(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3566973号
(P3566973)
(45)発行日　平成16年9月15日(2004.9.15)　(24)登録日　平成16年6月18日(2004.6.18)

(51)Int.Cl.[7]		F I		
H04N	1/413	H04N	1/413	D
H04N	1/21	H04N	1/21	

請求項の数 18　(全 27 頁)

(21)出願番号	特願平5-220287
(22)出願日	平成5年9月3日(1993.9.3)
(65)公開番号	特開平6-233141
(43)公開日	平成6年8月19日(1994.8.19)
審査請求日	平成12年9月4日(2000.9.4)
審判番号	不服2003-2860(P2003-2860/J1)
審判請求日	平成15年2月21日(2003.2.21)
(31)優先権主張番号	940111
(32)優先日	平成4年9月3日(1992.9.3)
(33)優先権主張国	米国(US)

(73)特許権者　398038580
ヒューレット・パッカード・カンパニー
ＨＥＷＬＥＴＴ－ＰＡＣＫＡＲＤ　ＣＯＭＰＡＮＹ
アメリカ合衆国カリフォルニア州パロアルト　ハノーバー・ストリート　３０００
(74)代理人　100087642
弁理士　古谷　聡
(74)代理人　100076680
弁理士　溝部　孝彦
(72)発明者　ラス・キャンベル
アメリカ合衆国アイダホ州８３７０６ボイス，スピネーカー・コート・１４７３

最終頁に続く

(54)【発明の名称】メモリの最小化のための適応型データ圧縮方法

(57)【特許請求の範囲】

【請求項１】

入力データフローをページ構成出力に変換する周辺装置（１０）であって、前記ページ構成出力の一部だけを記憶するためのメモリ空間を割り当てられたランダムアクセスメモリ（２０）と、プロセッサ（１２）と、複数のデータ圧縮手順が格納された制御メモリ（２２）とを備え、前記各圧縮手順（３４、３６、３８）は性能特性を示すことからなる、前記周辺装置において、前記ランダムアクセスメモリ（２０）に記憶するために前記入力データフローの各部を圧縮する方法であって、

前記入力データフローの各部を前記ランダムアクセスメモリ（２０）の記憶領域に割り当てるステップと、

前記入力データフローの各部を割り当てるのに使用可能な前記ランダムアクセスメモリの量が不十分な場合を判定するステップ（７８）と、

前記判定するステップにおいて、前記ランダムアクセスメモリの量が不十分であると判定された場合に、前記入力データフローの一部に第１のデータ圧縮手順（８２）を使用して圧縮データ部を生成するステップであって、前記第１のデータ圧縮手順は、一般に、前記周辺装置において利用可能な他のデータ圧縮手順よりも短い時間でデータ圧縮を実行することからなる、ステップと、

前記圧縮データ部の圧縮レベルを圧縮しきい値に対して検査し、前記圧縮レベルが前記しきい値を超えているか否かを判定し（８０、４６６）、超えていない場合は、別のデータ圧縮手順（８４、８６）を使用して、あるしきい値に対して前記検査を繰り返し、これに

より、前記入力データフロー部の圧縮レベルが前記しきい値を超えることを可能にする圧縮手順を発見して、それを前記データフロー部の圧縮に使用するステップ
とからなる方法。
【請求項２】
前記データ圧縮手順（８２、８４、８６）が、各圧縮手順の性能特性の所望度が低減するレベルに基づく所定の順序で使用される、請求項１の方法。
【請求項３】
前記データ圧縮手順は、無損失方式でデータを圧縮する手順（８２、８４）と、有損失方式でデータを圧縮する少なくとも１つの他の手順（８６）を含み、無損失方式で圧縮されたデータは、データを損失することなく復元可能であり、有損失方式で圧縮されたデータの復元にはデータの損失が伴い、前記無損失圧縮手順（８２、８４）が、前記有損失圧縮手順（８６）の前にまず使用される、請求項１または２の方法。
【請求項４】
前記無損失圧縮手順の１つが、Lempel／Ziv／Welch（LZW）圧縮手順（８４）であり、該手順は、受け取ったデータストリングのテーブルを構築し、該データストリングの各々にコードを割り当てるものであり、
前記ストリングを識別するのに必要なコード長を制限するために、前記テーブル内のストリング項目の数を、最適なデータ圧縮を得るのに必要な最大値より少ない値に制限するステップであって、これにより、圧縮及び復元処理をより高速で行うことを可能にするステップ
からさらになる、請求項３の方法。
【請求項５】
前記割り当てるステップの後で、前記判定するステップの前に、
任意の前記データ部分が、所定のデータサイズより大きいか否かを判定し（１０２）、大きければ、前記ランダムアクセスメモリに残存する割り当て可能なメモリが所定のレベルより少ないかどうかをさらに判定し（１０４）、これらの判定結果がいずれも真の場合には、前記データ部分をデータ圧縮するステップ（１０６、１０８）
からさらになる、請求項１乃至４のいずれかの方法。
【請求項６】
前記データ圧縮が、無損失圧縮手順を使用して実施される（１０８）、請求項５の方法。
【請求項７】
前記データ部分がデータ圧縮された後に、該データ部分をランダムアクセスメモリに記憶するステップ（１２０、１２２）と、
前記記憶されたデータ部分によって占められたメモリ空間を前記所定のレベルから減じることによって該レベルを調整するステップ（１２４）
とからさらになる、請求項６の方法。
【請求項８】
前記周辺装置（１０）が、動作時に一定速度で動作する印刷エンジン（１４）を備えており、
ページ内のデータ部分について複雑度を計算し（９２）、該複雑度が、前記印刷エンジンの速度に関して、前記データ部分をラスタビデオフォーマットに変換するためには非常に長い時間がかかるであろうことを示しているか否かを判定するステップと、
その判定に基づいて、少なくとも前記データ部分をラスタビデオフォーマットにすぐに変換して、それに対してデータ圧縮を実行するステップ（９４）
からさらになる、請求項１乃至７のいずれかの方法。
【請求項９】
ある１つの前記データ部分の前記変換及び圧縮に続く任意の時点で、残りのデータ部分を、メモリ切れ指示を生じることなく処理することができると判定（３５８）した場合には、前記残りのデータ部分を圧縮せずにラスタビデオに変換する（３８０、３５６）、請求項８の方法。

【請求項１０】
入力データフローをページ構成出力に変換する周辺装置（１０）であって、前記データフローが、画像を表すデータを含み、前記画像が、一組の副画像から構成され、各副画像が、さらに、ビデオラスタビットマップと、該ビデオラスタビットマップの記憶場所を特定するフィールドを有する記述子を含むことからなる、周辺装置において、前記画像及び副画像を表すデータを圧縮するための方法であって、
前記一組の副画像のサブセットを表す一組の記述子を選択するステップ（４６０）と、
第１の圧縮技法を使用して、前記選択された各記述子に関連するビデオラスタビットマップの圧縮を試行するステップ（４６２）と、
前記圧縮を試行するステップにより得られた前記副画像のサブセットの圧縮レベルをあるしきい値と比較し（４６６）、前記しきい値と最低でも等しい場合には、前記第１の圧縮技法を使用して（４７２）、前記画像を構成するすべての副画像をデータ圧縮するステップ（４５０）
からなる、方法。
【請求項１１】
前記各記述子が、関連するビデオラスタビットマップに関する情報を示すために予約されたフィールドを有し、
前記圧縮レベルが前記しきい値と等しくない場合には、前記副画像のサブセットに対して第２の圧縮技法を使用し（４７２）、再度、得られた圧縮レベルをあるしきい値と比較し、該しきい値と最低でも等しい場合には、前記画像のすべての副画像に対して前記第２の圧縮技法を使用するステップ
とからさらになる、請求項１０の方法。
【請求項１２】
前記各記述子内の前記情報が、前記画像の復元の間使用され、前記情報は、前記ビデオラスタビットマップを圧縮するために使用される圧縮方法を示す（４４６）ものである、請求項１１の方法。
【請求項１３】
少なくとも前記第１の圧縮技法が無損失である、請求項１２の方法。
【請求項１４】
前記周辺装置が、ランダムアクセスメモリ（RAM）（２２）を備え、その記憶容量が、前記ページ構成出力全体を記憶するには不十分であり、
前記圧縮技法を適用した後に、圧縮された画像を記憶するのに利用可能な十分な空間が前記RAMにあるか否かを判定し（４６６）、もしなければ、前記副画像に対して有損失圧縮技法を適用するステップ（４７０）
とからさらになる、請求項１３の方法。
【請求項１５】
前記有損失圧縮技法の一部として、少なくともいくつかの前記副画像のサブセットのビデオラスタビットマップを、クラスタ化されたビット配列と分散化されたビット配列からなるビットパターンを規定する表形式のエントリと比較することにより、前記画像を、分散化されたもの、または、クラスタ化されたもの、のいずれかに分類する（２２６）ことからなる、請求項１４の方法であって、この分類は、前記有損失圧縮画像の復元の間使用されることからなる、方法。
【請求項１６】
先行して適用された無損失圧縮技法が、前記有損失圧縮技法で利用可能な前記画像の圧縮レベルを超えない場合のみ、前記画像に対して前記有損失圧縮技法が使用される（４６６、４６８）、請求項１４の方法。
【請求項１７】
前記入力データフローがフォントデータを含み、
前記フォントデータによって表されるフォントが、そのサイズにおいて、所定のレベルと最低でも等しい場合に、前記フォントデータに対して無損失圧縮技法を自動的に適用する

ステップ（４３４）
をさらに含む、請求項１の方法。
【請求項１８】
前記フォントサイズが前記所定のレベルに等しくない場合は、前記入力データフローを記憶するためのランダムアクセスメモリの容量が不十分である場合にのみ、前記無損失圧縮技法を適用する（４３６）、請求項１７の方法。
【発明の詳細な説明】
【０００１】
【産業上の利用分野】
本発明はページプリンタに関し、特に、フルページのプリンタデータに必要なものより小さいバッファメモリサイズに適応させるためのデータ圧縮機能を備えたページプリンタに関する。
【０００２】
【従来の技術】
従来のページプリンタは、通常は、画像を紙上に印刷する前にページ全体を捕捉する。かかるプリンタでは、フォーマッティングは、ホストコンピュータ上で実行される（大量のラスタ化データがプリンタに送られる）か、又はプリンタのフォーマッタ上で行われる。レーザー印刷「エンジン」は一定速度で動作するので、このエンジンの動作に追い付くだけの速度で新しいラスタ化データが得られない場合には印刷の「オーバーラン」が発生し、ページ印刷が不能になる。
【０００３】
印刷オーバーランを防止するために様々な技術が用いられている。ある技術では、印刷機構が常に動作待ち状態のラスタ化データを持つように或るページ全体の全てのラスタビットマップが記憶される。３００ドット／インチの解像度では、この技術には各ページにつき約１Ｍバイトのラスタメモリが必要である。６００ドット／インチでは、４Ｍバイトのメモリが必要である。更に、レーザープリンタは、ラスタデータのパイプライン処理によりその定格速度を達成するので、プリンタをその定格速度で動作させるためには別のラスタメモリが必要となる。さもなければ、前のページがプリンタの出力トレーに排出されるまで次のページの作成を開始することができない。レーザープリンタの価格を低レベルに抑えるために、必要なラスタメモリ量を削減する努力が払われてきた。
【０００４】
メモリ低減の技術の１つとして、ページ記述の構築がある。ページ記述は２つのステップで構築される。即ち、第１のステップでは、フォーマッティング中に、ホストコンピュータから受信したデータが、何を印刷すべきかを記述する表示コマンドと呼ばれる簡単なコマンドのリストに変換される。第２のステップでは、印刷用の表示コマンドリストが準備され、表示コマンドの構文解析と記述されたオブジェクトのラスタビットマップへの変換とが行われる。この手順には、フルページのラスタビットマップメモリが必要である。これは、連続するページに同じメモリが用いられるからである。
【０００５】
必要なメモリ量を削減するために、この表示コマンドリストによる手順は、各表示コマンドをそのページ上の垂直位置に従ってソートすることにより修正された。ページは次いでページストリップ又は「ページ中間体（ｉｎｔｅｒｍｅｄｉａｔｅ）」と呼ばれるセクションに分割され、その各ページストリップが印刷を行うために印刷エンジンに順次送られる。あるページストリップ中の表示コマンドが充分な速度でラスタ化データに変換される際、最初のページストリップの記憶に用いられた同じメモリを（この最初のページストリップが紙上に転写されれば）そのページ中の下方の後続ページストリップに対して再使用することができる。
【０００６】
本出願人が製造するレーザープリンタ　Ｌａｓｅｒ　Ｊｅｔ　ＩＩＩ　には、表示コマンドリストアルゴリズムが採用されており、３つのラスタバッファが用いられている。第１

バッファはページの第４ストリップについて再使用され、第２バッファは第５ストリップについて再使用される、といった具合である。８ページ／分のプリンタの場合、ストリップ１の終了と印刷機構がストリップ４を要求する時間との間にはわずかな時間しか存在しない。ストリップが時間内に供給されないと、「印刷オーバーラン」が発生し、ページは適正に印刷されない。

【０００７】

本出願人による「Ｍｅｔｈｏｄ ａｎｄ Ａｐｐａｒａｔｕｓ Ｆｏｒ Ｐｒｅｖｅｎｔｉｎｇ Ｐｒｉｎｔ Ｏｖｅｒｒｕｎｓ」と題する米国特許出願第０７／７０１，２３５号では、ページプリンタが各ストリップ中の全ての表示コマンドについて総ラスタ化実行時間を判定する。ストリップのラスタ化実行時間が所定値を超えた場合、印刷機構への次のデータ転送に用する時間を削減するためにストリップが事前ラスタ化される。なお、上記引用をもって上記特許出願に係る発明の開示内容を本明細書に包含させたものとし、その詳細な説明は省略する。

【０００８】

最近では、６００ドット／インチの解像度のプリンタが市場に導入されている。かかるプリンタは、テキストだけでなくラインアートや各種の画像も扱うものである。かかるプリンタに要するメモリ量を最小限にするために、データ圧縮技術が用いられている。例えば、プリンタへのデータ転送処理に、ホストプロセッサによりランレングス（ｒｕｎ ｌｅｎｇｔｈ）データ圧縮が用いられている。ランレングスエンコード法では、繰り返し現われるデータは、そのデータの同一性と繰り返しのランレングスとを示すことによりエンコードされる。ランレングス圧縮は、テキスト及びラインアートに用いる場合には成功する。しかし、画像データに用いる場合には、かかる圧縮法ははるかに不充分なものとなる。

【０００９】

ある種の画像は「順序付（ｏｒｄｅｒｅｄ）ディザ」又は「誤差拡散（ｅｒｒｏｒ ｄｉｆｆｕｓｅｄ）」のいずれかに分類される。順序付ディザ画像（「クラスタ化画像」とも呼ばれる）は、ページ全体にわたり中間調グレー表示を含む中間調画像である。かかる画像は、一般に、かなりのデータ冗長を反映し、無損失データエンコード技術（例えばランレングス）に資するものである。これに対し、誤差拡散（「分散（ｄｉｓｐｅｒｓｅ）」とも呼ぶ）画像はデータの冗長性がほとんどなく、異なる圧縮法を必要とする。その結果、ページプリンタに１つの圧縮法だけを用いると、かかるプリンタは、最小限のオンボードラスタメモリ量を維持しつつ画像データを扱うことができなくなる。

【００１０】

ランレングスエンコードに加え、従来の技術には他のデータ圧縮法も多数存在する。例えば、Ｗｅｌｃｈの米国特許第４，５５８，３０２号には、周知のＬｅｍｐｅｌ／Ｚｉｖ／Ｗｅｌｃｈ（ＬＺＷ）データ圧縮技術が説明されている。この方法では、入力データストリーム中で遭遇する文字ストリングをストリングテーブルに記憶することにより、入力データストリームを圧縮する。「圧縮器」が、入力ストリームを探索して、記憶されたストリングと一致する最長のものを判定する。圧縮器は、多くのストリングに遭遇するに連れて「賢く」なっていき、次第により長い文字を圧縮できるようになる。入力文字ストリングが記憶されたテーブルストリングに一致するたびにコードが受信器に送出される。その受信器には、その送信のデコードを可能にする同一のストリングテーブルが存在する。

【００１１】

プリンタのラスタ走査線についてのランレングスデータ圧縮が、Ｓｐｉｖｅｙによる「Ｄａｔａ Ｃｏｍｐｒｅｓｓｉｏｎ Ｔｅｃｈｎｉｑｕｅ ｆｏｒ ＡＰＡ Ｐｒｉｎｔｅｒ（Ｃｈａｎｇｅ Ｂｌｏｃｋ Ｓｋｉｐｐｉｎｇ）」及び「Ｈｙｂｒｉｄ Ｄａｔａ Ｃｏｍｐｒｅｓｓｉｏｎ Ｔｅｃｈｎｉｑｕｅ ｆｏｒ Ｃｈａｎｇｅ Ｂｌｏｃｋ Ｓｋｉｐｐｉｎｇ ｉｎ ａｎ ＡＰＡ Ｐｒｉｎｔｅｒ」（ＩＢＭ Ｔｅｃｈｎｉｃａｌ Ｄｉｓｃｌｏｓｕｒｅ Ｂｕｌｌｅｔｉｎ，ｖｏｌ．２３，Ｎｏ．１２，Ｍａｙ １９８１，５４６４－５４７０頁）に説明されている。Ｓｐｉｖｅｙの方法では、現在の走査線を前の画像における対応する走査線と比較し、その間の一致の有無に応じてコー

ドを出力する。Ｓｐｉｖｅｙはまた、前述の方法をランレングスエンコードと組み合わせて用いる複合技術を説明している。

【００１２】

【発明が解決しようとする課題】

従って、本発明の目的は、データ圧縮用のシステムを含み、受信データの特性に応じて適応可能なページプリンタを提供することである。

【００１３】

本発明の別の目的は、限られた量のオンボードメモリを効率的に用いることを可能にするようにデータ圧縮技術を知的に（ｉｎｔｅｌｌｉｇｅｎｔｌｙ）選択するシステムを採用したページプリンタを提供することである。

【００１４】

また、本発明の目的は、データ圧縮が行われるまで印刷を継続することができない状態にオンボードメモリがおかれた際に選択的に呼び出されるデータ圧縮システムを備えたページプリンタを提供することである。

【００１５】

更に、本発明の他の目的は、ページプリンタで使用するための改良されたデータ圧縮技術を提供することである。

【００１６】

【課題を解決するための手段】

周辺装置は、入力データフローをページ構成された出力に変換し、ページ全体のラスタデータの記憶にはサイズ的に不充分であるランダムアクセスメモリ容量を有するものである。この周辺装置はまた、プロセッサと複数のデータ圧縮手順を保持する制御用メモリとを有し、その各手順は異なる性能特性を示す。周辺装置は、以下のステップを含む入力データフローの各部を圧縮する方法を実行する。即ち、その方法は、入力データフローの各部にランダムアクセスメモリを割り当て、かかる割り当てに対して使用できるランダムアクセスメモリの量の不足を判定し、入力データフローの各部に対して第１のデータ圧縮手順を用いて圧縮データ部を生成し、この圧縮データ部を検査してしきい値を超える或るレベルの圧縮が達成されたか否かを判定し、それが達成されていなければ次のデータ圧縮手順を用いて、このしきい値に対する各手順毎に検査を繰り返し、これにより、しきい値の圧縮レベルの達成を最初に可能にする圧縮手順をそのデータフロー部の圧縮に用いる圧縮手順とする、というステップを含むものである。また、総体的（ｉｎｔｅｇｒａｌ）及び独立の画像記述子を用いた入力データフローを扱うための改良された圧縮方法及び圧縮技術が説明される。

【００１７】

【実施例】

以下、本発明をレーザプリンタの場合について説明するが、プロッタ、ファクシミリ装置等の他の周辺装置にも本発明で考案された手順を利用可能であることが理解されよう。

【００１８】

プリンタシステム

図１において、プリンタ１０は、プロセッサ１２、印刷エンジン１４及び入出力（Ｉ／Ｏ）ポート１６を含み、これらは全てバス１８で接続されている。印刷エンジン１４は、一定速度で動作するレーザプリンタを有し、このレーザプリンタにはその動作についていける速度でビデオラスタ印刷データが提供されねばならない。また、ランダムアクセスメモリ（ＲＡＭ）２０及びリードオンリーメモリ（ＲＯＭ）２２がバス１８に接続されており、それらは、利用可能なＲＡＭ　２０が最も効率的に使用されること、及び印刷オーバーランを防止するために印刷エンジン１４が印刷待ち状態のデータを有することを確実にするために必要な全ての手順を含む。

【００１９】

この実施例では、プリンタ１０は、ページプリンタであり、Ｉ／Ｏ　モジュール１６を介してホストプロセッサから画像データを受信する。受信した画像データの記憶に利用可能

なＲＡＭ　２０の記憶容量は、１ページ全てのビデオラスタビットマップ画像の記憶に必要な容量よりかなり小さいものである。ＲＡＭ　２０のサイズは、プリンタ１０が競争力のある最低価格で販売しうるよう制限されている。

【００２０】

ホストプロセッサからの画像データは、Ｉ／Ｏ　モジュール１６を介して受信され、ＲＡＭ　２０に読み込まれる。この画像データはいくつかの形態のうちの１つをとる。その画像データは、副画像を含み、この場合、記述子が各副画像と連続していて各副画像と共に処理される。かかる画像データが受信されると、この画像データ自体が調べられて、データ圧縮が必要であるか否かが判定される。本発明の第２の態様では、副画像の記述子が実際の副画像データとは別個に受信される（ただし、実際の副画像データにリンクされている）。第２の態様では、記述子中のデータと他の画像固有のデータとを用いて、一連のデータ圧縮技術（もしあれば）のどれをこの副画像データに適用すべきかの判定の一助にする。

【００２１】

本発明のいずれの態様においても、プリンタ１０が画像データを受信すると、構文解析が行われ、コマンド及びデータが処理されて、ＲＡＭ　２０に記憶されるラスタ図形データが導出される。この際に副画像が生成される。次に、ラスタ図形データベースに記憶された副画像が中間ページ表現（「ページ中間体」）にリンクされる。ページ中間体は、最小限のメモリしか消費せずに１ページ分の情報をビデオに簡単に変換することのできるフォーマットで表現する手段である。ページ中間体の動作は基本的には１ページを複数のストリップに分割し、その各ストリップは所定数の「命令バケット」を含む。各ストリップは、高さ（例えば１２８ドット）が割り当てられ、一連の命令バケットへのポインタ（例えば、各バケットのサイズは２５６バイト）を含み、各「バケット」はストリップに対してラスタビデオをどのように生成すべきかを定義する命令を含む。２つ以上のストリップにまたがる副画像（及び他の中間体オブジェクト）は、それが存在する各ストリップ毎に１つの命令を有する。

【００２２】

ラスタ図形データのページ中間体への変換においては、ストリップの内容がそのストリップのビデオバッファのサイズに基づくしきい値を超えることが明らかになる場合がある。また、複数のストリップの変換後に、ページ中間体の変換を完了するだけのメモリが残っていないという事態も発生し得る。更に、１ページ全体を変換した後に、このページが利用可能なＲＡＭに記憶するには複雑すぎることが明らかになる場合がある。上記のいずれの場合にも、このシステムはそのメモリの制約を克服しようとしてデータ圧縮動作を実行する。各場合において、そのとき存在する変数の検討結果に応じて複数の圧縮技術のうちの１つが選択される。このようにして、利用可能なＲＡＭを最も高速かつ効率的に使用することができるように圧縮手順が選択される。

【００２３】

以下に３つのデータ圧縮法を説明する。使用すべきデータ圧縮法の選択がプリンタ機構から求められる性能特性によって決まることが理解されよう。最高速の圧縮を行う場合には、モードｍと呼ばれる手順（ｍは多次元変更（ｍｕｌｔｉ－ｄｉｍｅｎｓｉｏｎｃｈａｎｇｅ）を意味する）が、ラスタ走査行に加えられるランレングスと、行から行への変化とに基づいてラスタ画像データブロックを圧縮する。モードｍは極めて高速なデータ圧縮（及び圧縮解除。本明細書では、圧縮解除とは圧縮された画像を解凍すなわち復元することを意味する）法であるが、ある種の画像データの圧縮には有効ではない。モードｍは「無損失」圧縮技術（この圧縮中には、圧縮解除時に回復不能な画像データの損失は存在しない）である。

【００２４】

また、モードｍより低速であるが、ある種の画像データに対してはより効率的である別の無損失圧縮技術が、改良型のＬｅｍｐｅｌ／Ｚｉｖ／Ｗｅｌｃｈ（ＬＺＷ）手順である。第３の「有損失」圧縮手順が採用され、これは一般に少なくとも４：１の圧縮率を保証す

るが、圧縮解除時に回復不能な画像データの損失が幾分生じることになる。

【００２５】

ここで図１に戻る。このデータ変換及び圧縮手順に用いられるサブルーチンはＲＡＭ　２０及びＲＯＭ　２２にリストされている。ＲＡＭ　２０は、ユーザに割り当てられたメモリ領域２４と、ラスタ画像の記憶に割り当てられた領域２６と、圧縮された画像に割り当てられた領域２８と、ページ中間体データの記憶のための領域３０と、以下で説明する手順に用いられる変数の記憶に割り当てられた領域３２とを有する。

【００２６】

ＲＯＭ　２２は、ＬＺＷ圧縮（３４）と、モードm圧縮（３６）と、有損失圧縮（３８）とを実行するサブルーチンを含む。更にＲＯＭ　２２は、「ストリップビデオ事前生成」（ＳＶＰＧ）を実行するための手順を記憶する領域４０と、「適応型圧縮ページ保護」（ＡＣＰＰ）を実行するための手順を備えた領域４２と、圧縮解除時に有損失圧縮解除の行われた画像が元の画像を（データ損失があるにもかかわらず）より有効に表現し得るようにする有損失ディザテーブルを記憶する領域４４と、領域３４，３６，３８で発見された１つ以上の圧縮手順を用いてデータ圧縮を行うための手順を記憶する領域４６とを有する。

【００２７】

モードm圧縮を実行する手順の動作中には、複数のカウンタ５０，５２，５４を用いて或る圧縮カウントの追跡を行うことが可能である。かかるカウンタの動作を以下で考察する。

【００２８】

概略フローチャート

ここで図２を参照する。同図は、１ページの生成過程中にプリンタ１０が実行する動作を示す概略フローチャートである。まず、データがホストプロセッサから受信されて（ステップ７０）、ラスタ図形副画像を導出するための事前処理を受ける。次に、この副画像を用いて画像データのページ中間体ストリップが形成される（ステップ７２）。上述のように、この手順は、図形副画像のページストリップへの割り当てを含み、各ストリップは命令バケットからなる。その命令バケットは、それぞれが所定サイズを有し、恐らくは所定サイズの副画像命令を含むものである。プリンタ１０はまた、（一般には３つの）ページ中間体のストリップの変換の結果として発生するラスタビデオビットマップを保持するためにＲＡＭ　２０の領域３０を事前割り当てする。従って、プリンタ１０は１ページを１０以上のストリップに分割することができるが、ＲＡＭ　２０はページ中間体の３つのストリップから結果的に得られるラスタビデオビットマップしか保持することができない。

【００２９】

各ページ中間体ストリップが完成すると、プリンタ１０は、各ストリップ中の命令をラスタビデオビットマップデータに変換し、このデータをＲＡＭ領域２６に記憶させる。このＲＡＭ領域２６は（本実施例では）３つのラスタバッファ領域に細分化されている。３つのバッファ領域の全て（ＲＡＭ領域２６）が一杯の場合、印刷エンジン１４が始動し、そこに記憶されているラスタビデオ情報がプロセッサ１２の制御下で印刷エンジン１４へと供給される。ラスタビデオデータが印刷エンジン１４に読み込まれている間には、印刷エンジン１４へのデータ転送が完了したバッファ領域を用いてページ中間体のストリップバッファビデオへの変換処理が続行される。

【００３０】

ページ中間体の形成過程において、１つのストリップに対して命令数が多すぎる場合がある（判定ステップ７４）。この場合には、大きすぎると判明したラスタ図形部分は直ちにビデオラスタデータに変換され、無損失圧縮技術、例えばモードｍを用いて圧縮される（ステップ７６）。この手順の詳細については図３を参照して後で説明する。

【００３１】

ページ中間体の形成中には、利用可能なストリップラスタバッファ領域２６の量が追跡される。残りのストリップラスタバッファ領域が「少ない」又は「無し」と判定された場合

（判定ステップ７８）、この手順は、ページ中間体命令の指示する図形画像データをデータ圧縮してメモリの空部分を作ることができるか否かを判定するために分岐する。このため、この手順は圧縮技術の選択へと移行する（ステップ８０）。最初に、モードm無損失圧縮を試行し（ステップ８２）、これによりメモリ切れ又はメモリ不足状態が解消された場合、この手順はこの処理を出てページ中間体の形成を続行する（ステップ７２）。一般に、メモリ切れ又はメモリ不足状態が継続する場合には、無損失ＬＺＷ圧縮手順が用いられて（ステップ８４）、メモリ切れ／不足検査が繰り返される。ここでメモリ切れ／不足指示が残る場合には、有損失圧縮手順（ステップ８６）が試行され、再びその結果が検査されて、メモリ切れ又は不足状態が解消されたか否かが判定される。メモリ切れ又は不足状態が継続する場合、そのページは印刷することができない。しかし、ほとんどの場合、有損失圧縮技術は、通常は４：１のデータ圧縮を行うので、メモリ切れ又は不足状態が解消される。

【００３２】

ページ中間体ストリップ処理が完了すると（判定ステップ９０）、ページ中間体データが構文解析されて、そのデータをビデオラスタビットマップデータに変換するのに要する時間が見積もられる。この見積もり時間が印刷エンジン１４の許容可能な時間より長い場合、適応型圧縮ページ保護（ＡＣＰＰ）が実施されて「ページが複雑すぎる」という指示が発せられる（判定ステップ９２）。かかる条件下で、高レベルの複雑性を有するページ中間体のストリップがページ中を下に向かって順に処理されて、ラスタに変換され、一時バッファに入れられる。次いで、そのラスタデータストリップに圧縮技術を用いて、「ページが複雑すぎる」という指示の除去、及び利用可能なＲＡＭへの既変換ストリップの適合化を試みる。

【００３３】

各ストリップが変換されて（可能であれば）圧縮されると、その記憶領域のＲＡＭ　２０中における発見を試みる。圧縮されたオブジェクトの複雑性は一般に印刷エンジン１４に許容される最大の複雑性より遥かに低いため、印刷オーバーランを起こすことなく１ページが印刷される。

【００３４】

ストリップビデオ事前再生

ここで図３を参照して（図２のステップ７４，７６に示す）ストリップビデオ事前生成中に用いられる手順を説明する。ページがオープンされると、各ストリップ毎にページ中間体のサイズのしきい値が設定される（ステップ１００）。このしきい値は、バケット単位で設定され、１バケットはページ中間体の記憶領域の最小の割り当て単位である。この実施例では、１バケットが約２５６バイト長であると仮定する。各バケットは、必要であれば、ストリップのページ中間体データの記憶のためにリンクされる。バケットのしきい値は、利用可能なプリンタのメモリ量を、１ページ上のストリップ数と１バケット当たりのバイト数との積で除算することにより算出される。利用可能なメモリを約８００，０００バイトと仮定すると、前記しきい値は約１２５バケットに設定され、各ストリップのしきい値がその値に初期設定される。

【００３５】

次に、ページ中間体処理が開始されて（ステップ７２）、ページ中間体データ構造が生成される。ラスタ図形のページ中間体への変換処理中の任意時間において、或る１つのストリップについての命令バケット数が予め設定されたしきい値を超えていると判定された場合、そのストリップを直ちにラスタに変換し圧縮してプリンタメモリ量を一定に保たねばならない場合がある。従って、しきい値を超えた場合には判定ステップ１０２により出力ＹＥＳが提供される。次いで、ストリップラスタビットマップ用の残りのＲＡＭが２つのストリップのためのスペースより小さいか否かが判定される（判定ステップ１０４）。その答がＮＯである場合、ページ中間体の生成が続行される。しかし、その答がＹＥＳである（２つのストリップを保持するにはＲＡＭ残量が不充分である）場合には、そのしきい値を超えるストリップは、そのページ中間体データが直ちにラスタビデオデータに変換さ

れる（ステップ１０６）。
【００３６】
判定ステップ１０４に示す条件により、変換を伴わずに一層多くのページ中間体の記憶を行うのに充分なメモリが存在する際にストリップがラスタビデオデータに変換されることを防止することができる。従って、ビデオ事前生成を必要としないページは、そのページ上の２、３のストリップが非常に大きい（他は小さい）というだけで時間的不利益を払わねばならないということはない。
【００３７】
「大きすぎる」ストリップがラスタビデオに変換されると、以下で説明するように、そのストリップのデータのモードm圧縮を用いた圧縮が試行される（ステップ１０８）。モードm圧縮動作の結果としてストリップビデオデータが圧縮されている場合には（判定ステップ１１０）、その既圧縮ストリップが用いられる（ステップ１１２）。また、ストリップが圧縮されていない場合には、その未圧縮ストリップが用いられる（ステップ１１４）。或る条件下では、圧縮技術を用いることによりストリップデータの拡張が発生することになる、という点に留意されたい。このような場合、拡張されたストリップデータの代わりに未圧縮ストリップデータが用いられる。
【００３８】
次に、ストリップラスタビデオ用のスペースのＲＡＭ　２０における割り当てが試行される（ステップ１１６）。連続したメモリスペースが利用可能である場合（判定ステップ１１８）、ストリップはそのメモリスペースに記憶される（ステップ１２０）。また、メモリ中に連続したスペースが見つからず、充分なフラグメント（断片）メモリスペースが利用可能である場合（判定ステップ１１９）には、ストリップはＲＡＭ　２０中のリンクされたメモリフラグメントに記憶される（ステップ１２２）。更に、連続した又は断片的なメモリスペースが不充分にしか存在しない場合には、「メモリ切れ」が発生し、そのページの印刷は、印刷できるものを印刷する、という状態で続行される（ステップ１２１）。
【００３９】
ストリップがメモリに記憶されると、現在のメモリ状態に基づき、そのストリップ用の新しいバケットしきい値が確定される（ステップ１２４）。このしきい値は初期のしきい値と同様に計算される。ただし、変換されたばかりのストリップが消費したバケット数が引かれることになる。従って、新しいしきい値は、（利用可能なＲＡＭ量の変化に従って調整された）古いしきい値から既変換ビデオラスタストリップのサイズを減算し、１バケット当たりのバイト数で除算したものに等しい。この調整により、しきい値は、ストリップが変換されて利用可能なメモリが少なくなるにつれて小さくなる。新しいしきい値が設定されると（ステップ１２４）、ページ中間体の生成が続行される（ステップ７２）。
【００４０】
図３に示す手順は、メモリ切れ又はメモリ不足の指示に先立って発生する。この手順は、ページ中間体データの複雑なストリップの存在の結果として発生するメモリ上の制約を予測するはたらきをする。しかし、ページ中間体の処理中に、どのストリップも１つのみではストリップビデオ事前生成しきい値を超えない状態で、複数の複雑なストリップがメモリ切れ又はメモリ不足の指示を発生させることになる可能性がある。このような場合、この手順はプリンタ１０で利用可能な圧縮法から１つを選択してページ中間体のラスタ図形データの圧縮を試行する。
【００４１】
圧縮手順の選択を可能にする判定手順を説明する前に、ＲＯＭ　２２に含まれる各圧縮手順について考察する。図４を参照して先ずモードm圧縮について考察する。
【００４２】
モードm圧縮
モードm圧縮は、ページ中間体データのラスタビットマップブロックを考慮し、各行毎にランレングスエンコードを用い、及び、ブロック中の行の間から発生するデルタ変化のエンコードにより、そのブロックを圧縮しようとする。ここで用いる「行」という用語は、

任意長のデータを意味し、そのデータのサイズは画像のラスタ走査行のサイズに等しくても等しくなくてもよい。

【００４３】

モードm手順は、図１に示す「同一」カウンタ５０、「異種」カウンタ５２、及び「ラン」カウンタ５４を用いる。モードm圧縮手順は、ページ中間体の副画像ブロックにアクセスし、この手順に対して２つの初期条件即ち「同一」と「異種」との一方を設定することにより開始される。「同一」初期条件は、ブロックの行のあるワードが前の行の垂直方向に並んだワードと同一である毎にカウントを累算させることを可能にする。このカウントは「同一」カウンタ５０に入力される。これに対し、「異種」条件が有効である場合には、１つの行のあるワードが前の行の垂直方向に並んだワードと異なる毎に異種カウンタ５２がインクリメントされる。第３のカウンタ即ちランカウンタ５４は、ある行に現われる同一ワードのランレングスをカウントするのに用いられる。

【００４４】

ここで図４に戻る。初期条件は「同一」に設定されていると仮定する（ステップ１５０）。副画像ブロックの第１の行が次いで考慮され、それに対し周知の方法でランレングス圧縮が行われる（ステップ１５２）。何らかのランレングスの文字が存在する限り、各「ラン」文字及び「ラン」カウントについてランレングスコマンドが発せられる。更に、その行の残りの文字は（以下で説明する）同一／異種コマンド中の「異種」ワードとして出力される。この場合、同一の値は０である。更に（ステップ１５４に示すように）、第１のブロック行中の各ワードは（最初の行の上には行が存在しないので）明らかに「異種」であるから、異種カウンタ５２において異種カウントがインクリメントされ、その行の最後に（及び第１行中の各ランの最後に）リセットされる。次いで、次行がアクセスされ、左側から各ワードが調べられて、それが、直前の行中の垂直方向に並んだワードと同じであるか否かが判定される。「同一」の条件が満足される限り、「同一」カウンタ５０は、前の行中の垂直方向に並んだワードと異なるワードが現われるまでインクリメントされる（ステップ１５６）。

【００４５】

次いで、この手順は「異種」モードに切り換わり、「異種」条件が満足される限り、カウンタ５２中の「異種カウント」をインクリメントする。必要な範囲で、ランカウンタ５４中のランカウントは、繰り返し現われる「異種」ワードについて入力される。これにより、この処理が高圧縮解除速度（即ち「同一」モード中には「ラン」をカウントしないもの）に「調整される」。同一モード中のランカウントを考慮することにより一層高い圧縮率を達成することができる。

【００４６】

モード状態が切り換わるたびに、「同一」カウント、「異種」カウント、及び「同一」ワードに続く「異種」ワードを含む同一／異種コマンドが発せられる。異種モードは、（１）行の中に、その上の行中の垂直方向に並んだワードと同じワードが現われるか、（２）２より大きいランレングスが現われて終了するか、（３）行が終了するまで、継続する（ステップ１５８）。この手順が依然として「異種モード」にあることに留意すると、上の行と同じワードが現われた際に、「同一」への条件切り換えが行われ、上述のように同一／異種コマンドが出力される。２より大きいランレングスが現われて終了すると、ランカウントとランされるワードとを含むランコマンドが出力される。また、同一／異種コマンドが、異種カウンタ５２からの異種カウントと、現われた異種ワードと共に出力される（ステップ１６０）。この異種カウントは、ランデータのカウントを含まない。

【００４７】

この時点で、その行が終了したか否かが判定され（ステップ１６２）、終了していなければ、同一条件への切り換えが行われ（ステップ１６４）、手順は図４のステップ１５６に戻る。これに対し、行が終了している場合には、カレント行全体が上の行と同じであるか否かが判定される（ステップ１６６）。同じであれば、その行に対して発せられた同一／異種コマンドは、その同一／異種コマンドよりはるかに簡潔な「行繰り返し」コマンドに

置き換えられる（ステップ１６８）。しかし、カレント行が上の行と同じでない場合には、最後に発せられたコマンドが０の連続を示したか否かが判定される（判定ステップ１７０）。その判定結果がＹＥＳであれば、その最後のコマンドが「ライン終了」コードに置き換えられ、圧縮解除時にはそのラインはゼロパディングされることになる（ステップ１７２）。この段階で、副画像ブロックが終了したか否かが判定され、その判定結果がＹＥＳであれば、モードｍ圧縮手順が終了する。またその判定結果がＮＯであれば、手順はステップ１５６に戻り、次の行に対して続行される。代替方法では、「行終了」境界を無視して「同一」又は「異種」カウントを「行終了」境界を超えて行うことができる。この機能により、データ圧縮は幾分改善される。

【００４８】

モードｍデータ圧縮手順、圧縮解除手順は、極めて高速であり、レーザプリンタによりオンザフライで用いることができる。このため、一般に、この手順が最初に選択される圧縮技術となる（例外もある）。モードｍの使用後にメモリ切れ又は不足の指示が依然として存在する場合には、第２の無損失の技術、例えばＬＺＷが用いられる。ＬＺＷは、無損失態様で圧縮及び圧縮解除を行うが、モードｍより低速である。しかし、ＬＺＷテーブル中における突き合わせの結果として出力されるコードの長さを制限することにより（例えば３２ビットワードにつき３コード）、その圧縮速度及び圧縮解除速度を上げることができる。しかし、ＬＺＷは、分散画像を圧縮することはできない（モードｍも同様）。分散画像の圧縮にＬＺＷが用いられた場合に発生する時間の浪費を無くすために、かかる画像は、モードｍ圧縮の結果として圧縮率が１．１２５～１より小さくなる場合を判定することによってその一部が検出され、上記値より小さくなっている場合には、その画像は分散されたものと判定される。ＬＺＷはかかる画像には用いられない。これは、その場合にデータ圧縮を行うことができないことが周知であるためである。

【００４９】

クラスタ化／分散化の判定

クラスタ化画像は、規則的に間隔のあいたドットを呈し、モードｍ、ＬＺＷその他の技術を用いて無損失に圧縮することができるものである。これに対し、分散化画像は、ドットが規則的な繰り返しパターンを有さず、従って無損失の圧縮法には不適当なものである。画像又は副画像は、その画像の２進値を（以下に示す）「隣接（ｎｅｉｇｈｂｏｒ）テーブル」中の項目と比較することにより、クラスタ化画像又は分散化画像のいずれかに分類することができる。隣接テーブルのクラスタ化部分中の項目に一致する画像の各バイト毎に値「＋１」が割り当てられる。また、隣接テーブルの「分散化」部分中の項目に一致する画像の各バイト毎に値「－１」が割り当てられる。そのテーブル中の項目に一致しない画像の他の全てのバイトは「不明」であり、値「０」が割り当てられる。画像圧縮の最後に、その割り当てられた値が合計され、負の値が示された場合には画像は分散化されているものとみなされ、また正の値が示された場合には画像はクラスタ化されているものとみなされる。かかる分類を可能にする隣接テーブルの各項目を以下に示す。

【００５０】

画像タイプを予測するための隣接テーブル

２進値	指示	値
00001110	クラスタ化	1
00011100	クラスタ化	1
00011110	クラスタ化	1
00100100	クラスタ化	1
00110011	クラスタ化	1
00111000	クラスタ化	1
00111100	クラスタ化	1
00111110	クラスタ化	1
01001001	クラスタ化	1
01100110	クラスタ化	1
01110000	クラスタ化	1
01111000	クラスタ化	1
01111100	クラスタ化	1
01111110	クラスタ化	1
10010010	クラスタ化	1
11001100	クラスタ化	1
00010100	分散化	－1
00100110	分散化	－1
00101000	分散化	－1
00110110	分散化	－1
01010101	分散化	－1
01011010	分散化	－1
01011101	分散化	－1
01100100	分散化	－1
01101100	分散化	－1
10010101	分散化	－1
10011010	分散化	－1
10100101	分散化	－1
10101001	分散化	－1
10111010	分散化	－1

ＬＺＷ圧縮

ここで図６を参照する。モードｍは、圧縮手順として既に試みられており、ＬＺＷを試みるという判断が既に下されていることが仮定されている。判定ステップ２００に示すように、第１の問い合わせは、画像のいずれかの部分に有損失圧縮技術が用いられたか否かということである。有損失圧縮技術が用いられていれば、ＬＺＷ手順は中止され、調べられている画像部分について有損失圧縮が続行される（副画像に有損失圧縮が適用されると、画像の一部のみについて無損失圧縮技術が用いられた場合に画像の美しさが失われることになる）。

【００５１】

有損失圧縮が画像のどの部分にも用いられていない場合には、次いでモードｍ圧縮の結果として画像全体に対して少なくとも１．１２５：１の圧縮率が得られるか否かが判定される（ステップ２０２）。その判定結果がＹＥＳであれば、この画像はクラスタ化されているものとみなされ、手順がステップ２０４に移行して、ＬＺＷ手順が以前に試行されていない限り、各副画像に対してＬＺＷ手順が試行される。

【００５２】

これに対し、モードｍ圧縮の結果として圧縮率が１．１２５：１より小さくなった場合には、その画像から副画像が選択され（ステップ２０６）、上述と同様の態様で隣接テーブルを用いてその副画像がクラスタ化されたものであるか分散化されたものであるかが判定される。その副画像がクラスタ化されたものであると判定されると（判定ステップ２０８）、手順はその副画像に対してＬＺＷ圧縮技術を試行する。しかし、その副画像が分散化したものであると判定されると、ＬＺＷではそれを圧縮することができないことがわかっており、その副画像は「ＬＺＷ不能」とマークされる（ステップ２１０）。

【００５３】

この時点で、その画像中の全ての副画像が検査されたか否かが判定され（判定ステップ２１２）、その判定結果がＮＯであれば、手順はステップ２０６に戻って続行される。また、全ての副画像が検査されていると判定された場合には、全ての画像が終了しているか否かが判定され、その判定結果がＮＯであれば、手順がステップ２０２に戻って続行される（判定ステップ２１４）。また、その判定結果がＹＥＳであれば、メモリ切れ又はメモリ不足の指示を調べて（判定ステップ２１６）その指示が除去されたか否かが判定される。その判定結果がＹＥＳであれば、ページ中間体の処理が続行される。また、その判定結果がＮＯであれば、手順は有損失圧縮技術の試行へと移行する。

【００５４】

有損失圧縮

有損失圧縮手順は最後の手段として用いられる。これは、この技術が回復不能なデータ損失を伴うからである。有損失圧縮技術は、主として無損失技術を用いて圧縮することのできない画像に用いることを意図したものである。一般にこの範疇に含まれる画像は、誤差拡散手順（即ち「分散化」）を用いた中間調の画像である。

【００５５】

図７に示すように、有損失圧縮技術は、まず、モードｍ又はＬＺＷを用いて以前に行われたデータ圧縮のレベルに基づいてページ上の画像をソートすることから始まる（ステップ２２０）。次いで、最も圧縮度の悪い画像が最初に調べられ（ステップ２２２）、その画像が有損失圧縮が達成可能なものより良好に圧縮されていれば、その画像はスキップされる（ステップ２２４）。一例として、４×４ビットセルのラスタビデオデータが４ビットワードに圧縮されるものと仮定する。従って、有損失圧縮技術によって４：１の圧縮が達成される。しかし、それ以前に無損失圧縮技術によって４：１より大きい圧縮率が達成されていれば、有損失圧縮技術は試行されない。各圧縮手順は、画像又は副画像の未圧縮状態のものに適用され、既に圧縮されている画像又は副画像は、まず圧縮解除しない限り更に圧縮されることは決してないことを想起されたい。

【００５６】

この段階で、カレント画像が調べられて、それがクラスタ化されたものであるか分散化されたものであるかを指示するためにタグ付けされ（ステップ２２６）、そのいずれの場合

にも有損失圧縮が行われる（ステップ２２８）。そのタグは、後で明らかになるように、圧縮解除時に用いられる。画像が圧縮されると、その画像中の全ての副画像が有損失圧縮され得るように有損失フラグがセットされてその画像に関係づけられる（ステップ２２８）。次いで、メモリ切れ又はメモリ不足の指示が除去されたか否かが判定される（判定ステップ２３２）。その判定結果がＮＯであって全ての画像が完了していれば（判定ステップ２４０）、メモリ切れの状態のままであっても印刷動作が開始され、それ以外の場合には次の画像がアクセスされ（ステップ２４１）、それに対して有損失圧縮が行われる。メモリ切れ又はメモリ不足の指示が除去されると（判定ステップ２３２）、手順はページ中間体変換を続行する。

【００５７】

ここで図８ないし図１０を参照する。同図において、図７のステップ２２８，２３０に示す手順の詳細について考察する。まず、上述のように、副画像中の各バイトが調べられて隣接テーブルのクラスタ化／分散化の値と比較される（ステップ２４２）。クラスタ化項目に一致する任意のバイトには値「＋１」が割り当てられ、分散化項目に一致する任意のバイトには値「－１」が割り当てられ、そのどちらでもない（不明の）ものには値「０」が割り当てられる（ステップ２４４）。その割り当てられた値は次いで合計されて（ステップ２４６）その副画像と関係付けて記憶される。次に、この４×４ビットブロックの副画像がアクセスされ（ステップ２４８）、その副画像中の「オン」ビット数がカウントされ、「オン」ビット数を示す４ビット値が前記ブロックに割り当てられて記憶される（ステップ２５０）。１６のカウント値は１５として記憶される。このようにして、４×４ビットセルの副画像のそれぞれが、４ビット値と、その副画像がクラスタ化されているか分散化されているかを示す更に割り当てられた値とによって表わされる。この４ビットの圧縮された値は、元のセルの元の黒の領域を表わすのに用いられる。黒のデータが必ずしも同じ位置になくても圧縮解除はその領域を維持することができる。

【００５８】

この有損失圧縮された画像が４×４のブロックに圧縮解除されるとき、まずそのブロックにクラスタ化値又は分散化値が割り当てられていたか否かが判定される。クラスタ化値が割り当てられていた場合、そのブロックの４ビット値に応じて図９の４×４のマトリクスが選択される。例えば、その４ビット値が１１個の「オン」又は黒のドットがあったことを示す場合には、図９に示すマトリクスのうちマトリクス２５２が選択される。これに対し、このブロックが分散化されたものとして示され、４ビット値がやはり１１に等しい場合には、図１０の「オン」ドットのマトリクス２５４が選択される。従って、クラスタ化又は分散化標識により、圧縮解除の実行にどのマトリクスグループを用いるかが決定され、また４ビット値により、マトリクスグループのうちのどのマトリクスを選択するかが決定される。図９及び図１０のマトリクスのパターンは、実験的に導出され、他の方法では４ビット圧縮値から直接回復することのできない損失情報のいくつかを回復することを可能にする。これらのテーブルは一般的な場合を表わすものである。このようなテーブルは装置毎に異なるものとすることができる。

【００５９】

メモリ切れ又はメモリ不足手順

以上、モードｍ、ＬＺＷ、及び有損失の圧縮技術を説明してきたが、ページ中間体の処理中にメモリ切れ又はメモリ不足の指示があった場合に本発明で行う手順について考察する。図１１及び図１２に示すように、メモリ切れ又はメモリ不足の指示があった場合（判定ステップ３００）、まず画像のいずれかの部分で以前に有損失圧縮が適用されたものがあるか否かが判定される（判定ステップ３０２）。その判定結果がＹＥＳの場合、この手順は（図７に示すように）その画像に対する有損失圧縮の実行へと分岐する。また前記判定結果がＮＯの場合には、その画像を調べてその画像に対して以前にモードｍ圧縮が行われているか否かが判定される（判定ステップ３０４）。その判定結果がＹＥＳであれば、モード圧縮を再度行う理由はなく、手順はＬＺＷ圧縮の試行（図６参照）へと分岐する。この画像に対してそれ以前にモードｍ圧縮が行われていなければ、モードｍが試行される（

ステップ３０６）。
【００６０】
この時点で、その画像の１つの副画像が調べられて、それが有損失圧縮された場合のサイズが判定される。この値は、その画像の高さ及び幅に基づき直接計算することができる（ステップ３０８）。その計算値が次いで有損失圧縮中に用いられ（図７のステップ２２４参照）、その有損失圧縮において、以前に試行された無損失圧縮により有損失圧縮を用いた場合に得られるものより良好な圧縮レベルが達成されたか否かが判定される。
【００６１】
次に、この手順は、モードｍ圧縮によって１対１より大きな副画像圧縮が行われたか否かを判定する（判定ステップ３１０）。その判定結果がＹＥＳであれば、圧縮された副画像は保持され（ステップ３１２）、ＮＯであれば圧縮されていない副画像が保持され、それは、モードｍに失敗したものとしてマークされる（ステップ３１４）。次いでこの手順は、全ての副画像が処理されたか否かを判定し（判定ステップ３１６）、その判定結果がＮＯであれば次の副画像を獲得して（ステップ３１８）判定ステップ３０４に戻って圧縮動作を繰り返す。全ての副画像が処理済みであれば、ページ上の最後の画像が処理されているか否かを判定し（判定ステップ３２０）、その判定結果がＹＥＳであれば、この手順はメモリ切れ又はメモリ不足状態が除去されたか否かの判定に移行する（判定ステップ３２２）。その判定結果がＮＯであれば、この手順は、そのページの副画像に対するＬＺＷ圧縮の試行へと分岐する。
【００６２】
更に圧縮すべき画像が残っていると判定された場合（判定ステップ３２０）、新しい画像がアクセスされ（ステップ３２２）、この手順は判定ステップ３０２に戻る。メモリ切れ又はメモリ不足状態が除去されると、この手順はページ中間体を継続する（ステップ３２６）。
【００６３】
以上の説明から、本発明を実施したプリンタには様々なデータ圧縮技術を用いることができ、これらのデータ圧縮技術は各圧縮技術の利点に基づいた順序で用いられ（即ち、最も高速なものが最初に用いられる）、用いるべきデータ圧縮技術の選択は以前に用いられたデータ圧縮技術の成功（又は成功の欠如）に基づいて行われる、ということが理解されよう。このようにして、画像データは、必要なときにのみそれに最も適した技術を用いて圧縮される（有損失技術は最後の選択肢である）。
【００６４】
適応型圧縮ページ保護
ここで図２に戻る。ページ中間体処理（ステップ７２）の最後に、ページ完了指示（判定ステップ９０）が発生する。ページ中間体ストリップの処理中、各命令が記憶される度に、この命令について実験的に導出された「複雑性定数」が累算されて、そのストリップについての複合「複雑性値」が生成される。
【００６５】
ページの複雑性は、ページ中間体のビデオラスタへの変換に要する時間の見積もったものである。見積もり時間（又はその一部）が印刷エンジンに許容される時間より長い場合に、ページ保護が採用されて、図１３ないし図１５に示す動作が開始される。ページ複雑性時間は、副画像オブジェクト（及びそれに含まれる命令）を含む各ストリップのページ中間体オブジェクトを読み出し、ラスタビデオへの変換過程で各オブジェクトの要する時間に基づいて見積もり時間を累算することにより、見積もられる。この見積もりは、画像オブジェクトの高さ及び幅と、画像処理がオブジェクトの各命令を実行するのに要する時間を示す定数とに基づいて行われる。
【００６６】
ページ複雑性時間（即ちストリップのラスタビデオへの変換に要する時間）の見積もり処理中であれば如何なるときでも、累算見積もりが印刷エンジンの許容範囲より最低１エンジンストリップ時間だけ大きくなると、圧縮ページ保護がはたらく。ページ複雑性時間は

また、プリンタによるストリップのラスタへの変換を可能とすること及びこの処理を刻時する（刻時された事前ラスタ化）ことによっても、計算することができる。

【００６７】

ページ複雑性時間は、ストリップ作成をシステムタイマで刻時することによって実験的に測定することができる。この技術により、ページ処理速度を落とすことなくストリップの複雑性の計算に伴う不確実性を低減させることができる。計算されたストリップの複雑性が、良好な（ＯＫ）しきい値を下回っている場合には、それは印刷オーバーランを引き起こすことは全くなく、事前ラスタ化されることはない。同様に、計算されたストリップの複雑性が、強制（ｗｉｌｌ）印刷オーバーランしきい値を上回っている場合には、その有害ストリップは確実に印刷オーバーランを引き起こすものであり、事前ラスタ化されなければならない。また、ストリップの複雑性が、良好しきい値と強制印刷オーバーランしきい値との間にある場合には、実験的な測定によって印刷オーバーランの不確実性は大幅に軽減される。実験的測定によってストリップが非印刷オーバーランとして分類され、中間体に要するメモリがストリップ（又はストリップの圧縮されたもの）より小さい場合、事前ラスタ化されたデータは破棄され、そのストリップは他の非印刷オーバーランストリップと同様にして処理される。それ以外の場合には、事前ラスタ化されたデータが用いられ、処理は他の印刷オーバーランストリップと同じ経路に沿って続行される。不確実なストリップの実験的測定を用いないページ圧縮の実際の適用においては、問題になるストリップは安全のためにおそらく事前ラスタ化される。実験的測定を用いれば、顕著な時間又は処理的な不利益を伴うことなくメモリの使用を低減させることができる。

【００６８】

図１３において、ブロック７２，９０は図２と同様である。ページ中間体が完了したと示されると、ページストリップの複雑性の複合値が前述のように計算される（ステップ３５０）。その複雑性値が、ページストリップを直ちに印刷することができることを示している場合に、印刷が開始される（判定ステップ３５２）。これに対し、ページが複雑すぎる場合には、ページ中間体の第１のストリップがアクセスされ（ステップ３５４）、ラスタビデオに変換される（ステップ３５６）。

【００６９】

ストリップがラスタビデオに変換されると、残っている各ページ中間体ストリップが調べられて、それが未圧縮状態でユーザＲＡＭに入るか否かが判定される（判定ステップ３５８）。その判定結果がＹＥＳであれば、そのストリップが未圧縮のものとしてフラグがセットされる（ステップ３６０）。また、前記判定結果がＮＯであり、そのページに対して以前にＬＺＷ圧縮が用いられている場合には（判定ステップ３６２）、ラスタビデオストリップに対してＬＺＷ圧縮が試行される（ステップ３６２）。これに対し、そのページに対してＬＺＷ圧縮が用いられていない場合には、モードｍ圧縮が試行される（ステップ３６４）。次いで、そのストリップがいずれかの圧縮動作により実際に圧縮されたか否かが判定され（判定ステップ３６６）、その判定結果がＮＯである場合、（考えられる既拡張ストリップの使用を防止するために）未圧縮ストリップが用いられる（ステップ３６８）。

【００７０】

次いで、そのストリップに対するスペースの割り当てが試行される（ステップ３６９）。図１５に示すように、連続したスペースを利用可能であることが発見された場合（判定ステップ３７０）、そのストリップはその発見されたスペースに記憶される（ステップ３７２）。また、連続したスペースが発見されない場合には、そのストリップをリンクされたメモリフラグメントとして記憶するメモリ空間があるか否かが判定される（判定ステップ３７４）。その判定結果がＹＥＳであれば、そのストリップはリンクされたフラグメントとして記憶され（ステップ３７６）、次いで手順がページ中間体の次のストリップにアクセスし（ステップ３７８）、更に処理すべきストリップがあるか否かが判定される（判定ステップ３８０）。ストリップがそれ以上存在しない場合には、印刷が開始される。また、更にストリップが存在する場合には、次のストリップがページ中間体からラスタビデオ

に変換されて（ステップ３５６）、処理が続行される。

【００７１】

また、判定ステップ３７４に戻り、ストリップをリンクされたフラグメントとして記憶するメモリ空間がなく、このストリップに対して未圧縮のフラグがセットされている場合（判定ステップ３８２）には、メモリ切れ指示が発生して印刷が開始される。また、そのストリップに対して未圧縮のフラグがセットされていない場合には、モードｍ圧縮又はＬＺＷ圧縮が、以前に試行されたのがどちらであるかに応じて（判定ステップ３８４，３８６）試行される。モードｍ圧縮及びＬＺＷ圧縮の両方が既に試行されている場合に印刷が開始される。

【００７２】

つまり、ページ中間体のストリップ（又は複雑なストリップの選択された部分集合）がページの下に向かって順次ラスタビデオに変換されて、一時バッファに入れられる。ＬＺＷ圧縮がページ上のどこかで既に用いられている場合にはＬＺＷ圧縮が試行され、そうでなければモードｍ圧縮が試行される。これは、ＬＺＷ圧縮が以前にそのページ上で、そのページがページ保護が必要とするほど使用された（非常に複雑である）場合には、モードｍは、恐らくはそのページをより良好に圧縮することができないからである。

【００７３】

圧縮後、そのラスタストリップのための記憶領域をメインメモリ中で探す。メインメモリ中に記憶領域がなかった場合には、かかる記憶領域をメインメモリの断片又はかかる状況のために別途設けられる緊急メモリプールで探す。この手順が終了すると、ページ印刷が開始される。「ストリップの大きさの」圧縮されたオブジェクトの複雑性は印刷エンジンの許容し得る最大限の複雑性よりはるかに低いので、ページはオーバーランを起こすことなく印刷される。

【００７４】

以上の説明では、連続的に記憶された記述子を有する画像に適用されたデータ圧縮について考察してきた。画像記述子は、ビットマップ副画像とは別に記憶することもできる。記述子があれば、本発明の２つの実施方法を区別する任意の手段となる。ここで図１６に画像記述子の一例を示す。この記述子は、画像の高さ、幅、画像識別子（例えば連続番号）、及び「クラス」の表記の定義を含む。更に、画像記述子は、メモリ中の他の場所にある関連する画像ビットマップのアドレスを示す。

【００７５】

画像記述子中の「クラス」表記は、データ圧縮及び圧縮解除中に用いられる画像に関する事実を示す項目である。例えば、項目「２」は、画像が「未加工」であり、データ圧縮操作を受けていないことを示す。また、項目「３」は、画像がモードｍデータ圧縮されていることを示す。更に、項目「８」又は「９」は、状況に応じて画像がクラスタ画像であるか分散化画像であるかを示す。他のクラス表記は、画像に用いられたデータ圧縮手順、即ちＬＺＷ、有損失その他を示す。不連続の画像記述子が用いられる場合、ラスタ図形のラスタビデオビットマップは記述子とは全く別個のものであり、圧縮が実行された際に記述子のビットマップに対するポインタだけが変更される。ページ中間体（表示リスト）を変更する必要は全くなく、圧縮システムはページ中間体の知識を全く有さない。

【００７６】

サンプリングされた副画像圧縮

この圧縮技術は、メモリ割り当て要求に失敗したことによりメモリ不足状態がトリガされた際に開始される。この技術には、画像中の副画像のサンプリングと、そのサンプリングされた副画像履歴から得られる期待される全体的な画像圧縮の予測と、画像中の全ての副画像への適切な圧縮技術の適用とが含まれる。この手順において、圧縮を含む所定組のメモリ割り当て操作が優先順位に従って順序づけられ、既に試行された無損失方法が必要メモリスペースを提供しない場合に、有損失方法が用いられる。最後の手段として、無損失圧縮から有損失圧縮への変換を、４倍の標準的な有損失の場合の圧縮率より低く圧縮されたあらゆる無損失画像について実行することができる。

【００７７】
図１７に示すように、この手順は「選択された全ての副画像を圧縮」と題されている。まず、ある画像中の全ての副画像が、場合に応じて未圧縮リスト又は既圧縮リストのいずれかに割り当てられる。従って、この手順の始めには、全ての副画像が未圧縮リストに列挙される。これは、まだそれらが圧縮を受けていないからである。この処理は、未圧縮リストの次の細副画像にアクセスすることにより開始される（ステップ４４２）。次いで、そのリスト中の全ての副画像が完了したか否かが判定され（判定ステップ４４４）、その判定結果がＹＥＳであれば、手順は終了する（ステップ４４５）。また、前記判定結果がＮＯであれば、その副画像の記述子がチェックされて、その副画像が属する画像が「未加工」状態、即ち未圧縮であるか否かが判定される（判定ステップ４４６）。その画像が未加工でないと判定された場合、副画像は図１７の左下に示す手順が実行される。

【００７８】
また、その画像が「未加工」であると判定された場合には、図１７の右側に示す分類手順に進む。この分類手順は、その画像に属する全ての副画像に対して用いるべき圧縮方法の割り当てを含むものである。この手順は、ステップ４６０に示すように、画像中の全ての副画像をカウントし、及び副画像の適当な大きさのサンプルを選択することにより、開始される。サンプリングされる部分集合の大きさは、副画像の利用可能数によって変動する。副画像の数が１６より少なければ、全ての副画像が用いられる。それより大きな数の副画像が存在する場合には、その総数のうちの数分の一が用いられ、その分割数は副画像の数が増えるにつれて大きくなる。好適には、部分集合は、６つの副画像毎に１つという比率より決して小さくはならない。明らかに、上記の数及び比率は好適例として挙げたものであり、その数及び比率の値は具体的な状況に応じて変更し得るものであることが、当業者には理解されよう。

【００７９】
次いで、選択された副画像のそれぞれが「次の」優先順位の無損失方法で圧縮される。最初にモードＭが用いられる。これは、その方法が圧縮及び圧縮解除において最も高速であるからである。サンプル中の全ての副画像が、選択された無損失方法を用いて圧縮されると、平均圧縮率が計算されて（ステップ４６４）所定の圧縮しきい値と比較される。その圧縮方法による圧縮結果の圧縮率が前記しきい値を超えることがわかった場合、判定ステップ４６６，４７２に示すようにその方法が直ちに選択される。方法が選択されると、それが画像の全副画像に割り当てられ（ステップ４７４）、手順は図１７の左側に移行する。

【００８０】
しきい値を超えず、未試行の他の無損失圧縮技術が存在する場合には（判定ステップ４６８）、手順はステップ４６２に戻り、次に低い優先順位の無損失方法を実行する。判定ステップ４６８で、利用可能な無損失方法がそれ以上存在しないことが示された場合には、有損失技術が用いられ、画像は分散化画像又はクラスタ化画像のいずれかとして判定される（ステップ４７０）。この場合、有損失方法は画像の全ての副画像に対して割り当てられ、手順は判定ステップ４４８に移行する。

【００８１】
分類手順中に、副画像の選択されたサンプルに対して圧縮が発生するが、かかる圧縮の結果は保持されない（圧縮率の計算及び前記しきい値との比較を目的とする場合を除く）。画像及び副画像の実際の圧縮は、図１７の左側に示すように発生する。

【００８２】
あるページ上の画像の圧縮中には、共通の圧縮法を割り当てられた全ての画像が順次処理され、別の圧縮法を割り当てられた画像はその次に順次処理される。従って、全ての場合において、圧縮ルーチン「選択された全副画像の圧縮」で現在用いられている方法と一致する圧縮方法が割当てられた画像に現在の副画像が属するか否かが判定ステップ４４８でチェックされる。その判定結果がＮＯの場合、ステップ４４２で未圧縮リスト中の次の副画像がアクセスされて手順が繰り返される。また、前記判定結果がＹＥＳであり、適当な

圧縮ルーチンが実行されている場合には、個々の副画像に対して直ちに圧縮が実行され、その結果がステップ４５０に示すように予備メモリに記憶される。副画像の圧縮が終了すると、古いビットマップメモリは割り当てが解除される。次いで、既圧縮画像の記述子が既圧縮リストに移動され（ステップ４５４）、手順は次の副画像のアクセスに移行する。

【００８３】

フォント圧縮

高速レーザプリンタでは、フォントは一般には圧縮ルーチンによる処理を受けることはない。しかし、プリンタ制御言語によっては、フォントビットマップがプリンタのメモリ内で連続していることを必要とする（例えばＰＣＬ）。これは、小さいフォントの場合には問題にならない。なぜなら、それに伴うビットマップが小さく、大きなメモリ領域を占有することがないからである。しかし、大きなフォントはメモリ切れの問題を生じさせるものとなる。これは、かかるフォントの記憶に要するブロックサイズがフォントの「ポイント」サイズの２乗で大きくなるためである。そのような大きなフォントを受け取った場合に、プリンタ内で多数の連続ブロックを利用することができないことがある。更に、非常に大きな連続するメモリブロックを確保するということは浪費的であり、できれば避けるべきである。また、（図１８に示すように）ページ４３０上に印刷された大きなフォントサイズの場合を考えると、所望のフォントでページ全体が印刷されるより充分前にメモリ切れ状態が発生する。これを防止するために、受信フォントに対してデータ圧縮が加えられる。

【００８４】

ここで図１９を参照する。フォント圧縮手順が、まず文字フォントサイズを識別する（ステップ４３２）。フォントサイズが所定のしきい値より大きい場合（判定ステップ４３３）、このフォントはモードｍを用いて直ちに圧縮される。これに対し、文字フォントサイズがしきい値より小さい場合、直ちに圧縮を行うことはなく、メモリ切れ又はメモリ不足指示が出るまで圧縮動作は保留される（ステップ４３６）。

【００８５】

以上の説明は本発明の例示にすぎないことが理解されよう。当業者であれば、本発明から逸脱することなく様々な代替態様や修正態様を実施することが可能である。従って、本発明は、特許請求の範囲内に含まれる代替、修正、変形態様の全てを含むことが意図されている。

【００８６】

【発明の効果】

本発明は上述のように構成したので、データ圧縮用のシステムを含み、受信データの特性に応じて適応可能なページプリンタを提供することが可能となり、また、限られた量のオンボードメモリを効率的に用いることを可能にするようにデータ圧縮技術を知的に選択するシステムを採用したページプリンタを提供することが可能となる。

【図面の簡単な説明】

【図１】本発明の実施に適合された周辺装置を示す高水準ブロック図である。

【図２】本発明の手順全体を示す概略的なフローチャートである。

【図３】ストリップビデオの事前生成の為の手順を示す高水準フローチャートである。

【図４】ここで用いるモードｍと呼ばれる高速データ圧縮技術を示す高水準フローチャートである（１／２）。

【図５】ここに用いるモードｍと呼ばれる高速データ圧縮技術を示す高水準フローチャートである（２／２）。

【図６】Ｌｅｍｐｅｌ／Ｚｉｖ／Ｗｅｌｃｈデータ圧縮手順を示す高水準フローチャートである。

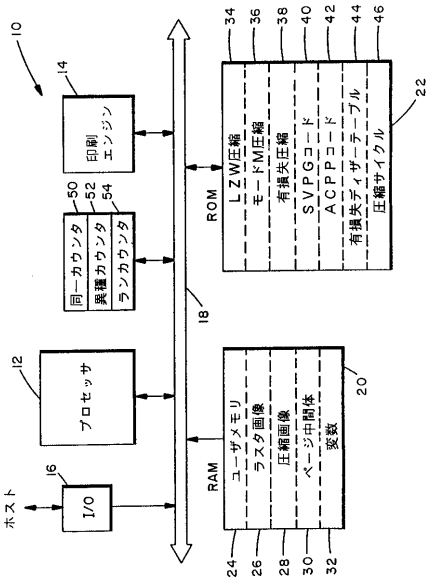
【図７】Ｌｏｓｓｙ圧縮手順を示す高水準フローチャートである。

【図８】画像がクラスタ化されているか分散しているかを判定するのに用いられる手順を示す高水準フローチャートである。

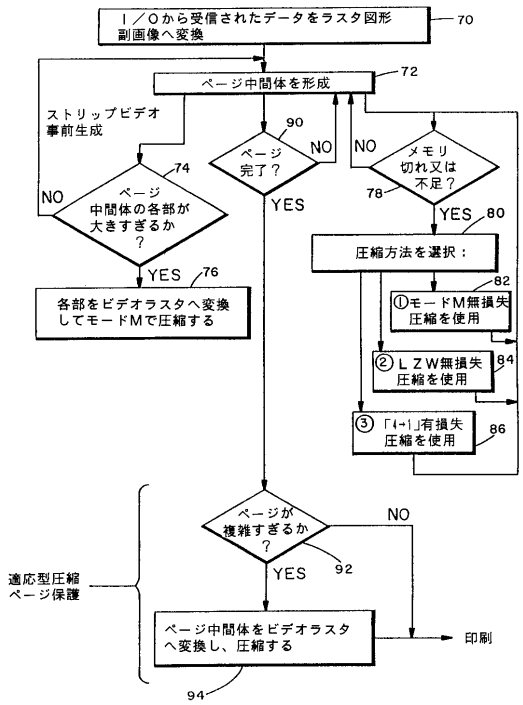
【図９】図８の手順の結果に従って用いられるクラスタ化有損失圧縮解除マトリクスを示

す説明図である。
【図１０】図８の手順の結果に従って用いられる分散化有損失圧縮解除マトリクスを示す説明図である。
【図１１】図１のシステムにおいてメモリ残量のない状態又は不足した状態において用いるべき圧縮手順を判定する際に用いられる手順を示す高水準フローチャートである（１／２）。
【図１２】図１のシステムにおいてメモリ残量のない状態又は不足した状態において用いるべき圧縮手順を判定する際に用いられる手順を示す高水準フローチャートである（２／２）。
【図１３】「ページ中間体」動作が完了し、ページが複雑すぎることが発見された場合に実行される手順を示す高水準フローチャートである（１／３）。
【図１４】「ページ中間体」動作が完了し、ページが複雑すぎることが発見された場合に実行される手順を示す高水準フローチャートである（２／３）。
【図１５】「ページ中間体」動作が完了し、ページが複雑すぎることが発見された場合に実行される手順を示す高水準フローチャートである（３／３）。
【図１６】本発明の第２の態様におけるビットマップ副画像を伴う記述子を示す説明図である。
【図１７】採用すべき圧縮手順を判定するために本発明の第２の態様において実行される手順を示すフローチャートである。
【図１８】大きなポイントサイズのフォントの印刷に応じて「メモリ切れ」指示が発生した場合のページを示す説明図である。
【図１９】受信フォントに圧縮が加えられる場合を示すフローチャートである。
【符号の説明】
１０　　プリンタ
２０　　ランダムアクセスメモリ
２２　　リードオンリーメモリ

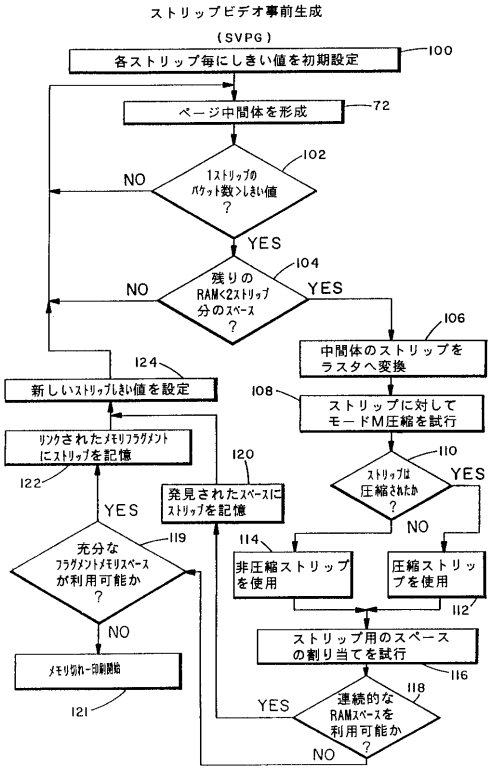
【図１】


10
14
印刷エンジン
50
52
54
同一カウンタ
異種カウンタ
ランカウンタ
12
プロセッサ
16
I/O
ホスト
18
ROM
34
36
38
40
42
44
46
LZW圧縮
モードM圧縮
有損失圧縮
SVPGコード
ACPPコード
有損失ディザーテーブル
圧縮サイクル
22
RAM
ユーザメモリ
ラスタ画像
圧縮画像
ページ中間体
変数
20
24
26
28
30
32

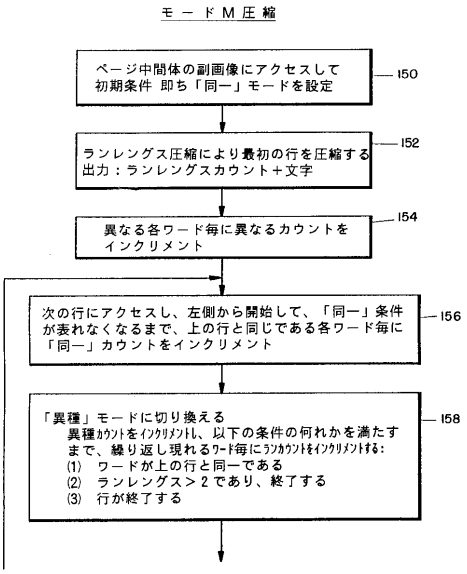
【図２】


I／Oから受信されたデータをラスタ図形副画像へ変換
70
ページ中間体を形成
72
ストリップビデオ事前生成
90
ページ完了？
NO
YES
74
ページ中間体の各部が大きすぎるか？
NO
YES
76
各部をビデオラスタへ変換してモードMで圧縮する
78
メモリ切れ又は不足？
NO
YES
80
圧縮方法を選択：
82
①モードM無損失圧縮を使用
84
②LZW無損失圧縮を使用
③「1→1」有損失圧縮を使用
86
適応型圧縮ページ保護
ページが複雑すぎるか？
NO
YES
92
ページ中間体をビデオラスタへ変換し、圧縮する
印刷
94

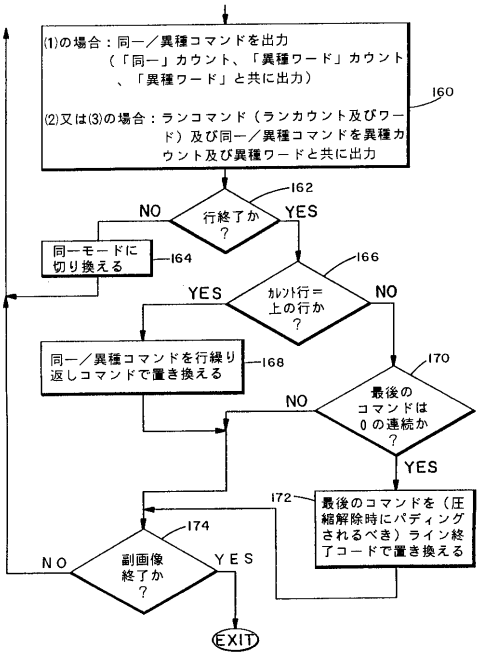
【図３】


ストリップビデオ事前生成
(SVPG)
各ストリップ毎にしきい値を初期設定
100
ページ中間体を形成
72
102
1ストリップのバケット数>しきい値？
NO
YES
104
残りのRAM<2ストリップ分のスペース？
NO
YES
106
中間体のストリップをラスタへ変換
108
ストリップに対してモードM圧縮を試行
110
ストリップは圧縮されたか？
YES
NO
112
ストリップ用のスペースの割り当てを試行
114
非圧縮ストリップを使用
圧縮ストリップを使用
116
118
連続的なRAMスペースを利用可能か？
YES
NO
120
発見されたスペースにストリップを記憶
119
充分なフラグメントメモリスペースが利用可能か？
YES
NO
121
メモリ切れ→印刷開始
122
リンクされたメモリフラグメントにストリップを記憶
124
新しいストリップしきい値を設定

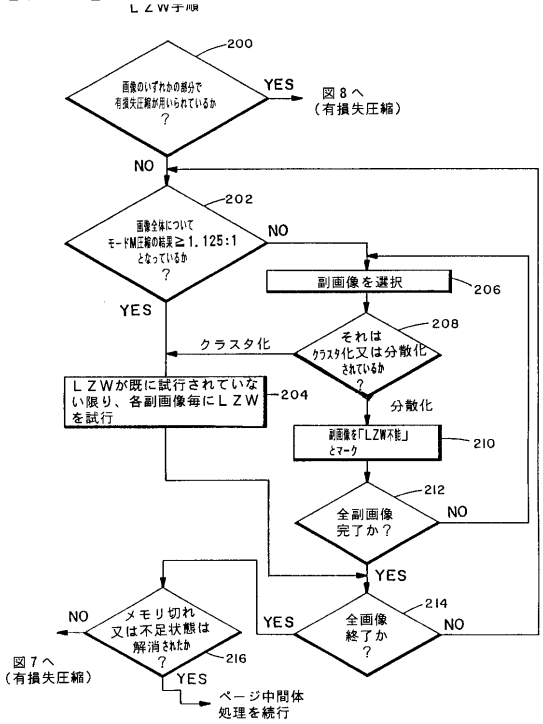
【図４】


モードM圧縮
ページ中間体の副画像にアクセスして初期条件 即ち「同一」モードを設定
150
ランレングス圧縮により最初の行を圧縮する
出力：ランレングスカウント＋文字
152
異なる各ワード毎に異なるカウントをインクリメント
154
次の行にアクセスし、左側から開始して、「同一」条件が表れなくなるまで、上の行と同じである各ワード毎に「同一」カウントをインクリメント
156
「異種」モードに切り換える
異種カウントをインクリメントし、以下の条件の何れかを満たすまで、繰り返し現れるワード毎にランカウントをインクリメントする：
(1) ワードが上の行と同一である
(2) ランレングス＞2であり、終了する
(3) 行が終了する
158

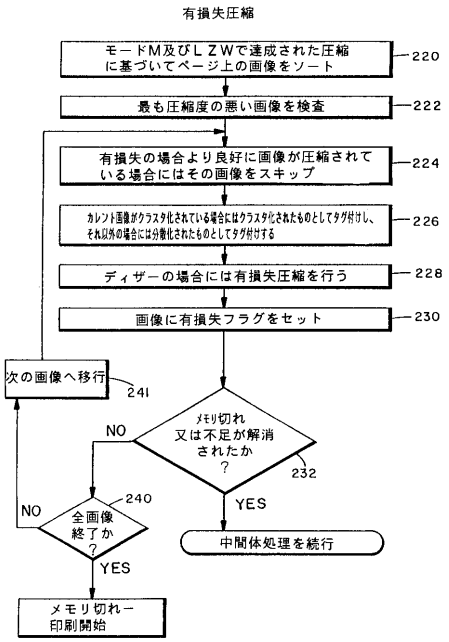
【図５】


(1)の場合：同一／異種コマンドを出力
（「同一」カウント、「異種ワード」カウント、「異種ワード」と共に出力）
(2)又は(3)の場合：ランコマンド（ランカウント及びワード）及び同一／異種コマンドを異種カウント及び異種ワードと共に出力
160
行終了か？
162
NO
YES
同一モードに切り換える
164
カレント行＝上の行か？
166
YES
NO
同一／異種コマンドを行繰り返しコマンドで置き換える
168
最後のコマンドは0の連続か？
170
NO
YES
最後のコマンドを（圧縮解除時にパディングされるべき）ライン終了コードで置き換える
172
副画像終了か？
174
NO
YES
EXIT

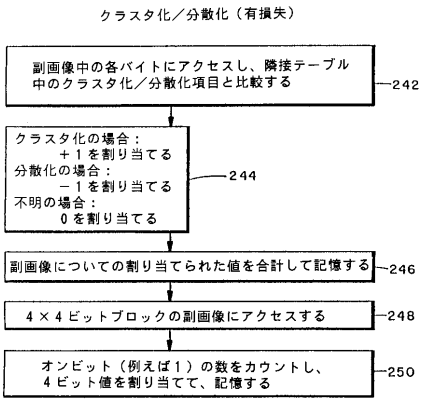
【図６】


ＬＺＷ手順
画像のいずれかの部分で有損失圧縮が用いられているか？
200
YES
図８へ（有損失圧縮）
NO
画像全体についてモードM圧縮の結果≧1.125:1となっているか？
202
NO
YES
副画像を選択
206
クラスタ化
それはクラスタ化又は分散化されているか？
208
分散化
ＬＺＷが既に試行されていない限り、各副画像毎にＬＺＷを試行
204
副画像を「LZW不能」とマーク
210
全副画像完了か？
212
NO
YES
全画像終了か？
214
NO
YES
メモリ切れ又は不足状態は解消されたか？
216
NO
YES
図７へ（有損失圧縮）
ページ中間体処理を続行

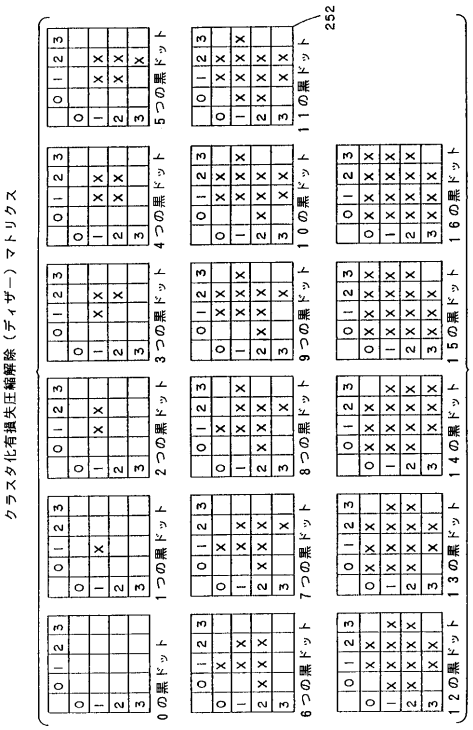
【図７】


有損失圧縮
モードM及びＬＺＷで達成された圧縮に基づいてページ上の画像をソート
220
最も圧縮度の悪い画像を検査
222
有損失の場合より良好に画像が圧縮されている場合にはその画像をスキップ
224
カレント画像がクラスタ化されている場合にはクラスタ化されたものとしてタグ付け し、それ以外の場合には分散化されたものとしてタグ付けする
226
ディザーの場合には有損失圧縮を行う
228
画像に有損失フラグをセット
230
次の画像へ移行
241
メモリ切れ又は不足が解消されたか？
232
NO
YES
全画像終了か？
240
NO
YES
中間体処理を続行
メモリ切れー印刷開始

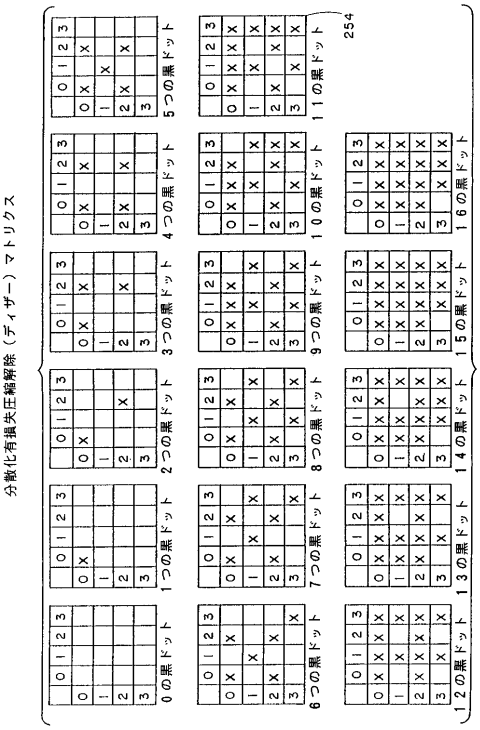
【図８】


クラスタ化／分散化（有損失）
副画像中の各バイトにアクセスし、隣接テーブル中のクラスタ化／分散化項目と比較する
242
クラスタ化の場合：＋１を割り当てる
分散化の場合：－１を割り当てる
不明の場合：0を割り当てる
244
副画像についての割り当てられた値を合計して記憶する
246
４×４ビットブロックの副画像にアクセスする
248
オンビット（例えば１）の数をカウントし、４ビット値を割り当てて、記憶する
250

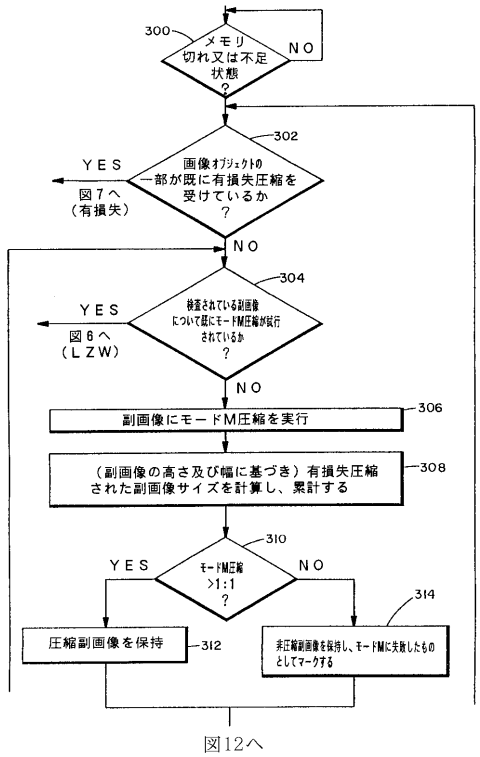
【図９】



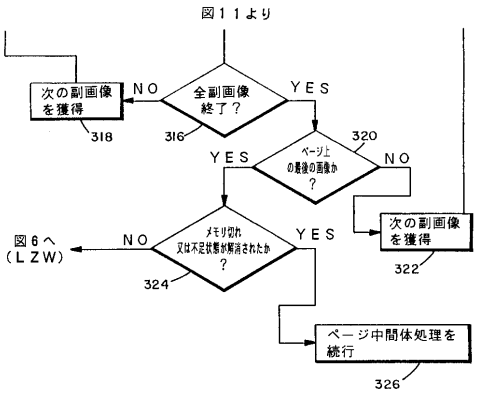
【図１０】



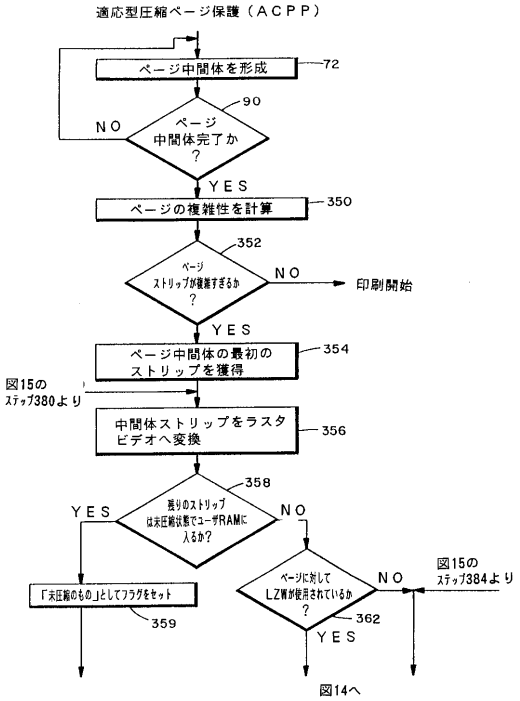
【図１１】



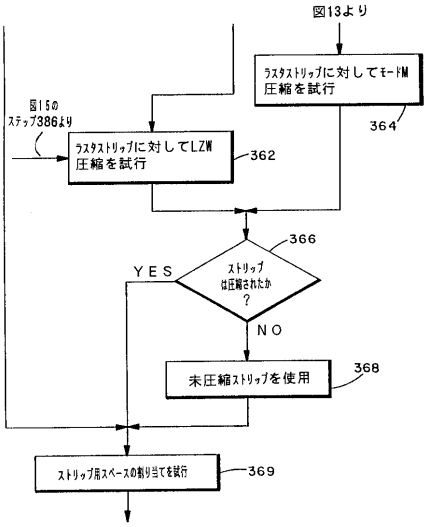
【図１２】



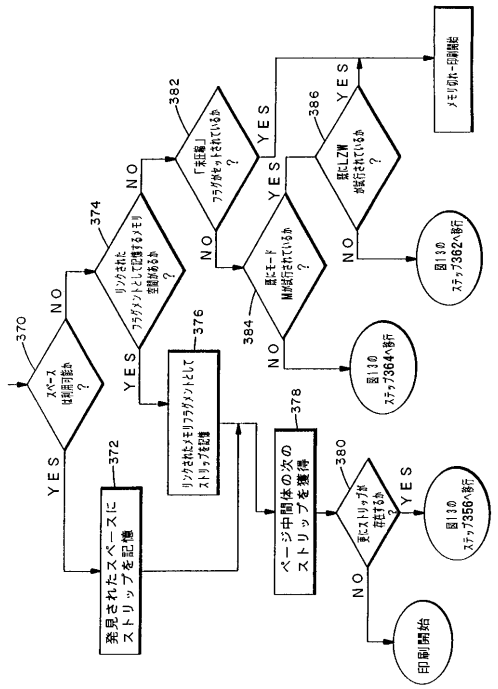
【図１３】



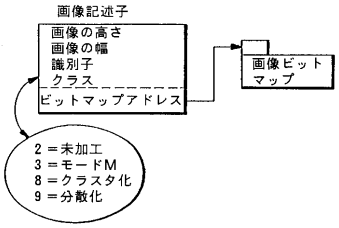
【図１４】



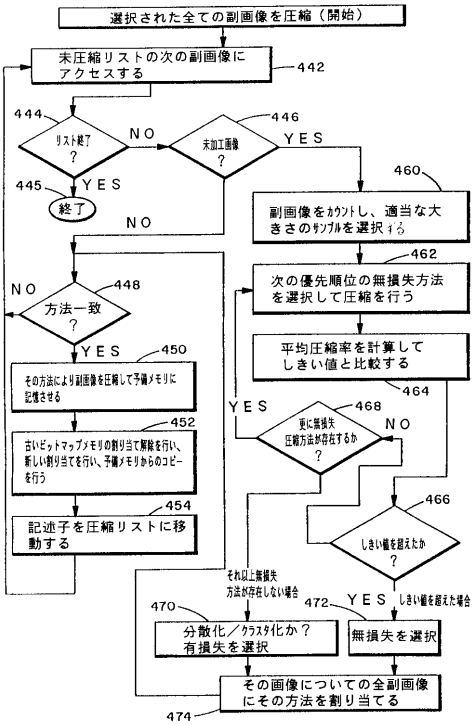
【図１５】



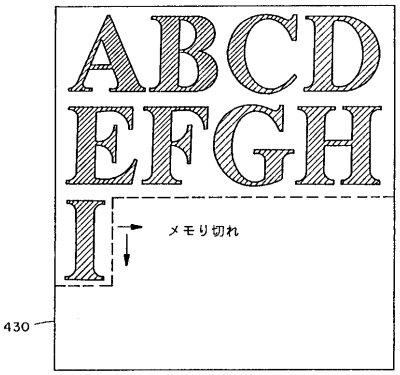
【図１６】



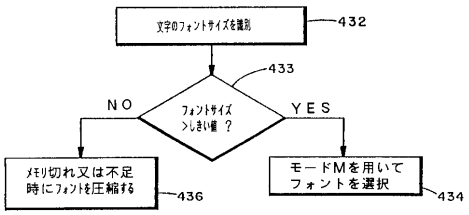
【図１７】


選択された全ての副画像を圧縮（開始）
未圧縮リストの次の副画像にアクセスする
442
444
リスト終了？
NO
446
未加工画像？
YES
445
終了
YES
460
副画像をカウントし、適当な大きさのサンプルを選択する
NO
462
次の優先順位の無損失方法を選択して圧縮を行う
NO
448
方法一致？
YES
450
その方法により副画像を圧縮して予備メモリに記憶させる
452
古いビットマップメモリの割り当て解除を行い、新しい割り当てを行い、予備メモリからのコピーを行う
454
記述子を圧縮リストに移動する
平均圧縮率を計算してしきい値と比較する
464
YES
468
更に無損失圧縮方法が存在するか？
NO
466
しきい値を超えたか？
それ以上無損失方法が存在しない場合
470
分散化／クラスタ化か？有損失を選択
472
YES
しきい値を超えた場合
無損失を選択
その画像についての全副画像にその方法を割り当てる
474

【図１８】


ABCD
EFGH
I
メモリ切れ
430

【図１９】


文字のフォントサイズを識別
432
433
フォントサイズ＞しきい値？
NO
YES
メモリ切れ又は不足時にフォントを圧縮する
436
モードMを用いてフォントを選択
434

フロントページの続き

(72)発明者 ゲーリー・ツィンマーマン
アメリカ合衆国アイダホ州８３７０４ボイス，ノース・ファロウ・４８４９
(72)発明者 トーマス・ジー・バージ
アメリカ合衆国アイダホ州８３７１２ボイス，イースト・ジェーファーソン・ストリート・７０２
(72)発明者 テリー・エム・ネルソン
アメリカ合衆国アイダホ州８３７０９ボイス，モホーク・ドライヴ・１０９３０

合議体
審判長 小川 謙
審判官 加藤 恵一
審判官 井上 信一

(56)参考文献 特開平４－８７４７１（ＪＰ，Ａ）
特開昭６０－７４７２９（ＪＰ，Ａ）
特開平４－２９１８２６（ＪＰ，Ａ）