

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4610450号
(P4610450)

(45) 発行日 平成23年1月12日(2011.1.12)

(24) 登録日 平成22年10月22日(2010.10.22)

(51) Int.Cl.		F I		
HO4N	1/41	(2006.01)	HO4N	1/41 B
GO6T	3/60	(2006.01)	GO6T	3/60
HO4N	1/387	(2006.01)	HO4N	1/387

請求項の数 5 (全 19 頁)

(21) 出願番号	特願2005-259577 (P2005-259577)	(73) 特許権者	000006747
(22) 出願日	平成17年9月7日(2005.9.7)		株式会社リコー
(65) 公開番号	特開2007-74412 (P2007-74412A)		東京都大田区中馬込1丁目3番6号
(43) 公開日	平成19年3月22日(2007.3.22)	(74) 代理人	100110319
審査請求日	平成20年7月11日(2008.7.11)		弁理士 根本 恵司
		(72) 発明者	鈴木 正宏
			東京都大田区中馬込1丁目3番6号株式会 社リコー内
		審査官	富永 達朗

最終頁に続く

(54) 【発明の名称】 固定長圧縮画像と属性情報のパッキングデータを処理する画像処理装置

(57) 【特許請求の範囲】

【請求項1】

画像データを記憶する記憶部と、

前記画像データを正方形領域単位でブロック化して固定長圧縮する手段と、前記画像データの属性情報を同じブロック単位でブロック化する手段と、前記正方形領域単位のブロックで固定長圧縮された前記画像データ及び属性情報を1ブロックごとにパッキングして前記記憶部に格納する手段とを備えた2次元ブロックデータフォーマット変換部と、

前記画像データをライン方向の複数画素単位でブロック化して固定長圧縮する手段と、前記画像データの属性情報を同じライン方向のブロック単位でブロック化する手段と、前記ライン方向の複数画素単位のブロックで固定長圧縮された前記画像データ及び属性情報を1ブロックごとにパッキングして前記記憶部に格納する手段とを備えた1次元ブロックデータフォーマット変換部と、

前記2次元ブロックデータフォーマット変換部により前記記憶部に格納された主走査方向及び副走査方向の複数の正方形領域単位ブロックにより構成されるブロック群における各正方形領域単位ブロックの向きを固定したまま前記ブロック群を90°の整数倍回転させる画像回転部と、

前記1次元ブロックデータフォーマット変換部により前記記憶部に格納されたパッキングデータをライン方向のブロック単位で復号してラスタデータに変換する伸張部とを備えたことを特徴とする画像処理装置。

【請求項2】

固定長圧縮画像と属性情報が正方形領域単位ブロックごとにパッキングされているデータについて正方形領域単位ブロック単位に90°回転したアドレスを発生する手段と、回転したアドレスに従って一度のメモリリードアクセスで正方形領域単位ブロックを前記記憶部から読み出す手段と、読み出した1ブロックのデータを回転する画像回転器と、回転した1ブロックのデータをラスタ順に前記記憶部に書き込む手段とを備えたことを特徴とする請求項1記載の画像処理装置。

【請求項3】

固定長圧縮画像と属性情報が正方形領域単位ブロックごとにパッキングされているデータについて正方形領域単位ブロック単位にデータをラスタ順に前記記憶部から読み出す手段と、読み出した1ブロックのデータを回転する画像回転器と、ブロック単位に90°回転したアドレスを発生する手段と、回転した正方形領域単位ブロックを、回転したアドレスに従って一度のメモリライトアクセスで前記記憶部に書き込む手段とを備えたことを特徴とする請求項1記載の画像処理装置。

10

【請求項4】

前記画像データは、非圧縮画像データである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5】

前記画像データは、可変長圧縮画像データまたは固定長圧縮画像データであることを特徴とする請求項1に記載の画像処理装置。

【発明の詳細な説明】

20

【技術分野】

【0001】

本発明は、画像処理装置に関し、特に、ブロック単位で固定長圧縮した画像情報と属性情報をパッキングして、画像データを高速に処理する画像処理装置に関する。

【背景技術】

【0002】

画像のデータフォーマットには、JPEG等の標準の可変長圧縮フォーマットが存在する。それらは、固定長圧縮方式より、伸張後の画像劣化が少ないことが多い。したがって、配信に用いられるか紙出力されるか分からない画像を、とりあえず圧縮して保存するには、JPEGの様な可変長圧縮フォーマットが適している。この際、属性情報は、画像とは別に保存される。属性情報を可逆圧縮して保存するか、そのまま保存するかは任意である。

30

【0003】

画像を回転するためには、例えば、図17(a)に示すように、主走査方向8line×副走査方向8画素の画像を、あらかじめパッキング(1回のメモリアクセスで読み出せるように1かたまりにまとめること)しておくと、メモリアクセス回数を減らし、高速な回転処理を行うことができる。1画素が1byte(8bit)で表される場合、8画素だと8byteである。通常、画像を回転処理する場合、8×8画素ずつ回転させるとすると、メモリアクセスは、8byte×8回必要となる。これに対し、あらかじめ8×8画素ずつパッキングされていれば、メモリアクセスは、64byte×1回で済む。メモリは、連続したアドレスであれば、バーストアクセスができるので、図17(b)に示すような8回の2wordバーストアクセスに対し、図17(c)に示すように、1回の16wordバーストアクセスで済む後者のアクセス方法では、転送準備と終了処理が1回ですむので、メモリアクセス時間が短縮される。

40

【0004】

バースト転送によりメモリアクセス数を減少させると、メモリアクセス時間が短縮されることの他に、システムとしてみて、他のDMACによるメモリアクセスを邪魔しないで済むというメリットが得られる。1画素の情報量が少なければ少ないほど、このメリットは大きくなる。MFPのように、1画素1bit(CMYK画像)の情報量の場合、32×32画素ずつ回転させるとすると、メモリアクセスは1word(4byte)×32回必要となるが、あらかじめ32×32画素ずつパッキングしてあれば、32word(128byte)×1回のアクセスで済む。

50

【 0 0 0 5 】

画像を回転することを考慮すると、圧縮方法としては、固定長符号が有利である。固定長符号は、数画素ごとに固定されたビット長に圧縮される。圧縮された後の符号が、ある単位（ブロックと呼ぶ）ごとに決まったビット長になることから、ブロックごとの先頭アドレスを特定できる。したがって、隣のブロック同士に相関の無い固定長符号方式の場合は、必要なブロックのみを取り出すことができる。隣のブロック同士に相関の無い固定長符号の場合は、メモリ上には圧縮した画像を置いたまま、回転に必要なブロックを取得して回転し、再圧縮してメモリに戻すことが可能であり、メモリの使用効率を高めることが可能である。

【 0 0 0 6 】

画像の固定長圧縮には、1 lineの数画素を1ブロックとして圧縮する方法と、例えば、図18に示すように、8 × 8画素を1ブロックとするような、2次元の画像を1ブロックとして圧縮する方法がある。画像を回転するためには、メモリアクセス数が減る分、後者の方が有利である。例えば、特許文献3には、固定長の符号化を行い、かつ90°単位の回転処理が可能な方法が開示されている。可変長符号方式でも、例えば、特許文献1などに開示されている方法のように、ブロックとアドレスの関連を記憶することで、回転を行うことはできる。

【 0 0 0 7 】

ところで、画像情報のうちには、属性情報と呼ばれる画素の色以外の情報がある。例えば、その画素が、写真か文字またはグラフィックか等を表すフラグを、画像処理のためのパラメータとして、画素ごとにもつ場合である。文字と写真を分離する信号という意味で、分離情報または分離信号とも呼ばれる。属性情報は画素ごとに持つが、情報量は、画素ごとに1bitであったり2bitであったりする。この属性情報付き画像のデータフォーマットとしては、一般的に、次の2種類がある。(1) 画像は画像で1ファイルとし、属性情報は属性情報で1ファイルとして扱う。(2) 1画素に付き1属性情報をセットとして、点順次で並べて扱う。例えば、RGB(24bit)の画像に8bitの属性情報があるとすると、まとめて32bitを1画素として扱う。例えば、特許文献2に開示されている方法のように、画像と属性情報を、ある単位ブロックでパッキングする方法がある。シリアルに入力された画像と属性情報を、直接ブロック単位でパッキングしてパラレル化し、回転処理を行った後、再びシリアルに変換して出力している。これは、ビデオI/Fからのシリアル入力及びシリアル出力を意図しているものである。

【 0 0 0 8 】

画素の色深度情報と属性情報は、その特性が異なるため、圧縮する場合のアルゴリズムは異なる方が効率的である。例えば、RGB画像と文字写真分離情報では、RGBに対してJPEG等の標準圧縮方式が提案されているのに対し、文字写真分離情報は、同じ情報が並ぶため、例えば、ランレングス圧縮等が向いている。また、JPEGは画像を扱うので、人間の目をごまかせれば良いため、量子化の段階でデータの欠落が起こる非可逆性圧縮であるが、分離情報は、可逆圧縮されることが必要である。図19に示すように、文字部分と写真部分は連続しているため、分離情報も同じパターンが繰り返し発生するため、圧縮しやすい。このように、画像と付随情報は、圧縮率や情報の欠落の可否を考慮すると、別々に扱った方が効率的な面があることがわかる。

【 0 0 0 9 】

図20は、画像回転器の従来例の機能ブロック図である。メモリからのリードは、固定長符号画像がm wordであり、属性情報がn wordである。固定長圧縮画像と属性情報は、別々にa × a画素ずつブロック化されて保存されているので、画像用と属性用の2アクセスで、1ブロックのデータリードを行うことができる。データライトも2アクセス必要である。DMAコントローラは、回転前画像リード用と回転後画像ライト用、及び回転前属性リード用と回転後属性ライト用の4つが必要である。図20中の画像ブロックは、例えば、8 × 8画素のRGB画像などである。

【 0 0 1 0 】

10

20

30

40

50

図21を参照しながら、非圧縮画像を90°回転する従来の方法を説明する。非圧縮画像を90°または270°回転する場合は、メモリへのアクセスを考慮して、通常はいくつかのブロックに分けて回転を行う。例えば、図21に示すように、8画素×8ラインを1ブロックとしてブロックに番号を付ける。回転の順番は、大きく分けて2種類ある。1つ目は、A1ブロック B1ブロック C1ブロック・・・、A2ブロック B2ブロック C2ブロック・・・、・・・の順で回転を行っていく方法である。これを仮に「回転リード」と呼ぶ。回転後の並びの順番に従って読み出し、通常の順番で書き込む方法である。2つ目は、D1ブロック D2ブロック D3ブロック・・・、C1ブロック C2ブロック C3ブロック・・・、・・・の順で回転を行っていく方法である。これを仮に「回転ライト」と呼ぶ。通常の順番で読み出し、回転後の並びの順番に従って書き込む方法である。

10

【0011】

各ブロックは8ラインで構成されているため、ブロック1つにつき最低8回のメモリリードが必要である。回転リードの場合は、回転後にメモリに書き戻さなくても、ブロック-ラスタ変換バッファを用いて、直接ラスタ化が可能である。しかし、回転ライトの場合は、それは不可能である。なお、いずれの方法で回転を行う場合でも、画像とは別に属性情報も回転する必要がある。このようにして、画像をブロック単位で回転することができる。圧縮した画像の場合も、同様な方法で回転することができる。以下に、これに関連する従来技術の例をいくつかあげる。

【0012】

特許文献1に開示された「画像処理装置」は、合成画像を出力する際に使用するバッファメモリ量が小さくて済み、合成画像を作成する処理時間が短くて、さらに、画質の劣化が少ない画像処理装置である。入力された画像データを、画像圧縮伸張部で、ブロック毎に圧縮する。圧縮コードデータを、コードバッファに、ブロック毎に記憶する。コードバッファに記憶された圧縮コードデータを、コードバッファ制御部で、指定された編集モードに応じたブロックの順番で読み出す。画像圧縮伸張部は、コードバッファ制御部によって読み出された圧縮コードデータを伸張する。

20

【0013】

特許文献2に開示された「画像情報の複合処理方法」は、画像データビット数より大きいメモリデータバス幅を有効に活用して、メモリ読み書きを高速化し、画像回転角などの選択、設定の自由度を高くし、画像データと対の特徴信号を、その関係を保ったまま回転処理し、回転処理のメモリアドレス制御を簡易化する方法である。シリアル画像データを入力し、画像データの特徴を示す特徴信号を入力し、画像データと特徴信号からなる画像情報を、主走査数×副走査数のブロックパラレルデータに変換する。1アドレスにパラレルデータを記憶する。この記憶手段のデータバス幅に応じて、ブロックサイズを変更する。記憶手段から読み出すパラレルデータを、シリアル画像データと特徴信号に戻す。

30

【0014】

特許文献3に開示された「画像圧縮処理装置」は、周波数の高いエッジ部分も画質の良い画像を作成でき、かつラインごと固定された符号量であるため、ラインごとのスタートアドレスを管理するテーブルが必要なく、コストが低く、かつ信頼性の高い画像圧縮処理装置である。複数の固定長符号化手段で、画像データを複数画素のブロック単位に、相異なる符号長に変換する。所定数のブロックで発生する符号長の合計を、符号長合計計算手段で計算する。符号長合計計算手段の出力値を、所定値以下に収めるように、複数の固定長符号化手段の使用割合を制御する。

40

【特許文献1】特開平10-075345号公報

【特許文献2】特開2002-125118号公報

【特許文献3】特開2003-219188号公報

【発明の開示】

【発明が解決しようとする課題】

【0015】

しかし、従来の圧縮画像の回転方法では、画像と属性情報が別々にパッキングされてい

50

るため、メモリアクセス回数が多くなり、回転などの画像処理のためのメモリリードライ
ト時間が長くなるという問題がある。

【0016】

本発明の目的は、上記従来の問題を解決して、画像処理装置において回転処理などを行
うために、バースト転送でメモリに対して画像情報が入出力される場合のメモリアクセス
バースト数を減らし、メモリアクセス時間を短縮することである。

【課題を解決するための手段】

【0017】

本発明の画像処理装置は、画像データを記憶する記憶部と、前記画像データを正方形領
域単位でブロック化して固定長圧縮する手段と、前記画像データの属性情報を同じブロッ
ク単位でブロック化する手段と、前記正方形領域単位のブロックで固定長圧縮された前記
画像データ及び属性情報を1ブロックごとにパッキングして前記記憶部に格納する手段と
を備えた2次元ブロックデータフォーマット変換部と、前記画像データをライン方向の複
数画素単位でブロック化して固定長圧縮する手段と、前記画像データの属性情報を同じラ
イン方向のブロック単位でブロック化する手段と、前記ライン方向の複数画素単位のブロ
ックで固定長圧縮された前記画像データ及び属性情報を1ブロックごとにパッキングして
前記記憶部に格納する手段とを備えた1次元ブロックデータフォーマット変換部と、前記
2次元ブロックデータフォーマット変換部により前記記憶部に格納された主走査方向及び
副走査方向の複数の正方形領域単位ブロックにより構成されるブロック群における各正方
形領域単位ブロックの向きを固定したまま前記ブロック群を90°の整数倍回転させる画
像回転部と、前記1次元ブロックデータフォーマット変換部により前記記憶部に格納され
たパッキングデータをライン方向のブロック単位で復号してラスタデータに変換する伸
張部とを備えたことを特徴とする画像処理装置である。

【発明の効果】

【0018】

正方形領域単位ブロックの画像データを固定長圧縮したデータと属性情報をパッキング
して記憶部に格納し、パッキングしたデータを読み出して回転処理を行う構成としたので
、画像を回転する際に、記憶部に対する1回のアクセスで2次元のブロックを読み取るこ
とができ、記憶部に対するアクセス数が少なくなり、記憶部に対するアクセス時間が短
くなる。

また、ライン方向の複数画素単位で固定長圧縮された画像データと属性情報をパッキ
ングして記憶部に格納し、ラスタデータに変換して画像処理を行う構成としたので、画
像を回転しない場合は、圧縮されてパッキングされたデータを容易にラスタで取り出し
て、ライン単位で処理することができる。

【発明を実施するための最良の形態】

【0019】

以下、本発明を実施するための最良の形態について、図1～図16を参照しながら詳細に
説明する。

【実施例1】

【0020】

本発明の実施例1は、正方形ブロックの画像を固定長圧縮したデータと属性情報をパッ
キングして格納し、パッキングしたデータを読み出して回転処理を行う画像処理装置であ
る。

【0021】

本発明の実施例1における画像処理装置の基本的な構成は、従来 of 画像処理装置と同様
である。本発明の実施例1における画像処理装置は、固定長圧縮された画像と属性情報を
パッキングする点が、従来 of のものと異なる。

【0022】

図1は、本発明の実施例1における画像処理装置で用いるデータフォーマットの図であ
る。1ブロックの固定長圧縮された画像と属性情報がパッキングされた状態を示す。図2

は、画像処理装置で用いる回転器の機能ブロック図である。図3は、パッキングされたデータからラスタデータを得て、ラスタデータを画像処理する方法を示す図である。図4は、パッキングされたデータからラスタデータを得て印刷する方法を示す図である。図5は、ブロック-ラスタ変換器の機能説明図である。図6は、非圧縮画像などを2次元ブロックで固定長圧縮して、属性情報とともにパッキングする画像処理モジュールの機能ブロック図である。図7は、非圧縮画像などを2次元ブロックで固定長圧縮して、属性情報とともにパッキングし、ラスタデータに変換して画像処理を行うモジュールの機能ブロック図である。図8は、2次元ブロックで固定長圧縮して、属性情報とともにパッキングした画像データを回転する方法を示す図である。

【0023】

図2～図8において、画像回転器1は、ブロック単位の画像データを、ブロック内において画素単位で回転してブロック全体を回転する手段である。分離手段2は、固定長圧縮画像と属性情報を分離する手段である。伸張器3は、固定長圧縮画像を非圧縮画像に戻す手段である。画像回転手段4は、画像データを、画素単位で回転する手段である。ここでいう回転は、90°単位の回転である。属性情報回転手段5は、属性情報を、画素単位で回転する手段である。固定長圧縮手段6は、非圧縮画像を固定長圧縮する手段である。パッキング手段7は、固定長圧縮画像と属性情報を、1つのデータにまとめる手段である。ブロック-ラスタ変換器8は、ブロック単位の画像データからラスタデータを生成する手段である。画像処理部9は、ラスタデータを画像処理して新しいラスタデータを生成する手段である。プロッタ10は、ラスタデータ単位で印刷する手段である。CPU11は、画像処理のプログラムを実行する演算処理装置である。メモリ12は、画像処理のプログラムと画像データを記憶する手段である。データフォーマット変換器13は、非圧縮画像などから、ブロック単位の固定長圧縮画像と属性情報とのパッキングデータに変換する手段である。復号回転器14は、ブロック単位の固定長圧縮画像と属性情報とのパッキングデータを復号するとともに画像を回転する手段である。

【0024】

上記のように構成された本発明の実施例1における画像処理装置の機能と動作を説明する。最初に、図1を参照しながら、固定長圧縮した画像情報と、属性情報を、パッキングしてメモリに格納する方法を説明する。矩形の1ブロックの画像情報を固定長圧縮する。矩形ブロックは、例えば、画面上で8×8画素のブロックである。矩形ブロックは、長方形のブロックでも正方形のブロックでもよい。このブロックの画素の属性情報をまとめて、固定長圧縮画像情報の直後のアドレスに格納する。属性情報は、2次元ブロックにパッキングされていてよい。これを繰り返すことにより、メモリの一定アドレスごとに、1ブロックの画像情報と属性情報が連続して記憶される。

【0025】

次に、図2を参照しながら、画像処理装置で用いる画像回転器の動作を説明する。1ブロックの固定長圧縮画像のデータサイズは、 m wordである。それに対応する属性情報のデータサイズは、 n wordである。 m と n は、画像の特性や固定長圧縮方式により決まる一定の値である。したがって、メモリからのリードは、1アクセスにつき $(m+n)$ wordである。固定長圧縮画像及び属性情報は、 $a \times a$ 画素ずつブロック化されているので、1アクセスで、1ブロックのデータリードを行うことができる。 a は、例えば8や16である。データライトも同じく1アクセスでできる。DMAコントローラは、回転前データリード用と回転後データライト用の2つが必要である。図2中の画像ブロックは、例えば、8×8画素のRGB画像などである。

【0026】

画像回転器1において、最初に、分離手段2により固定長圧縮画像と属性情報を分離する。属性情報だけを、属性情報回転手段5により回転する。一方、固定長圧縮画像は、伸張器3により復号して非圧縮画像にする。この非圧縮画像を、画像回転手段4により回転する。回転した非圧縮画像を、固定長圧縮手段6により固定長圧縮する。固定長圧縮画像と属性情報を、パッキング手段7によりパッキングする。パッキングされたデータの

10

20

30

40

50

フォーマットとサイズは、回転前のパッキングデータのフォーマットとサイズに等しい。このようにして、画像回転器 1 は、1 ブロックの固定長圧縮画像と属性情報を回転する。

【 0 0 2 7 】

次に、図 3 と図 4 と図 5 を参照しながら、正方形ブロックの画像を固定長圧縮した画像データと属性情報から、ラスターデータを得て、ラスターデータを画像処理する方法を説明する。図 3 に示すように、 $a \times a$ 画素単位でブロック化された固定長圧縮画像データをラスター単位で画像処理する場合、伸張後にブロック - ラスター変換器が必要である。まず、 $a \times a$ 画素単位でブロック化された固定長圧縮画像データを、伸張器 3 で復号して非圧縮画像にする。 $a \times a$ 画素単位でブロック化された非圧縮画像を、ブロック - ラスター変換器 8 で、 a ライン分のバッファを使ってラスターデータに変換する。ブロック - ラスター変換器 8 については後述する。このラスターデータを、画像処理部 9 で処理して、処理後の画像のラスターデータを得る。

10

【 0 0 2 8 】

また、図 4 に示すように、 $a \times a$ 画素単位でブロック化された固定長圧縮画像データを印刷する場合も、伸張後にブロック - ラスター変換器が必要である。まず、 $a \times a$ 画素単位でブロック化された固定長圧縮画像データを、伸張器 3 で復号して非圧縮画像にする。 $a \times a$ 画素単位でブロック化された非圧縮画像を、ブロック - ラスター変換器 8 で、 a ライン分のバッファを使ってラスターデータに変換する。このラスターデータを、プロッタ

【 0 0 2 9 】

ブロック - ラスター変換器 8 では、図 5 に示すように、ブロックをラスターに変えるためには、1 ページ幅分のラインバッファを最低 a ライン分持つ必要がある。画像データは、ブロック単位で圧縮されてパッキングされているため、伸張後の 1 ブロックの画像は、図 5 の斜線部のブロックとなる。ブロック単位で解凍されたデータを、正方形ブロック単位でバッファに書き込んでいく。 a ライン分のデータを書き込んだら、ライン単位で読み取る。ブロック - ラスター変換を高速に行うには、 $a \times 2$ ラインのラインバッファを備えて、トグル処理を行う必要がある。しかし、1 ページの幅が大きければ大きいほど、ハードウェアのサイズが大きくなり、コストがかかるので、高速処理が必要ない場合は、交代バッファにしないでコストを削減する。

20

【 0 0 3 0 】

次に、図 6 を参照しながら、2 次元ブロック（正方形ブロック）の画像情報を固定長圧縮して、属性情報とともにパッキングするハードウェアモジュールについて説明する。入力画像データの例として示してある可変長符号画像は、例えば JPEG のように、 8×8 画素（サブサンプリングされている場合は 16×16 画素）ごとに可変長圧縮されている画像である。入力画像としては、可変長符号の他に、非圧縮画像でも固定長圧縮画像でもよい。入力画像は、何らかの手段で入力されて、データフォーマット変換器 13 により非圧縮画像に戻されて、属性情報とともにメモリ 12 に格納されているものとする。入力画像は、データフォーマット変換器 13 により、 8×8 画素のブロックごとに固定長圧縮されて、属性情報とともにパッキングされて、図 1 に示したフォーマットでメモリ 12 に書き込まれる。

30

【 0 0 3 1 】

ブロック単位で圧縮された可変長符号（JPEG 等）を、ブロック単位で圧縮された固定長符号に変換し、属性情報をブロック単位で付加してパッキングしてメモリに保存すると、次の 2 つのメリットが得られる。すなわち、（1）画像の回転を行う際に、非圧縮画像に展開する必要がなく、メモリ量を削減できる。さらに、（2）メモリアクセス回数が少なくなるとメモリアクセスが早くなり、他のデバイスのメモリアクセスも妨げない。ところで、可変長符号（JPEG）は、復号しつつ回転することは可能だが、メモリに書き込む際に、再び可変長符号（JPEG）にして書き込むことはできない。ただし、非圧縮画像または固定長圧縮画像にして書き戻すことは可能である。最初から非圧縮画像でメモリに保存されている場合、または、隣のブロック同士に相関の無い固定長符号でメモリに保存されている場合は、そのまま回転できる。固定長符号と属性情報のパッキングデータにフォーマッ

40

50

ト変換をしておく、後で回転処理がしやすいというメリットがある。

【0032】

次に、図7を参照しながら、2次元ブロック(正方形ブロック)の画像情報を固定長圧縮して、属性情報とともにパッキングし、パッキングしたデータを回転してラスタータに変換し、ラスタータを画像処理するハードウェアモジュールを説明する。入力画像を、データフォーマット変換器13により、8×8画素のブロックごとに固定長圧縮して、属性情報とともにパッキングしてメモリ12に書き込むところまでは、図6のモジュールと同じである。パッキングされたデータを、復号回転器14により復号して非圧縮画像にし、さらに回転して、回転した非圧縮画像とする。具体的な回転方法は後述する。回転した非圧縮画像を、ブロック-ラスタータ変換器8でラスタータに変換する。このラスタータデータを、画像処理部9で処理して、処理後の画像のラスタータデータを得る。

10

【0033】

固定長符号と属性情報のパッキングデータがメモリ12に保存されている状態であれば、各ブロックの先頭アドレスとブロックのデータ長が分かっている。したがって、90°または180°または270°方向に回転するように、ブロックごとにメモリ12からデータを読み出すことができる。ブロックを復号回転器14で回転した後にラスタータに変換すれば、メモリ12に書き戻すことなしに、直接ライン単位で処理する画像処理部9に入力できる。この場合のメリットは、(1)メモリに非圧縮画像を置くことなく、90°や270°の画像回転が可能であり、低コストとなる。(2)固定長符号と属性情報がパッキングされているので、メモリアクセス数が少なくなり、パフォーマンスが向上する。特に、復号回転後にメモリ12にデータを書き戻すことなくラスタータに変換し、画像処理部9に入力することができるので、メモリ容量が少なく済み、パフォーマンスも格段に向上するというメリットがある。

20

【0034】

次に、図8を参照しながら、ブロック化された画像データを90°回転する方法を説明する。8画素×8ラインの正方形ブロックごとに固定長圧縮された画像データと属性情報が、1つのデータにパッキングされて、メモリに連続的に格納されている。従来の非圧縮画像同様に、「回転リード」(回転したアドレスでリードする)と「回転ライト」(回転したアドレスでライトする)が可能である。各ブロックのデータはすでにパッキングされているため、一度のメモリリードアクセスで、8×8画素のブロックを読み取ることができる。各ブロックのデータ長は、固定長圧縮のアルゴリズムと属性情報の大きさによって変化する。固定長圧縮のため、各ブロックのスタートアドレスは決まっており、次のブロックのスタートアドレスは、各ブロックのデータ長に依存して決まる。ブロック1つあたりのデータ長を考慮して、「回転リード」(回転後の並びの順番に従って読み出す方法)または「回転ライト」(回転後の並びの順番に従って書き込む方法)を行う。そのためのメモリリード及びメモリライトアドレス発生を行う。図7の場合は、回転リードを行う必要があり、回転ライトはできない。

30

【0035】

「回転リード」の場合は、次のようにメモリリードアドレスとメモリライトアドレスを発生する。最初に、A1ブロックを読み出すために、A1ブロックの先頭アドレスを求める。(全体の先頭アドレス)+(ブロックサイズ)×(1ラインのブロック数)×(1列のブロック数-1)で求まる。A1ブロックを読み出して回転し、回転したA1ブロックを回転後のメモリ領域の先頭アドレスに書き込む。書き込みの順序は、通常のラスタータ順と同じである。次に、B1ブロックを読み出す。このアドレスは、(A1ブロックの先頭アドレス)-(ブロックサイズ)×(1ラインのブロック数)で求まる。B1ブロックを回転し、回転したB1ブロックを、回転後のメモリ領域のA1ブロックの次のアドレスに書き込む。これを繰り返す。1列の回転が終わったら、A2ブロックを読み出す。A2ブロックの先頭アドレスは、(A1ブロックの先頭アドレス)+(ブロックサイズ)で求まる。回転後のA2ブロックは、単に前回の次のアドレスに書き込めばよい。

40

【0036】

50

「回転ライト」の場合は、次のようにメモリリードアドレスとメモリライトアドレスを発生する。読出しの順序は、通常のラスタ順と同じである。すなわち、最初にD1ブロックを読み出し、以下、ラスタ順に次々にブロックを読み出す。D1ブロックを回転して、回転後のD1ブロックを、回転後のメモリ領域の対応アドレスに書き込む。D1ブロックの書込みアドレスは、(書込領域の先頭アドレス) + (ブロックサイズ) × (1列のブロック数 - 1) で求まる。回転後のD2ブロックは、(D1ブロックの先頭アドレス) + (ブロックサイズ) × (1列のブロック数) で求まる。これを繰り返し、1ライン分の回転が終わったら、C1ブロックを読み出す。C1ブロックの書込みアドレスは、(D1ブロックの先頭アドレス) - (ブロックサイズ) で求まる。

【0037】

上記のように、本発明の実施例1では、画像処理装置を、正方形ブロックの画像を固定長圧縮したデータと属性情報をパッキングして格納し、パッキングしたデータを読み出して回転処理を行う構成としたので、画像を回転する際に、1回のメモリアクセスで2次元のブロックを読み取ることができ、メモリアクセス数が少なくなり、メモリアクセス時間が短くなる。

【実施例2】

【0038】

本発明の実施例2は、ライン方向の8画素のブロック単位で固定長圧縮された画像データと属性情報をパッキングしてメモリに格納し、ラスタデータに変換して画像処理を行う画像処理装置である。

【0039】

本発明の実施例2における画像処理装置の基本的構成は実施例1と同じである。本発明の実施例2における画像処理装置が実施例1と異なるところは、入力画像をライン方向の複数画素単位で固定長圧縮し、その属性情報とともにパッキングする点である。

【0040】

図9は、本発明の実施例2における画像処理装置で用いる回転器の機能ブロック図である。図10は、画像処理装置で、パッキングされたデータからラスタデータを得て画像処理する場合を示す図である。図11は、画像処理装置で、パッキングされたデータからラスタデータを得て印刷する場合を示す図である。図12は、非圧縮画像などを1次元ブロックで固定長圧縮して、属性情報とともにパッキングする画像処理モジュールの機能ブロック図である。図13は、1次元ブロックで固定長圧縮して属性情報とともにパッキングされたデータのイメージである。図14は、非圧縮画像などを1次元ブロックで固定長圧縮して、属性情報とともにパッキングし、パッキングした画像データを回転し、復号し、ラスタ画像を処理する画像処理モジュールの機能ブロック図である。図15は、非圧縮画像などを1次元ブロックで固定長圧縮して、属性情報とともにパッキングし、パッキングした画像データを復号して回転し、ブロックデータをラスタデータに変換し、ラスタデータを画像処理する画像処理モジュールの機能ブロック図である。図16は、回転アクセスアドレス発生方法の説明図である。

【0041】

図9～図16において、画像処理部9は、ラスタデータを画像処理して新しいラスタデータを生成する手段である。プロッタ10は、ラスタデータ単位で印刷する手段である。CPU11は、画像処理のプログラムを実行する演算処理装置である。メモリ12は、画像処理のプログラムと画像データを記憶する手段である。画像回転器21は、正方形ブロックにまとめた画像データを、正方形ブロック内において画素単位で回転して正方形ブロック全体を回転する手段である。分離手段22は、固定長圧縮画像と属性情報を分離する手段である。伸張器23は、固定長圧縮画像を非圧縮画像に戻す手段である。画像回転手段24は、画像データを、画素単位で回転する手段である。属性情報回転手段25は、属性情報を、画素単位で回転する手段である。固定長圧縮手段26は、非圧縮画像をライン方向の複数画素単位、例えば8画素単位で固定長圧縮する手段である。パッキング手段27は、固定長圧縮画像と属性情報を、1つのデータにまとめる手段である。ブロック - ラスタ変換器28は、

10

20

30

40

50

正方形ブロック単位の画像データからラスターデータを生成する手段である。データフォーマット変換器29は、非圧縮画像などから、ライン方向の複数画素単位の固定長圧縮画像と属性情報とのパッキングデータに変換する手段である。パッキングデータ回転器30は、ライン方向の複数画素単位のパッキングデータを正方形ブロックにして回転する手段である。パッキングデータ復号器31は、ライン方向の複数画素単位のパッキングデータを復号してラスターデータにする手段である。パッキングデータ復号回転器32は、ライン方向の複数画素単位の固定長圧縮画像と属性情報とのパッキングデータを復号するとともに画像を回転する手段である。

【0042】

上記のように構成された本発明の実施例2における画像処理装置の機能と動作を説明する。最初に、ライン方向の複数画素単位で圧縮することの意味を説明する。画像を回転する場合は、実施例1の正方形ブロック単位の固定長圧縮が有利である。画像を回転しないでラスターで扱いたい場合は、ライン方向の複数画素単位の固定長圧縮が有利である。画像回転器は、2次元のブロックで画像を読み取り、回転してメモリに書き戻すので、ライン方向の複数画素単位でパッキングされたa画素分のパッキングデータは、副走査方向にaライン分読み取る必要がある。ライン方向の複数画素ごとに圧縮されてパッキングされたデータは、ラスターで取り出すことが容易である。画像処理はラスター単位で行うことも多いので、伸張後の画像データを直接画像処理部に入力したり、ラスター単位での入力が必要なプロッタに出力したりする場合に有利な画像フォーマットといえる。

【0043】

次に、図9を参照しながら、画像処理装置で用いる画像回転器の動作を説明する。1ライン(a画素)の固定長圧縮画像のデータサイズは、p wordである。それに対応する属性情報のデータサイズは、q wordである。したがって、メモリからのリードは、1アクセスにつき(p+q) wordである。固定長圧縮画像及び属性情報は、a画素ずつ符号化されているので、a回のアクセスで、1ブロックのデータを読み出すことができる。データライトも同じく、a回のアクセスでできる。DMAコントローラは、回転前データリード用と回転後データライト用の2つが必要である。図9中の画像ブロックは、例えば、8画素のRGB画像などである。

【0044】

画像回転器21では、最初に、分離手段22で固定長圧縮画像と属性情報を分離する。属性情報を、属性情報回転手段25で回転する。回転した属性情報を正方形ブロックにして、ライン方向の複数画素ごとに属性情報を取り出す。固定長圧縮画像を、伸張器23で復号して、非圧縮画像にする。非圧縮画像を、画像回転手段24で回転する。回転した非圧縮画像を、正方形ブロックにして、ライン方向の複数画素ごとに固定長圧縮手段26で固定長圧縮する。固定長圧縮画像と属性情報を、パッキング手段27でパッキングする。パッキングされたデータのフォーマットとサイズは、回転前のパッキングデータのフォーマットとサイズに等しい。このようにして、画像回転器21は、a画素を1ブロックとした固定長圧縮画像と属性情報を回転する。

【0045】

次に、図10を参照しながら、ライン方向の複数画素単位で圧縮することのメリットを説明する。メモリから、a画素単位でパッキングしたデータを読み込む。a画素単位でパッキングしたデータを、伸張器23で非圧縮画像に戻すと、a画素のラスターデータになる。このラスターデータを、画像処理部9で処理して、処理後のラスターデータを得る。メモリからa画素単位でパッキングしたデータを読み込んで伸張すると、伸張後のデータはラスターなので、画像を回転しないでライン単位で処理する場合は、そのまま画像処理部に入力できる。

【0046】

次に、図11を参照しながら、画像を回転しないでプロッタ出力する場合を説明する。メモリから、a画素単位でパッキングしたデータを読み込む。a画素単位でパッキングしたデータを、伸張器23で非圧縮画像に戻すと、a画素のラスターデータになる。このラスター

10

20

30

40

50

ーデータをプロッタ10で印刷する。プロッタにはラスターで画像を入力する必要がある。メモリから a 画素単位でパッキングしたデータを読み込んで伸張すると、圧縮データはライン方向に a 画素分ずつパッキングされているため、伸張後の画像もラスターで得られる。伸張後の画像データをメモリに書き戻すことなく、直接プロッタに出力できる。

【 0 0 4 7 】

次に、図12を参照しながら、非圧縮画像などを1次元ブロックで固定長圧縮して、属性情報とともにパッキングする画像処理モジュールについて説明する。図12では、入力データの例として、1次元可変長符号画像と固定長符号画像と非圧縮画像を示してある。1次元とあるのは、データの圧縮方向が1次元であることを示しており、実際には、2次元の画像を扱っている。入力画像は、何らかの手段で入力されて、データフォーマット変換器29により非圧縮画像に戻されて、属性情報とともにメモリ12に格納されているものとする。入力画像は、データフォーマット変換器29により、ラスター方向の8画素のブロックごとに固定長圧縮されて、属性情報とともにパッキングされて、図1に示したフォーマットでメモリ12に書き込まれる。

10

【 0 0 4 8 】

次に、図13を参照しながら、ライン方向の複数画素単位に固定長圧縮して属性情報とともにパッキングする方法を説明する。1ラインの画像データを、a画素ずつのブロックに分割する。a画素のブロックを固定長圧縮して、その属性情報とともに1つにまとめてメモリに格納する。90°回転する際には、このデータフォーマットでは、正方形ブロックになるように、aライン分のパッキングデータをメモリから読み取る。aライン分のパッキングデータを復号して、正方形ブロックにした後、画像回転を行う。回転した正方形ブロックを、再びa画素ごとに、a個の固定長符号に圧縮しなおす。このようにして回転を行うことができるので、画像全体を非圧縮画像に展開する必要はない。メモリ量が削減できるという利点はあるが、メモリアクセス数は実施例1より多い。

20

【 0 0 4 9 】

回転時のメモリアクセス数では実施例1に劣るが、ラスター変換をバッファなしに行うことができる点が、実施例1より優れている。可変長符号は、復号しつつ回転することは可能だが、メモリに書き込む際に、再び可変長符号(JPEG)にして書き込むことはできない。非圧縮画像または固定長圧縮画像にして書き戻すことは可能である。最初から、非圧縮画像または隣のブロック同士に相関の無い固定長符号でメモリに保存されている場合は、そのまま回転できるが、固定長符号と属性のパッキングデータにフォーマット変換をしておく、後で回転処理がしやすいというメリットがある。

30

【 0 0 5 0 】

次に、図14を参照しながら、非圧縮画像などを1次元ブロックで固定長圧縮して、属性情報とともにパッキングし、パッキングした画像データを回転し、復号し、ラスターデータを処理する画像処理モジュールについて説明する。入力画像を、データフォーマット変換器29により、8画素のブロックごとに固定長圧縮して、属性情報とともにパッキングしてメモリ12に書き込むところまでは、図12のモジュールと同じである。回転する場合は、パッキングデータ回転器30で回転する。パッキングデータ回転器30は、リードもライトもメモリ12に対して行うため、回転リードも回転ライトも可能である。パッキングされたデータを、パッキングデータ復号器31により復号してラスターデータにする。ラスターデータを、画像処理部9で処理して、処理後の画像のラスターデータを得る。

40

【 0 0 5 1 】

1次元ブロックで固定長圧縮して、属性情報とともにパッキングした画像データの場合、パッキングデータを復号してラスターデータに変換する際に、ブロックラスター変換用ラインバッファを持つ必要がない。しかしながら、画像を回転する際には、復号と回転を行いつつ、ラスターに変換することは不可能であるため、ブロックラスター変換用ラインバッファを持つ必要がある。あるいは、回転専用のモジュールを別に持って、メモリ経由であらかじめ回転を行っておく必要がある。

【 0 0 5 2 】

50

次に、図15を参照しながら、非圧縮画像などを1次元ブロックで固定長圧縮して、属性情報とともにパッキングした画像データを復号して回転し、ブロックデータをラスタデータに変換し、ラスタデータを画像処理する画像処理モジュールについて説明する。入力画像を、データフォーマット変換器29により、8画素のブロックごとに固定長圧縮して、属性情報とともにパッキングしてメモリ12に書き込むところまでは、図12のモジュールと同じである。パッキングされたデータを、パッキングデータ復号回転器32により復号して回転し、回転したブロックデータにする。この場合は、回転リードを行う必要があり、回転ライトは不可である。ブロックデータを、ブロック-ラスタ変換器28でラスタデータにする。ラスタデータを、画像処理部9で処理して、処理後の画像のラスタデータを得る。非圧縮画像などを1次元ブロックで固定長圧縮して、属性情報とともにパッキングした画像データの場合も、復号と回転を同時に行うためには、ブロックラスタ変換用ラインバッファを持つ必要がある。

10

【0053】

次に、図16を参照しながら、回転アクセスアドレス発生方法を説明する。データが、画像と属性情報を含めて8画素ごとにパッキングされているとする。従来例の非圧縮画像同様に、「回転リード」（回転後の並びの順番に従って読み出す方法）または「回転ライト」（回転後の並びの順番に従って書き込む方法）が可能である。各ブロックは、8画素ごとにパッキングされているため、ブロック1つにつき最低8回のメモリリードが必要である。回転リードの場合は、回転後にメモリに書き戻さなくても、ブロックラスタ変換バッファを用いて、直接ラスタ化が可能である。回転ライトの場合は、直接ラスタ化は不可能である。

20

【0054】

「回転リード」の場合は、次のようにメモリリードアドレスとメモリライトアドレスを発生する。最初に、A1ブロックの第1ラインブロックを読み出すために、その先頭アドレスを求める。 $(\text{全体の先頭アドレス}) + (\text{ラインブロックサイズ}) \times (\text{1ラインのブロック数}) \times (\text{1列のラインブロック数} - 8)$ で求まる。A1ブロックの第2ラインブロックの先頭アドレスは、 $(\text{A1ブロックの第1ラインブロックの先頭アドレス}) + (\text{ラインブロックサイズ}) \times (\text{1ラインのブロック数})$ となる。このようにして、A1ブロックの第8ラインブロックまで読み出す。A1ブロックを回転し、回転したA1ブロックを回転後のメモリ領域の先頭アドレスに書き込む。書込みの順序は、通常ラスタ順と同じである。ただし、A1ブロックの第2ラインブロックは、 $(\text{A1ブロックの第1ラインブロックの書込み先頭アドレス}) + (\text{ラインブロックサイズ}) \times (\text{1ラインのブロック数})$ となる。以下同様である。次に、B1ブロックを読み出す。この読出しアドレスは、 $(\text{A1ブロックの先頭アドレス}) - (\text{ラインブロックサイズ}) \times (\text{1ラインのブロック数}) \times 8$ で求まる。B1ブロックを回転し、回転したB1ブロックを、回転後のメモリ領域のA1ブロックの次のアドレスに書き込む。これを繰り返す。1列の回転が終わったら、A2ブロックを読み出す。A2ブロックの読出し先頭アドレスは、 $(\text{A1ブロックの読出し先頭アドレス}) + (\text{正方形ブロックサイズ})$ で求まる。回転後のA2ブロックは、正方形ブロックとしては、単に前回の正方形ブロックの次のアドレスに書き込めばよい。

30

【0055】

「回転ライト」の場合は、次のようにメモリリードアドレスとメモリライトアドレスを発生する。読出しの順序は、通常ラスタ順と同じである。すなわち、最初にD1ブロックを読み出し、以下、ラスタ順に次々にブロックを読み出す。ただし、D1ブロックの第2ラインブロックの先頭アドレスは、 $(\text{D1ブロックの第1ラインブロックの先頭アドレス}) + (\text{ラインブロックサイズ}) \times (\text{1ラインのブロック数})$ となる。D1ブロックを回転して、回転後のD1ブロックを、回転後のメモリ領域の対応アドレスに書き込む。D1ブロックの第1ラインブロックの書込みアドレスは、 $(\text{書込領域の先頭アドレス}) + (\text{ラインブロックサイズ}) \times (\text{1列のブロック数} - 1)$ で求まる。D1ブロックの第2ラインブロックの書込みアドレスは、 $(\text{D1ブロックの第1ラインブロックの先頭アドレス}) + (\text{ラインブロックサイズ}) \times (\text{1列のブロック数})$ である。回転後のD2ブロックの第1ラインブロック

40

50

の書込み先頭アドレスは、(D1ブロックの書込み先頭アドレス) + (正方形ブロックサイズ) × (1列のブロック数) で求まる。これを繰り返し、8ライン分の回転が終わったら、C1ブロックを読み出す。C1ブロックの書込みアドレスは、(D1ブロックの書込み先頭アドレス) - (正方形ブロックサイズ) で求まる。

【0056】

1次元ブロックで固定長圧縮して、属性情報とともにパッキングした画像データのメモリを以下に示す。

(1) 属性情報と画像情報がパッキングされているため、メモリアクセス数を1/2にできる。

(2) 画像自体が固定長圧縮されているため、メモリアクセス時間が短い。

(3) 実施例1のデータフォーマットと比べて、回転しない場合は、メモリからリードした後、ブロックラスタ変換バッファなしで、直接ラスタデータに変換できる。

【0057】

上記のように、本発明の実施例2では、画像処理装置を、ライン方向の8画素単位で固定長圧縮された画像データと属性情報をパッキングしてメモリに格納し、ラスタデータに変換して画像処理を行う構成としたので、圧縮されてパッキングされたデータを容易にラスタで取り出して、ライン単位で処理することができる。

【産業上の利用可能性】

【0058】

本発明の画像処理装置は、固定長圧縮した画像データを高速に回転処理する画像処理装置として最適である。

【図面の簡単な説明】

【0059】

【図1】本発明の実施例1における画像処理装置で用いるデータフォーマットの図である。

【図2】本発明の実施例1における画像処理装置で用いる回転器の機能ブロック図である。

【図3】本発明の実施例1における画像処理装置で、パッキングされたデータからラスタデータを取得する方法を示す図である。

【図4】本発明の実施例1における画像処理装置で、パッキングされたデータからラスタデータを取得して印刷する方法を示す図である。

【図5】本発明の実施例1における画像処理装置のブロック-ラスタ変換器の機能説明図である。

【図6】本発明の実施例1における画像処理装置の画像処理モジュールの機能ブロック図である。

【図7】本発明の実施例1における画像処理装置の他の画像処理のモジュールの機能ブロック図である。

【図8】本発明の実施例1における画像処理装置で画像を回転する方法の説明図である。

【図9】本発明の実施例2における画像処理装置で用いる回転器の機能ブロック図である。

【図10】本発明の実施例2における画像処理装置で、パッキングされたデータからラスタデータを取得して画像処理する場合を示す図である。

【図11】本発明の実施例2における画像処理装置で回転しないでプロッタ出力する場合を示す図である。

【図12】本発明の実施例2における画像処理装置の画像処理モジュールの機能ブロック図である。

【図13】本発明の実施例2における画像処理装置で固定長圧縮画像を属性情報とともにパッキングしたデータの概念図である。

【図14】本発明の実施例2における画像処理装置の画像処理モジュールの機能ブロック図である。

10

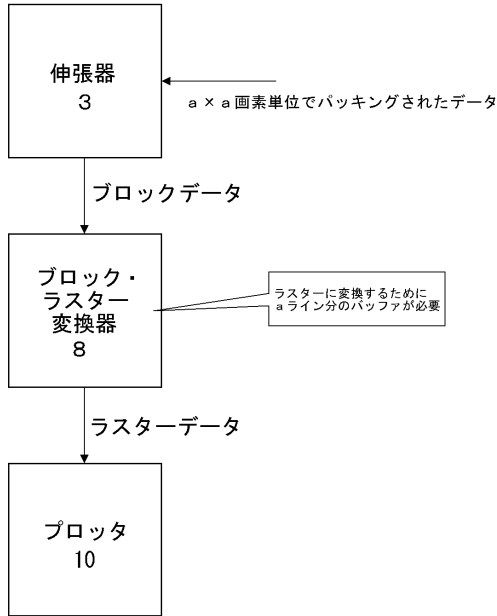
20

30

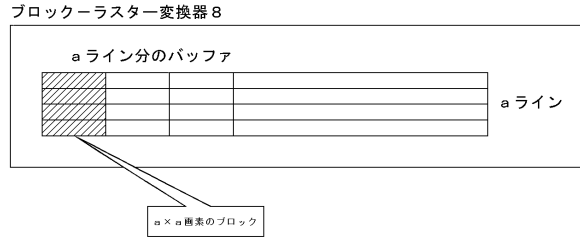
40

50

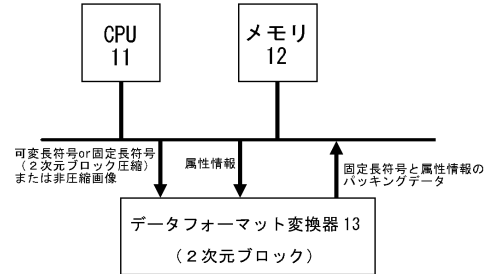
【図4】



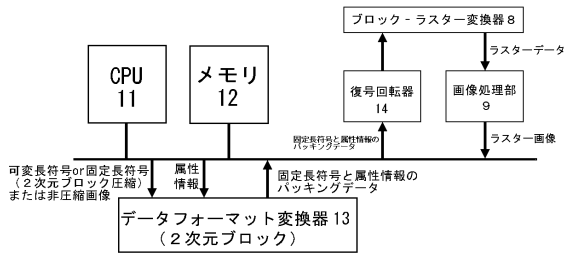
【図5】



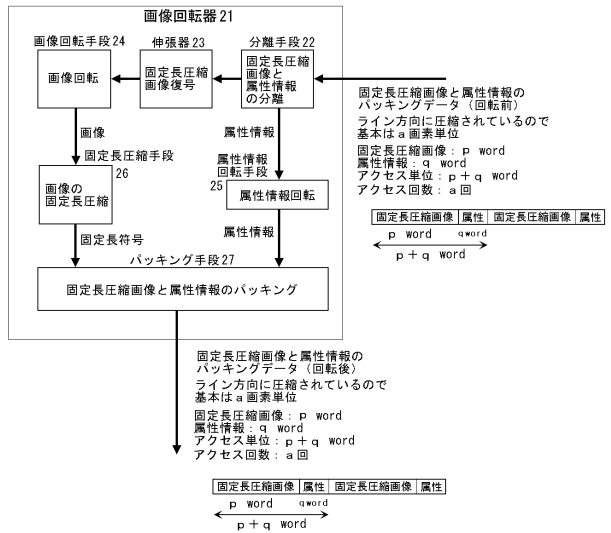
【図6】



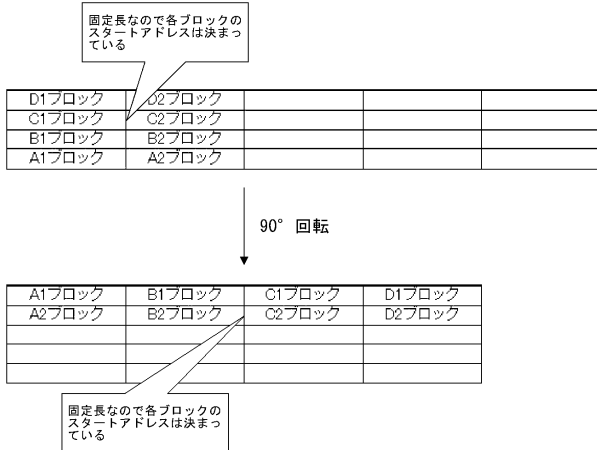
【図7】



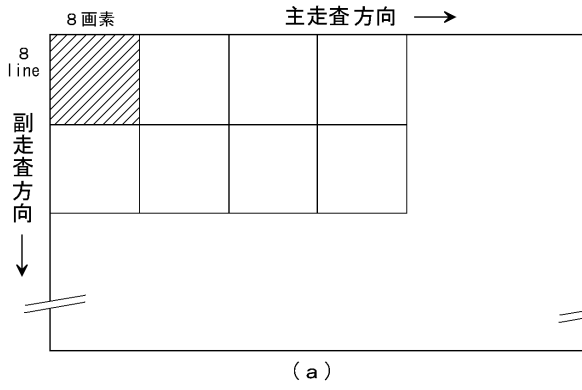
【図9】



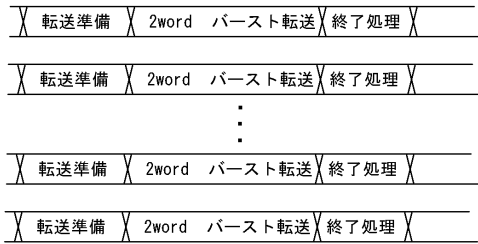
【図8】



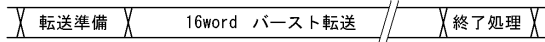
【図 17】



(a)

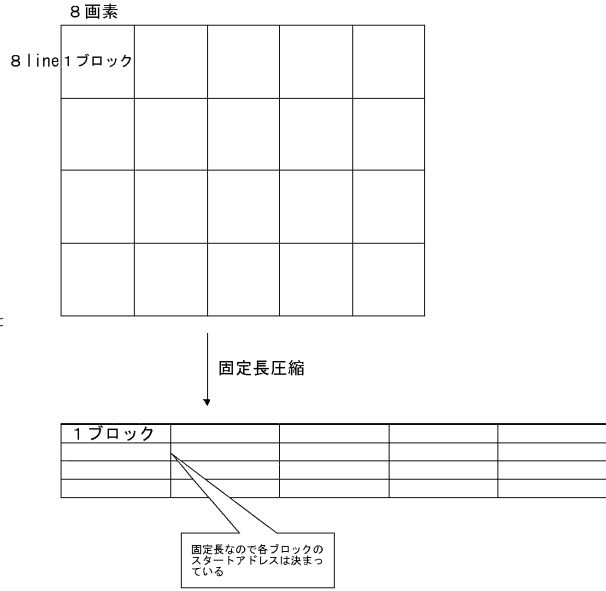


(b)

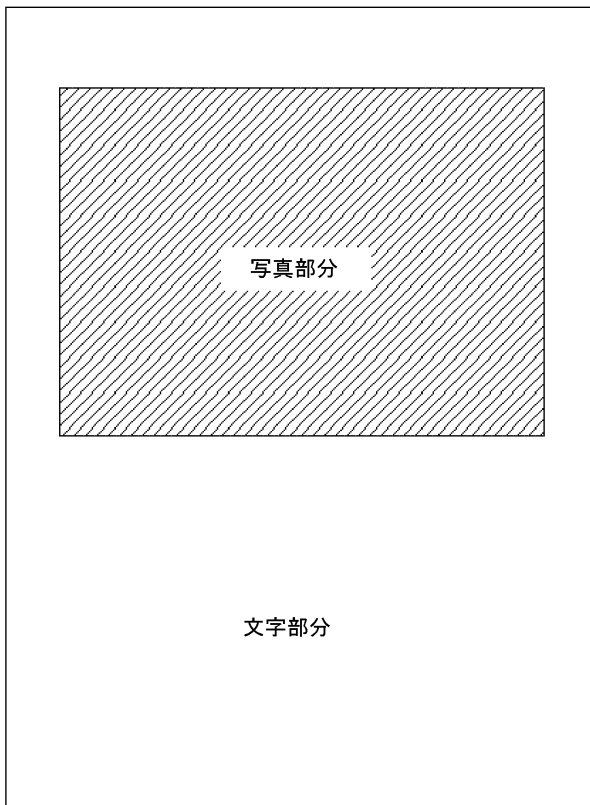


(c)

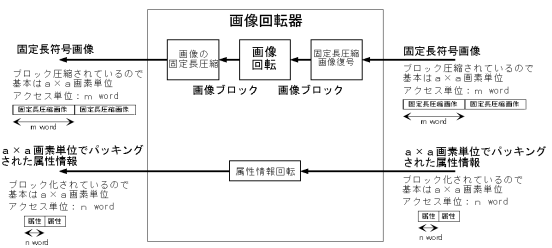
【図 18】



【図 19】



【図 20】



【図 2 1】



フロントページの続き

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

(58)調査した分野(Int.Cl. , D B 名)
H 0 4 N 1 / 3 8 7
H 0 4 N 1 / 4 1