。なお本例では、S513では、光沢インク用の成分データが付加されていない場合には、ページ全体に光沢インクを塗布すると判定し、光沢インク用の成分データをページ全面にわたり記録する。

10

【0031】

このように、プリンタドライバは、印刷データのビットマップ領域に矩形を描画したPDFデータ2を生成し、さらにOS202のラスタデータ生成部203によってPDFデータ2を領域判定用のラスタデータに変換し参照する。こうすることで、プリンタドライバは、ラスタデータ生成部203によって適用される様々なページに係る設定に依存せずに、印刷データのビットマップ領域を判定することが可能となる。また、色のついた矩形を描画することで、領域判定の対象となるビットマップオブジェクトの背景色とページの色が同色の場合でも、正確に領域を判定する事ができる。さらに、本発明を適用すれば、領域判定に応じた処理として、ビットマップ領域に特殊なインクを塗布する処理だけでなく、ビットマップ領域の背景に色をつけるような処理や誤差拡散処理、また色処理を切り替えるようなことも可能となる。

20

【0032】

[実施形態2]

次に第2の実施形態について説明する。なお以下の説明では、第1の実施形態と共に通する図及びその説明は省略し、異なる点についてのみ説明する。

30

【0033】

実施形態1では、プリンタドライバ204のプリントデータ生成部206が、ビットマップ領域に矩形を描画した領域判定用のPDFデータ2の生成処理(S508)と、PDFデータ2を領域判定用のラスタデータ2に変換する処理(S509)とを行なった。実施形態2は、実施形態1の变形であり、これらの処理をプリンタドライバ204のデータ変換部205が行なうものである。

【0034】

図7は、第2の実施形態における印刷データの処理順序を説明するブロック図である。同図は、データ変換部205とラスタデータ生成部203間のデータフロー701と、データ変換部205からプリントデータ生成部206へのデータフロー702とを持つ。データ変換部205は、PDF印刷データの内部コマンドを解析し、PDFデータ2を生成する処理(S508)と、内部的に起動したラスタデータ生成部203によってPDFデータ2をラスタデータ1に変換する処理(S509)を行なう。これら処理は第1の実施形態においてプリントデータ生成部206が行なった処理である。そして、データ変換部205は、生成した領域判定用のラスタデータ1を外部記憶装置104に保存する。図6のデータの例であれば、第1の実施形態では、図6(a)のようなPDF印刷データが外部記憶装置104に保存されたのに対し、本実施形態では、図6(c)のようなラスタデータが外部記憶装置104に保存される。次に、プリントデータ生成部206は、データ変換部205が外部記憶装置104に保存した領域判定用のラスタデータ1を取得し、その各ピクセルを参照する事で、第1の実施形態と同様に、印刷データに含まれるビットマ

40

50

ップ領域を判断することができる。

【0035】

すなわち、本実施形態では、データ変換部205は、図5(a)のステップS503に代えて、図5(b)のステップS508～S509を実行し、生成したラスタデータ1(ラスタ化したオブジェクト領域データ)を外部記憶装置104に保存する。そしてプリントデータ生成部206は図5(b)の手順のうち、ステップS508, S509を除くステップを実行する。ただし、ステップS507で取得されるデータは、PDFデータ1ではなく、保存されているラスタデータ1である。

【0036】

このように、データ変換部とプリントデータ生成部とによる機能の分担は、柔軟に変更することが可能である。そしてその変更によっても、実施形態1により奏する効果に変わりはない。

【0037】

[実施形態3]

次に第3の実施形態について説明する。なお以下の説明では、前述の実施形態と共に通する図及びその説明は省略し、異なる点についてのみ説明する。

【0038】

第3の実施形態は、第1の実施形態と第2の実施形態の变形である。本実施形態では、データ変換部205が、ビットマップ領域判定における、ビットマップ領域に矩形を描画したPDFデータ2の生成処理を行い、プリントデータ生成部206が、PDFデータ2を領域判定用のラスタデータ1に変換する処理を行なう。この場合、印刷データの処理順序を示すブロック図は、第1の実施形態を示す図3と同じになるが、データフロー301によりデータ変換部205からプリントデータ生成部206へ渡されるデータ形式が第1の実施形態とは異なる。

【0039】

データ変換部205は、PDF印刷データの内部コマンドを解析し、ビットマップ領域に内部を色で塗りつぶした矩形を描画したPDFデータ2を生成し(図5のS508に対応)、外部記憶装置104に保存する。図6のデータの例であれば、第1の実施形態では、図6(a)のようなPDF印刷データが外部記憶装置104に保存されたのに対し、本実施形態では、図6(b)のような領域判定用のPDFデータが保存される。次に、プリントデータ生成部206は、データ変換部205が保存したPDFデータ2を取得し、内部的に起動したラスタデータ生成部203によって、PDFデータ2をラスタライズし、ラスタデータ1に変換する(図5のS509に対応)。プリントデータ生成部206は、生成した領域判定用のラスタデータ1の色がついたピクセルを参照する事により、第1の実施形態と同様に、印刷データに含まれるビットマップ領域を判断することができる。

【0040】

すなわち、本実施形態では、データ変換部205は、図5(a)のステップS503に代えて、図5(b)のステップS507～S508を実行し、生成したPDFデータ2(オブジェクト領域データ)を外部記憶装置104に保存する。そしてプリントデータ生成部206は図5(b)の手順のうち、ステップS507, S508を除くステップを実行する。ただし、ステップS509で処理対象とするPDFデータ2は、データ変換部205により生成されて保存されているデータである。

【0041】

このように、データ変換部とプリントデータ生成部とによる機能の分担は、柔軟に変更することが可能である。そしてその変更によっても、実施形態1により奏する効果に変わりはない。

【0042】

[実施形態4]

次に第4の実施形態について説明する。なお以下の説明では、前述の実施形態と共に通する図及びその説明は省略し、異なる点についてのみ説明する。

10

20

30

40

50

【0043】

第4の実施形態は、プリンタドライバ204が、印刷データに含まれるビットマップ領域だけでなくテキスト領域やグラフィック領域等、他のオブジェクトの領域も判定するものである。なお、印刷データの処理順序を示すブロック図は、図3と同じ構成を持つ。また領域判定の対象となる領域にオブジェクトを描画したPDFデータ2の生成処理、および、生成したPDFデータ2を領域判定用のラスタデータ1に変換する処理は、実施形態1と同様にプリントデータ生成部206が行なうものとする。

【0044】

ビットマップ領域以外のテキスト領域やグラフィックス領域等を判定する際にも、上述した実施形態と同様に、領域判定のためのPDFデータ2を生成する段階で、PDFデータ2に対象のオブジェクトを同時に描画しておけば、その領域を判定する事ができる。図8は、この時の印刷データの例を表したものであり、ビットマップ領域に加えてテキスト領域も同時に判定する例である。なお、図8は、ビットマップ領域のみを判定する図6に対応しており、図6と同様に、図8(a)は、PDF形式の印刷データであり、図8(d)は「2ページ/枚」の設定で作成されたラスタデータ形式の印刷データである。図8(b)は、図8(a)のPDF印刷データを解析して得られるビットマップ領域とテキスト領域に色をつけたPDFデータ2であり、ビットマップ領域だけでなく、テキスト領域も同時に描画する。図8(c)は、プリントデータ生成部206が、ラスタデータ生成部203を内部的に起動し、図8(b)のPDFデータ2をラスタライズして生成する領域判定用のラスタデータ1の例である。プリントデータ生成部206は、生成したラスタデータ1のうち色のついたピクセルを参照する事で、ビットマップ領域だけでなく、テキスト領域も同時に判断することができる。なお、図8の例では、ビットマップ領域とテキスト領域は図示の都合上同じ黒色で描画している。しかし、図8(b)の領域判定用のPDFデータ2を生成する時点で、ビットマップ領域とテキスト領域を異なる色、例えばビットマップ領域は黒色、テキスト領域は赤色で描画しておけば、それぞれの領域を色によって判定することができる。

10

20

30

【0045】

すなわち本実施形態では、図5(a)のステップS502および図5(b)のステップS506において、「ビットマップ領域の判定を行なう?」との判断に代えて、「何らかのオブジェクト領域の判定を行なう?」との判断を行なう。ここで「何らかのオブジェクト」とは、ビットマップやテキスト、ラインアート等、判定対象とするオブジェクトであり、予め定められていても、ユーザによりその都度指定されてもよい。

40

【0046】

これにより本実施形態では、オブジェクトの種類としてビットマップ以外のオブジェクトについてもその領域をラスタデータから高精度に特定することができる。

【0047】

さらに、オブジェクト領域を、オブジェクトの種類毎ではなく、指定されたオブジェクトごとに特定することもできる。このためにはたとえば、ラスタデータ生成部に入力される形式のデータ(たとえばPDF)中から対象のオブジェクトを例えばユーザに選択させておく。そしてその選択されたオブジェクトを、ラスタデータ生成部の出力ラスタデータ中から特定することが可能である。

【0048】

さて、上記の実施形態1~4では、スプールファイルが、OSの標準形式であるPDF形式である場合を例に説明を行なったが、本発明はこれに限られるものではない。例えば、スプールファイル形式がアドビシステムズ社のPostScriptである場合にも効果的に適用することができる。この場合、プリンタドライバ204のデータ変換部205でPostScriptを扱えるようにすればよい。データ変換部205は、PostScript形式のデータを入出力とするものとなり、OS202のラスタデータ生成部203はPostScript形式のデータからラスタデータへ変換するものとなる。従って、プリンタドライバ204のデータ変換部205、プリントデータ生成部206で処理

50

する P D F 形式のデータを P o s t S c r i p t 形式のデータに置き換えることで容易に本発明を適用することができる。また、スプールファイルが P o s t S c r i p t 形式であっても、データ変換部 2 0 5 に P o s t S c r i p t から P D F への変換機能を備えれば、プリントデータ生成部 2 0 6 でビットマップ領域判定のために処理するデータ形式は P D F 形式にできる。上記の実施形態の要領でオブジェクト領域を特定できる。スプールファイルが P D F 形式であり、プリントデータ生成部 2 0 6 で処理するデータ形式を P o s t s c r i p t 形式にしたい場合も同様である。

【 0 0 4 9 】

また上記実施形態 1 ~ 4 では、オペレーティングシステムは M a c O S としたが、ラスタデータ生成部がオブジェクトの配置の変更を出力データに反映させる機能（例えば回転や変倍、レイアウト等）を持つ場合、M a c O S に限らず本発明は適用できる。これは、ラスタデータ生成部がオペレーティングシステムにより提供されず、別のベンダにより提供される外付け機能である場合も同様である。10

【 0 0 5 0 】

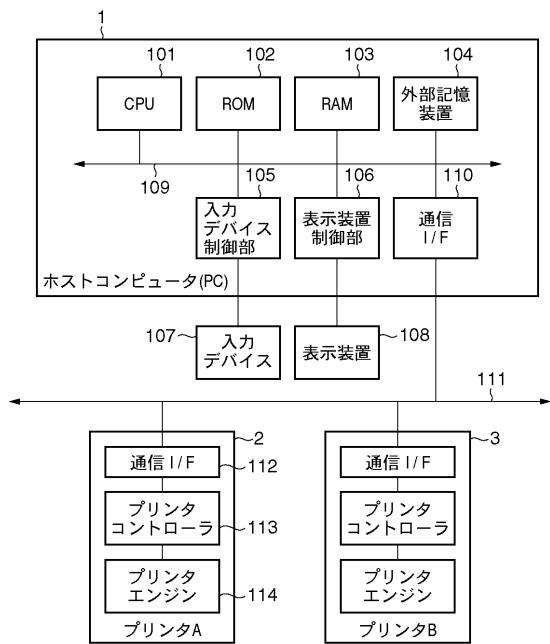
また、上記実施形態 1 ~ 4 では、ラスタデータ生成部の出力ラスタデータ中から特定されたオブジェクトに対しては、上記実施形態で説明した処理以外の処理を施すこともできる。たとえば特定されたオブジェクトが文字であり、その領域が所定のサイズよりも小さければ、他のオブジェクト（例えば X 印の文字列など）と置換するなどの処理を行なうこともできる。このように、オブジェクトの配置を変更可能なラスタデータ生成部から出力されるラスタデータを対象とし、その中の特定のオブジェクトに対して処理を施す場合、本発明によって対象のオブジェクトを特定することが容易である。20

【 0 0 5 1 】

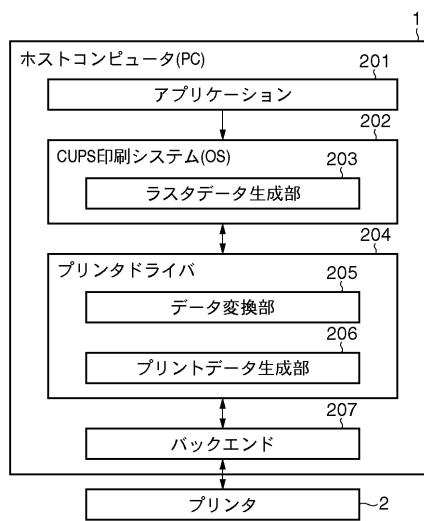
< 実施形態 5 >

また、本発明は、以下の処理を実行することによっても実現される。即ち、上述した実施形態の機能を実現するソフトウェア（プログラム）を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ（または C P U や M P U 等）がプログラムを読み出して実行する処理である。

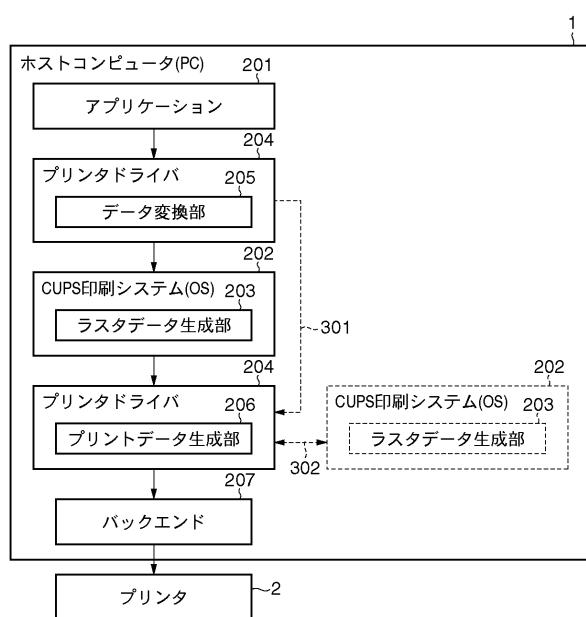
【図1】



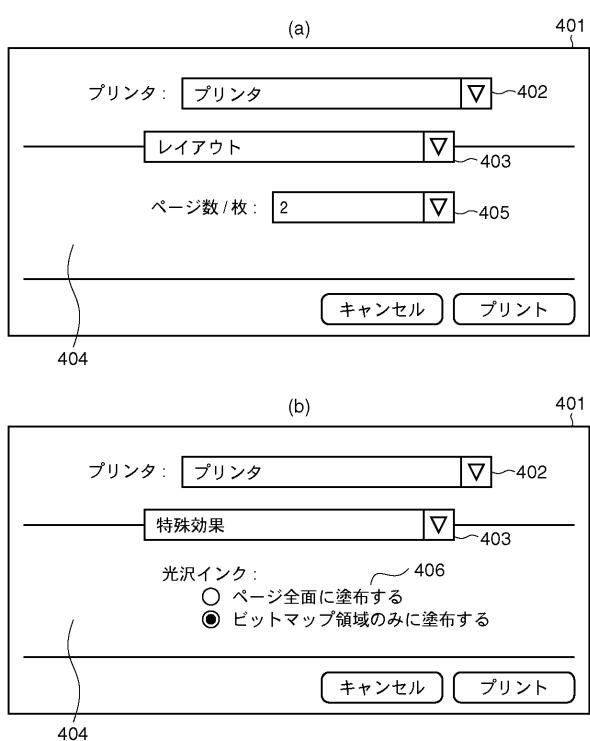
【図2】



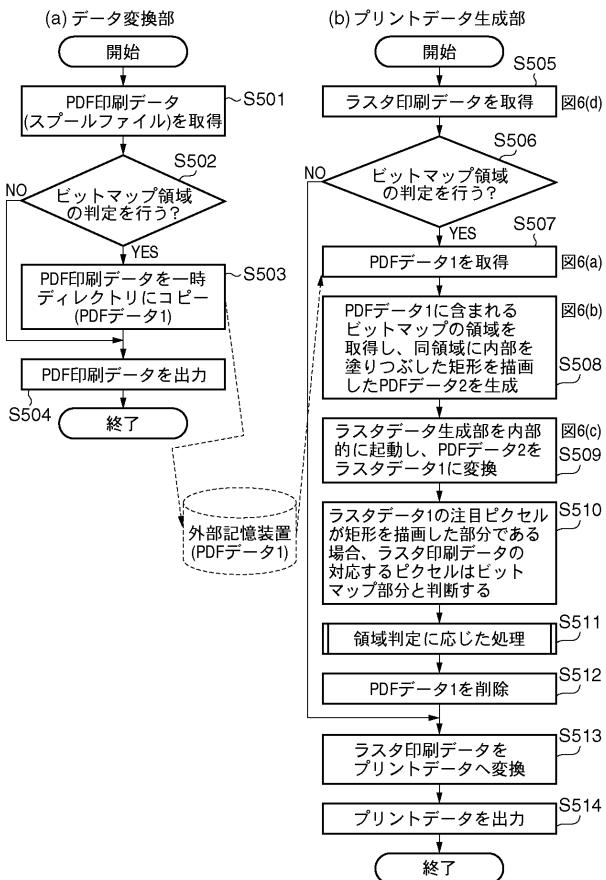
【図3】



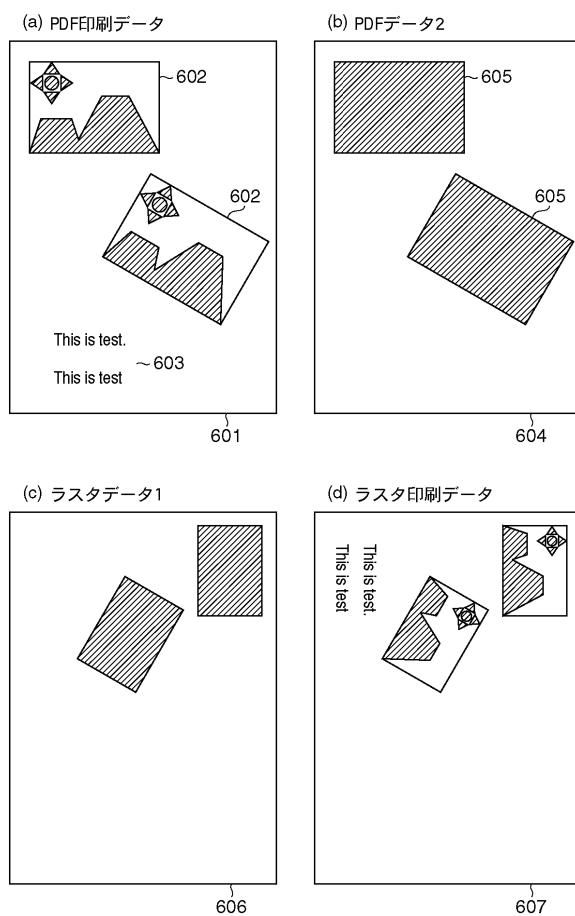
【図4】



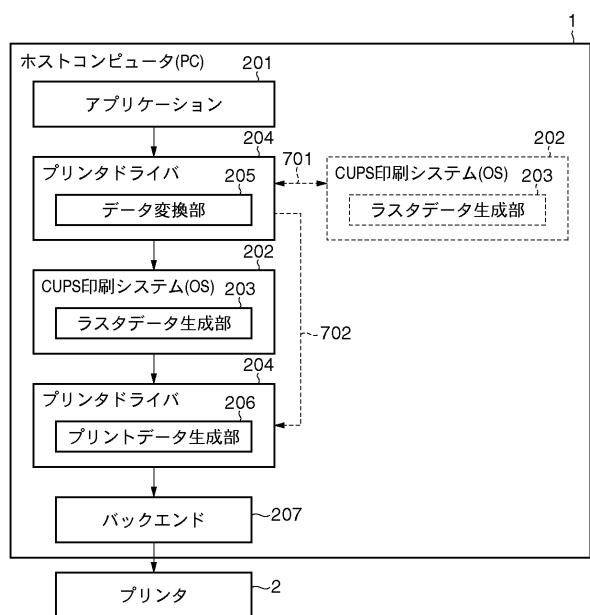
【図5】



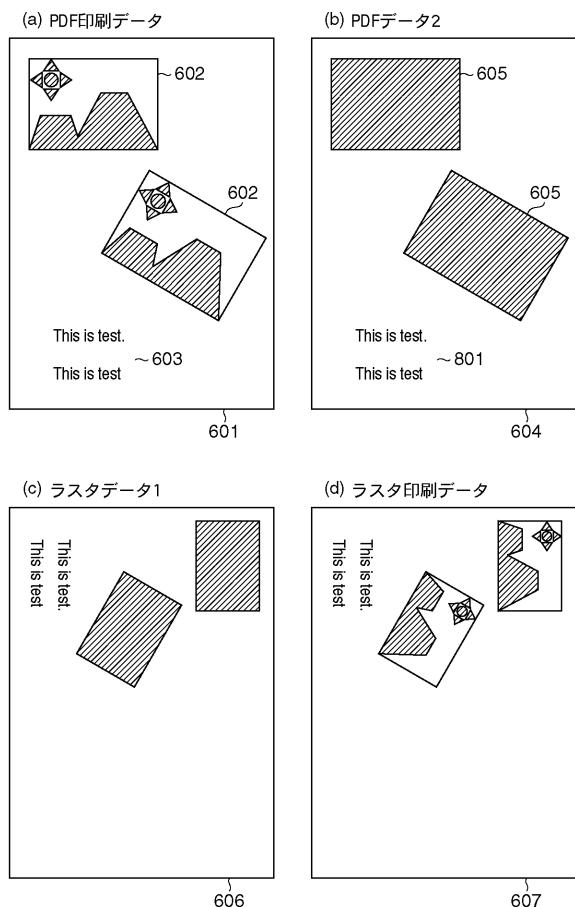
【図6】



【図7】



【図8】



フロントページの続き

(72)発明者 穴井 拓紀

東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

Fターム(参考) 2C061 AP01 AQ05