(19)日本国特許庁(JP)　(12)特 許 公 報(B2)　(11)特許番号
特許第4084802号
(P4084802)

(45)発行日 平成20年4月30日(2008.4.30)　(24)登録日 平成20年2月22日(2008.2.22)

(51)Int.Cl.		F I		
H04N 1/41	**(2006.01)**	H04N	1/41	C
H04N 11/04	**(2006.01)**	H04N	11/04	A

請求項の数 10　(全 12 頁)

(21)出願番号 特願2004-565444 (P2004-565444)
(86)(22)出願日 平成15年12月12日 (2003.12.12)
(65)公表番号 特表2006-512019 (P2006-512019A)
(43)公表日 平成18年4月6日 (2006.4.6)
(86)国際出願番号 PCT/US2003/039713
(87)国際公開番号 WO2004/061774
(87)国際公開日 平成16年7月22日 (2004.7.22)
審査請求日 平成18年9月11日 (2006.9.11)
(31)優先権主張番号 10/324,582
(32)優先日 平成14年12月19日 (2002.12.19)
(33)優先権主張国 米国 (US)

(73)特許権者 503003854
ヒューレット－パッカード デベロップメント カンパニー エル．ピー．
アメリカ合衆国 テキサス州 ７７０７０ ヒューストン ２０５５５ ステイト ハイウェイ ２４９
(74)代理人 110000039
特許業務法人アイ・ピー・エス
(72)発明者 ガディエル・セロウッシ
アメリカ合衆国カリフォルニア州クパーティノ ミルキイウェイ１１２３
(72)発明者 マルセロ・ワインバーガー
アメリカ合衆国カリフォルニア州サンノゼ タッカードライブ６３７８

最終頁に続く

(54)【発明の名称】画像を処理する装置

(57)【特許請求の範囲】

【請求項１】

画像を処理する装置であって、

異なる色度コンテキストに対応する複数のカラーキャッシュにアクセスし（１１４）、

処理中のピクセル値のカラーキャッシュを選択し（１１８）、

前記選択されたカラーキャッシュ内の情報を使用して、前記処理中のピクセルの値を予測する（１２０）マシン（４１０）

を備えた装置。

【請求項２】

前記処理中のピクセルのすべての色成分は、同時に符号化される

請求項１記載の装置。

【請求項３】

カラーキャッシュが、因果的近傍ピクセルの色度値から、色度コンテキストに従って選択される（１１８）

請求項１記載の画像を処理する装置。

【請求項４】

選択されたキャッシュ内の情報には、前記処理中のピクセルの予測輝度値を使用することによってインデックスが付与され（１２０）、

前記予測輝度値は、因果的近傍ピクセルから計算される（１１９）

請求項３記載の装置。

【請求項５】
前記マシンは、前記処理中のピクセルの実際のピクセル値が、前記選択されたカラーキャッシュ内にまだ含まれていない場合に、前記実際のピクセル値を、前記カラーキャッシュに加える（１２６）
請求項１記載の装置。
【請求項６】
前記マシンは、前記カラーキャッシュから見込みの低いピクセル値を除去する（１２６）
請求項１記載の画像を処理する装置。
【請求項７】
前記マシンは画像圧縮を行う
請求項１記載の画像を処理する装置。
【請求項８】
前記マシンは、コードワードを使用して、前記処理中のピクセルの前記予測値と前記実値の間の誤差ベクトルを記述し、前記誤差ベクトルを前記コードワードによって記述することができない場合、バックアップコーダを使用する（１２４）
請求項７記載の装置。
【請求項９】
前記マシンは、
前記処理中のピクセルについて前記選択されたカラーキャッシュに対するインデックスを予測し、
前記選択されたカラーキャッシュ内の実際のインデックスを求め、コードワードを使用して、前記予測されたインデックスと前記実際のインデックスの間の誤差を記述し、
前記誤差を計算することができない場合、または、前記コードワードによって記述することができない場合、バックアップコーダを使用する（１２４）
請求項７記載の装置。
【請求項１０】
前記マシンは画像の再構築を行う
請求項１記載の画像を処理する装置。
【発明の詳細な説明】
【技術分野】
【０００１】
［背景］
データ圧縮は、大きな画像ファイルの格納コストを削減するために利用される。
また、データ圧縮は、大きな画像ファイルの伝送時間を短縮するためにも利用される。
【背景技術】
【０００２】
特定の技法は特定の種類の画像の圧縮には適するが、他の種類の画像には適さない。
パレット化等の圧縮技法は、色数のごく少ない画像の可逆的圧縮に対しては良好なビットレートを提供する。
しかし、パレット化は、色数の多い画像に対しては実用的ではない。
場合によっては、パレット化は、パレット記述のコストによりファイルサイズを増大させてしまうことがある。
これとは対照的に、ＪＰＥＧ－ＬＳ等の可逆的圧縮技法は、写真および他の自然の画像に対して良好なビットレートを提供する。
【特許文献１】米国特許出願公開第２００２／１７６６３１号
【非特許文献１】WEINBERGER M V ET AL: "THE LOCO-I LOSSLESS IMAGE COMPRESSION ALGORITHM: PRINCIPLES AND STANDARDIZATION INTO JPEG-LS" IEEE TRANSACTIONS ON IMAGE PROCESSING IEEE INC. NEW YORK, US, vol. 9, no. 8, August 2000 (2000-08), pages 1309-1324, XP000954683 ISSN: 1057-7149 title abstract section "II.A. Model Cost a

nd Prior Knowledge" section "II.B. Application to LOCO-1" section "III.A. Prediction" section "III.B.1) Parameterization" section "III .B.2) Context Determination" appendix "Other Features of the Standard: Color Images" appendix "Other Features of the Standard: Palletized Images" figure l

【非特許文献２】WEINBERGER M J ET AL: "From LOGO-I to the JPEG-LS standard" IMAGE PROCESSING, 1999. ICIP 99. PROCEEDINGS. 1999 INTERNATIONAL CONFERENCE ON KOBE, JAPAN 24-28 OCT. 1999, PISCATAWAY, NJ, USA,IEEE, US, 24 October 1999 (1999-l0-24), pages 68-72, XP010368687 ISBN: 0-7803-5467-2 abstract section "2 Short description of JPEG-LS (for single-component images)" section "3 Additional Features of the standard" figure l

【発明の開示】

【発明が解決しようとする課題】

【０００３】

しかし、ＪＰＥＧ－ＬＳの潜在能力は、複合文書（テキストおよび自然の画像の両方を含む）の場合のように画像の部分部分にごく少数の異なる色が含まれる場合、十分には発揮されない。

【課題を解決するための手段】

【０００４】

本発明の一態様によれば、異なる色度コンテキストに対応する複数のカラーキャッシュにアクセスし、処理中のピクセル値のカラーキャッシュが選択され、選択されたカラーキャッシュ内の情報が、処理中のピクセルの値を予測するために使用される。

【０００５】

本発明の第２の態様によれば、処理中のピクセルの予測値と実値の間の誤差が求められ、この誤差は、制約を満たす場合には符号化され、制約を満たさない場合には、デフォルトとしてバックアップ技法が採られ、処理中のピクセルが符号化される。

【０００６】

本発明の第３の態様によれば、すでに処理済みのピクセル値のキャッシュが生成され、このキャッシュが、処理中のピクセルの値を予測するために使用され、処理中のピクセルの実値がキャッシュ内に見つからない場合には、ピクセル値がキャッシュに加えられる。

本発明の他の態様および利点が、例として本発明の原理を示す、添付図面と併せて行われる以下の詳細な説明から明らかになろう。

【発明を実施するための最良の形態】

【０００７】

［詳細な説明］

例示を目的として図面に示すように、本発明は、デジタル画像の可逆的圧縮を行うエンコーダ、およびデジタル画像を再構築するデコーダを備えたシステムにおいて具現される。

圧縮中、カラー画像がトラバースされ、ピクセル値が符号化される。

因果的近傍（causal neighborhood）におけるピクセル値が使用されて、符号化中のピクセルの色度コンテキストが求められる。

異なる色度コンテキストに対してカラーキャッシュが生成され、更新される。

各カラーキャッシュは、その色度コンテキストですでに処理済みのピクセル値を含む。

同様のピクセル値が、同じ色度コンテキストにおいて再び出現する見込みが高いと推測される。

カラーキャッシュは、符号化中のピクセルの値を予測するために使用される。

予測ピクセル値と実際のピクセル値の間の誤差のみが符号化され、ビットストリームに書き込まれる。

【０００８】

システムは、フルカラーベクトルと併せて働く。

複数の色成分が、別個にではなく同時に処理される。

【０００９】
システムは、異なる種類の画像および画像中の領域に適応する。
これにより、システムは、多くの色を含む文書、ごく少数の色を含む文書、および複合文書に対して有効になる。
【００１０】
デジタルカラー画像を圧縮する例示的な方法を示す図１を参照する。
圧縮は、画像にアクセスすることから始まる（１１０）。
画像は、ＲＧＢ等の色空間で表される。
ＲＧＢ色空間を表す典型的な２４ビットワードでは、８ビットが赤成分を表し、８ビットが緑成分を表し、８ビットが青成分を表す。
【００１１】
初期化が行われる（１１２）。
初期化の一環として、プライマリコーダが初期化される。
プライマリコーダはカラーキャッシュを使用する。
したがって、初期化は、異なる色度コンテキスト毎にカラーキャッシュを定義することによって行われる。
カラーキャッシュには初期ピクセル値が入っていてもよく、または空のままであってもよい。
【００１２】
また、初期化の一環として、バックアップコーダが初期化される。
バックアップコーダは、ＪＰＥＧ－ＬＳ等、従来の可逆的技法を使用することができる。
【００１３】
プライマリコーダおよびバックアップコーダが実行される（１１４）。
画像中のピクセルの符号化は、所与の順序で行われる。
たとえば、符号化の順序はラスタ走査を辿ることができる。
【００１４】
プライマリコーダは、処理中のピクセルの値を、（たとえば、因果的近傍ピクセルからの）すでに処理済みのピクセル値から予測する（１１６）。
すでに処理済みのピクセル値を使用して、色度コンテキスト、ひいてはカラーキャッシュが選択される（１１８）。
予測ピクセル値の輝度が予測され（１１９）、予測輝度を使用して、選択されたカラーキャッシュ内のピクセル値にインデックスが付与される（１２０）。
インデックス付与されたピクセル値は、符号化中のピクセル値の予測値として使用することができる。
したがって、予測値は、すでに出現したピクセル値であることができる。
別法として、予測値は、すでに出現した値から合成したものであってもよい。
【００１５】
予測ピクセル値と実際のピクセル値の間の誤差が計算される（１２２）。
この誤差は、予測値と実値の間の差分であってもよく、またはそれぞれのインデックスの差分であってもよい。
誤差は、許容可能な範囲内にある場合、符号化されてビットストリームに書き込まれる（１２４）。
いずれの符号化技法（たとえば、算術符号化）を使用してもよい。
ワード長を短縮する符号化技法について、図３と併せて以下説明する。
【００１６】
誤差が許容可能な範囲内にない（たとえば、誤差が大きすぎる、実際のピクセル値がキャッシュ内にないため、インデックスベースの誤差を計算することができない）場合、バックアップコーダが使用されて、処理中のピクセルの値が符号化される（１２４）。
バックアップコーダの出力がビットストリームに書き込まれる。

【００１７】
選択されたカラーキャッシュが更新される（１２６）。
処理中のピクセルの値は、同じ色度コンテキスト内で再び出現する見込みが高いと推測される。
したがって、処理中のピクセルの値は、選択されたカラーキャッシュに加えられる。
【００１８】
処理は、画像全体が圧縮されるまで続けられる（１２８）。
追加の各ピクセルが処理される都度、プライマリコーダの出力か、またはバックアップコーダの出力がビットストリームに書き込まれる。
さらに、追加の各ピクセル値が、選択されたカラーキャッシュに加えられる。
したがって、各カラーキャッシュにはすでに出現したピクセル値が入れられる。
【００１９】
選択されたカラーキャッシュは、再び出現する見込みの低いピクセル値を除去することによってさらに更新することができる。
見込みの低いピクセル値を除去して、カラーキャッシュが大きくなりすぎないようにすることができる。
ピクセル値は、図１に示すように、カラーキャッシュが選択される都度、除去することができる。
別の例として、ピクセル値は、画像のＭ行の処理が完了する都度、除去することができる。
カラーキャッシュの更新の例を図２と併せて以下提供する。
【００２０】
圧縮の開始時には、初期カラーキャッシュが空である、または意味のない値が入っているため、バックアップコーダが使用される見込みが高い。
カラーキャッシュに実際のピクセル値が入ると、予測がより正確になり、誤差がより小さくなる。
最終的には、プライマリコーダが使用される頻度が高くなり、バックアップコーダは必要な場合だけ使用されることになる。
【００２１】
バックアップコーダの出力がビットストリームに書き込まれない場合であっても依然として、バックアップコーダはアクティブなままである。
バックアップコーダは、必要な場合に符号化を行えるようにする統計を保持する。
【００２２】
フルカラーベクトルがキャッシュされ、インデックスが付与される。
これにより、複数の色成分を別個にではなく同時に処理することが可能になる。
【００２３】
これより、例示的なエンコーダ２１０を示す図２を参照する。
エンコーダについて、例示的なピクセルブロックと併せて説明する。
例示的なブロックは符号化中のピクセル（カレントピクセル）を含み、これは文字「Ｘ」で表される。
この例示的なブロックは、すでに処理済みのピクセル、すなわち北西（「ＮＷ」）ピクセル、北（「Ｎ」）ピクセル、北東（「ＮＥ」）ピクセル、および西「（Ｗ）」ピクセルも含む。
走査中、東ピクセルがカレントピクセルに先立って処理される場合、西ピクセルに代えて東ピクセルが使用される。
特定の場合では、東ピクセルまたは西ピクセルは利用可能ではない（たとえば、カレントピクセルが、処理する行の最初のピクセルである）。
【００２４】
エンコーダ２１０は、カレントピクセル（Ｘ）の輝度を予測する輝度予測器２１２を備える。

たとえば、カレントピクセル（Ｘ）の輝度（（?）Ｌ$_{x}$）は、以下のＭＥＤ予測器：

【００２５】

【数１】

$$\hat{L}_x = \begin{cases} \min(L_W, L_N) & if\, L_{NW} \geq \max(L_W, L_N) \\ \max(L_W, L_N) & if\, L_{NW} \leq \min(L_W, L_N) \\ L_W + L_N - L_{NW} & otherwise \end{cases}$$

【００２６】

によって予測することができる。

ここで、Ｌ$_{NW}$、Ｌ$_{N}$、Ｌ$_{NE}$、およびＬ$_{W}$は、ＮＷ、Ｎ、ＮＥ、およびＷの各ピクセルの輝度を表す。

ピクセルのＲＧＢ記述が利用可能であるため、ＭＥＤ予測器を３つの成分のそれぞれに適用し、次いで、対応する輝度値が計算される。

したがって、各色成分の強度が推測される。

次いで、バイアス相殺を適用することができる。バイアス相殺は、ＪＰＥＧ－ＬＳで行われるものと同じであることができる。

【００２７】

エンコーダ２１０は、カラーキャッシュを保持するメモリ２１４、およびカレントピクセルのカラーキャッシュを選択する色度コンテキストセレクタ２１６を備える。

セレクタ２１６は、近傍ピクセルのＲＧＢ値をＹＵＶ値に変換し、ＵＶ値を使用して色度コンテキストを求め、カレントピクセルのカラーキャッシュを選択することができる。

色度コンテキストは、ＵＶプレーンを色領域に分割して、因果的近傍における各ピクセル値を領域にマッピングすることによって定義することができる。

領域への分割は、因果的近傍におけるすべての割り振りについて同じである必要はない。

使用する領域の数が多いほど、カレントピクセルへの色度コンテキストの割り当てを判断する際に特定の近傍ピクセルに大きなウェイトを与えることができる。

【００２８】

各カラーキャッシュに記憶されるピクセル値は、輝度値の順番に記憶して、選択されたカラーキャッシュからのアクセスを加速化することができる。

予測値に最も近い輝度値を有するピクセル値が予測色として選ばれる。

これより、カラーキャッシュを定義し選択する４つの異なる具現化について説明する。

【００２９】

第１の具現化では、ＵＶプレーンはｎ個の互いに素な領域（たとえば、４象限、８分限）に分割される。

各領域が色度コンテキストを定義する。

Ｗ、ＮＷ、Ｎ、およびＮＥピクセルをこれらｎ個の領域に分類することにより、選択に利用可能なｎ^{4}個の色度コンテキストが定義される。

四次元ベクトルが、１対１ベースで、範囲［０，ｎ^{4}－１］のインデックスにマッピングされる。

このインデックスは、色度コンテキストに関連するカラーキャッシュを識別する。

より一般的な手順では、ＵＶプレーンが分割される領域の数および形状は、Ｗ、ＮＷ、Ｎ、およびＮＥピクセルのそれぞれで異なり得る。

【００３０】

第２の具現化では、色度コンテキストは、カラーキャッシュと色度コンテキストの間に固定の対応性がないこと以外は第１の具現化と同様に定義される。

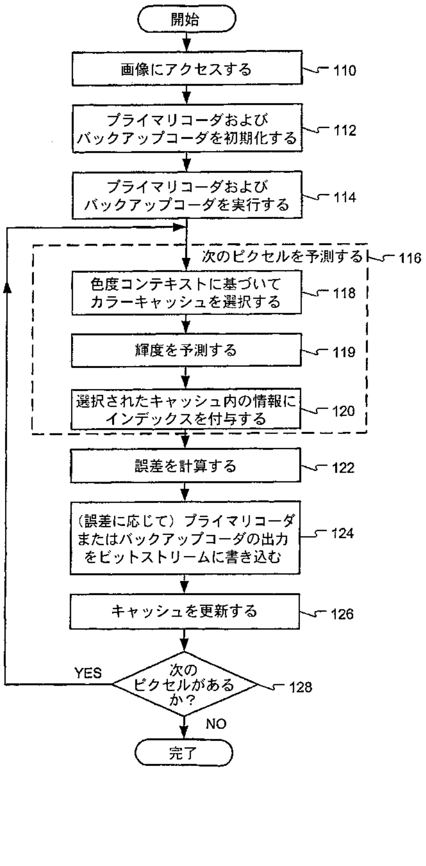
むしろ、カラーキャッシュには、ＵＶプレーンの互いに素の領域が関連付けられ、それぞれの数は色度コンテキストの数と異なり得る。
最大推定尤度を有するカラーキャッシュが選択される。
尤度は、このキャッシュに関連するＵＶ領域からの色がカレント色度コンテキストにおいて処理された回数を、この色度コンテキストの出現数で除算した比として推定することができる。
コンテキストセレクタ２１６はこの統計を追跡し、カラーキャッシュに関連するＵＶ領域の出現回数を求める。
【００３１】
第３の具現化では、ＵＶプレーンは、必要なカラーキャッシュと同数の領域に分割される。
１つのカラーキャッシュがただ１つの領域に割り当てられる。
Ｗ、ＮＷ、Ｎ、およびＮＥピクセルはチェックされて、属するカラーキャッシュが識別され、多数を得たカラーキャッシュがカレントピクセルに選択される。
同数の場合、最も近いピクセル（たとえば、西ピクセルまたは北ピクセル）のカラーキャッシュが選択される。
【００３２】
第４の具現化では、Ｗピクセルに対応する領域とＮＷピクセルに対応する領域が比較される。
これら領域が同一の場合、Ｎピクセルの色度コンテキストがカレントピクセルに割り当てられる。
その他の場合は、Ｗピクセルの色度コンテキストがカレントピクセルに割り当てられる。
この第４の具現化は、画像内のエッジの検出を目的として、ＭＥＤ予測器によって行われる。
Ｗピクセルおよびピクセルが同じ領域に属する場合、垂直エッジがカレントピクセルのすぐ左隣にあるものと想定される。
したがって、カレントピクセルは、Ｎピクセルと同じ色度コンテキストを有する見込みが高い。
同様に、ＮピクセルおよびＮＷピクセルが同じコンテキストを有する場合、水平エッジがカレントピクセルの真上にあるものと想定され、これにより、カレントピクセルは、Ｗピクセルと同じ色度コンテキストを有するものと想定される。
これら条件のいずれも満たされない場合、因果的近傍が平滑であると想定され、これにより、最も近いピクセルの色度コンテキストが割り当てられる。
【００３３】
エンコーダ２１０は、カラーキャッシュを更新するブロック２１８を備える。
再び出現する見込みが最も低いピクセル値は、画像のＭ行の処理が完了する都度、カラーキャッシュから削除することができる。
削除するピクセル値の数は、カラーキャッシュのサイズに依存し得る。
上限を固定して、カラーキャッシュのサイズが大きくなりすぎないようにすることができる。
このように、小さなカラーキャッシュの成長を許す。
削除するピクセル値を識別する基準は、ピクセル値がどのくらい最近に使用されたかに基づくことができる。
タイムスタンプを使用して、選択された色度コンテキストに相対した新しさを判断することができる。
【００３４】
エンコーダ２１０は、カレントピクセルの予測ピクセル値$Ｘ_p$と実値の間の誤差ベクトルを求めるブロック２２０、誤差ベクトルを符号化するプライマリ誤差エンコーダ２２２、およびバックアップとしてのＪＰＥＧ－ＬＳコーダ２２４を備える。

算術符号を使用して、誤差を符号化することができる。
符号化方策の一例は、各色成分が６つの値｛０，±１，±２，エスケープ｝のうちの１つをとることができる三次元ベクトルアルファベットを定義することであり、これにより、３Ｄ色空間の場合に２１６（６^{3}）個のアルファベット文字になる。
誤差ベクトルは、このアルファベットを使用して算術的に符号化することができ、符号化された値はビットストリームに加えられる。
誤差を符号化する符号化コンテキストは、色度コンテキストと異なるだろう。
【００３５】
成分のいずれかの絶対値誤差が限度（たとえば、２）よりも大きい場合には、誤差符号には代わりにエスケープが割り当てられ、エスケープに対応する成分（複数可）が、ＪＰＥＧ－ＬＳコーダ２２４によって符号化され、ビットストリームに加えられる。
このため、ＪＰＥＧ－ＬＳコーダ２２４は、ＪＰＥＧ－ＬＳ符号化に必要な統計を追跡すべきである。
ＪＰＥＧ－ＬＳコーダ２２４は、プライマリ誤差エンコーダ２２２と並列に実行することができ、またＪＰＥＧ－ＬＳコーダ２２４からの符号化ピクセル値は、必要な場合のみビットストリームに加えられる。
誤差ベクトルのアルファベットサイズを増大させることにより、ＪＰＥＧ－ＬＳコーダ２２４が使用される確率が低減するが、これは算術符号化のビットレートを増大させる。
【００３６】
エンコーダ２１０は、予測ピクセル値と実際のピクセル値の間の誤差ベクトルの符号化に限定されない。
ブロック２２０は、誤差ベクトルの符号化に限定されるのではなく、選択されたカラーキャッシュの予測インデックス（輝度予測器に基づく）と実際のピクセル値のインデックスの間の誤差も判断する。
プライマリ誤差エンコーダ２２２は、予測されたインデックスと実際のインデックスの間の誤差を符号化する。
実際のピクセル値が、選択されたカラーキャッシュ内で見つからない場合、インデックスの誤差はビットストリームに書き込まれず、バックアップコーダが代わりに使用される。
【００３７】
これら２つの符号化方策を組み合わせることができる。
たとえば、すべての誤差成分にエスケープが割り当てられる場合、インデックスを符号化してビットストリームに書き込むことができる。
【００３８】
これより、ビットストリームから画像を再構築する方法を示す図３を参照する。
再構築は、ビットストリームにアクセスすること（３１０）により開始される。
プライマリデコーダおよびバックアップデコーダが初期化され、実行される（３１２）。
プライマリデコーダは、色度コンテキストに基づいてカラーキャッシュを選択し（３１６）、輝度値を予測し（３１７）、予測された輝度値を使用して、選択されたキャッシュ内の情報にインデックスを付与する（３１８）ことにより、エンコーダとまったく同じように次のピクセルの値を予測する（３１４）。
ビットストリーム内の誤差が読み取られて復号化される（３２０）。
バックアップコーダが復号化ピクセル値を生成するか、または復号化ピクセル値が、復号化誤差を予測ピクセル値に加えることによって生成される（３２２）。
復号化ピクセル値は、選択されたカラーキャッシュに加えられ、再構築された画像に加えられ（３２４）、次のピクセルが再構築される（３２６）。
【００３９】
エンコーダおよびデコーダのハードウェア実装は、任意の特定のタイプに限定されない。

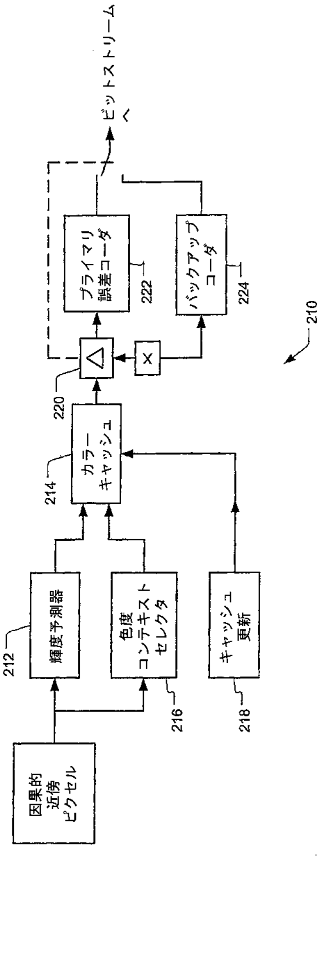
たとえば、図４に示すマシンを符号化および／または復号化に使用することができる。
マシン４１０は、プロセッサ４１２およびプログラム４１６が符号化されたメモリ４１４を備える。
プログラム４１６は、実行されると、プロセッサ４１２に、カラー画像にアクセスして本発明の一実施形態による圧縮を行うか、またはビットストリームを読み取って本発明の一実施形態によりデジタル画像を再構築するように命令する。
プロセッサ４１２は、ビットストリームか、または再構築された画像を表す信号４１８を出力する。
信号４１８は、マシン４１０に記憶することも、他のどこかに伝送することも、または記憶媒体（たとえば、光ディスク）４２０に記録することも可能である。
【００４０】
カラーキャッシュを選択し、輝度を推測するための情報は、カレントピクセルの因果的近傍から得られる。
上述した一例では、因果的近傍は、ＮW、Ｎ、ＮＥ、およびWピクセルからなる。
しかし、因果的近傍はこのように限定されない。
【００４１】
カラーキャッシュにおけるピクセル値の順序は「輝度」の順序として説明したが、ピクセル値は他の方法で並べてもよい。
たとえば、ピクセル値を、処理される順序で並べてもよい。
【００４２】
圧縮および再構築についてＲＧＢ色空間と併せて説明したが、圧縮および再構築はこのように限定されない。
他の色空間も使用することが可能である。
このような色空間の例としては、ＣＭＹＫ色空間が挙げられるが、これに限定されない。
【００４３】
圧縮および再構築について可逆的なものとして説明したが、圧縮および再構築はこのように限定されない。
たとえば、圧縮は、厳密な誤差ではなく、量子化誤差を使用して行う（ＪＰＥＧ－ＬＳの「準可逆」モードで実施されるように）ことも可能である。
【００４４】
本発明は、上で説明し図示した特定の実施形態に限定されない。
そうではなく、本発明は添付の特許請求の範囲に従って解釈される。
【図面の簡単な説明】
【００４５】
【図１】本発明の一実施形態による画像を圧縮する方法の図である。
【図２】本発明の一実施形態による画像を圧縮するエンコーダの図である。
【図３】本発明の一実施形態による画像を再構築する方法の図である。
【図４】図１および図３の方法のハードウェア実装の図である。
【符号の説明】
【００４６】
２１２・・・輝度予測器，
２１４・・・カラーキャッシュ，
２１６・・・色度コンテキストセレクタ，
２１８・・・キャッシュ更新，
２２２・・・プライマリ誤差コーダ，
２２４・・・バックアップコーダ，
４１０・・・マシン，
４１２・・・プロセッサ，
４１４・・・メモリ，

４１６・・・プログラム，
４２０・・・記憶媒体，

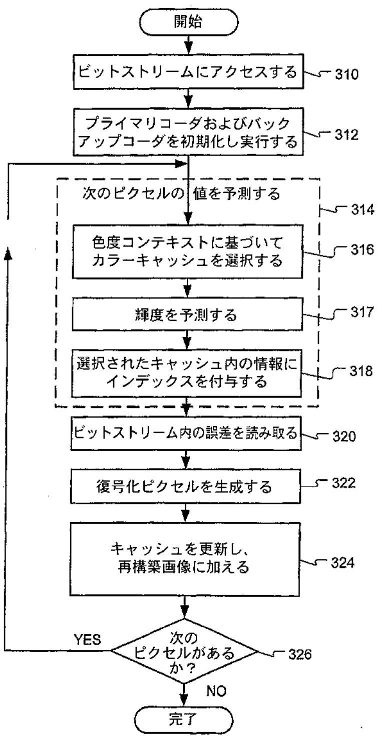
【図１】



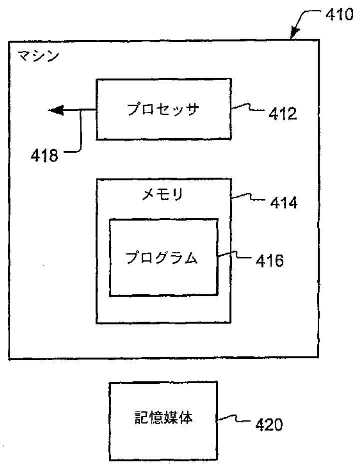
【図２】



【図３】


開始
ビットストリームにアクセスする
310
プライマリコーダおよびバックアップコーダを初期化し実行する
312
次のピクセルの値を予測する
314
色度コンテキストに基づいてカラーキャッシュを選択する
316
輝度を予測する
317
選択されたキャッシュ内の情報にインデックスを付与する
318
ビットストリーム内の誤差を読み取る
320
復号化ピクセルを生成する
322
キャッシュを更新し、再構築画像に加える
324
次のピクセルがあるか？
326
YES
NO
完了

【図４】


410
マシン
プロセッサ
412
418
メモリ
414
プログラム
416
記憶媒体
420

フロントページの続き

(72)発明者　アドリアーナ・ピアッツァ
ウルグアイ国モンテビデオ　ピーシー１１４００　サンティアゴデアンカー１５０１

審査官　松永　隆志

(56)参考文献　特開平０７－０４６４０９（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
H04N　1/41
H04N　11/04