

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4440051号
(P4440051)

(45) 発行日 平成22年3月24日(2010.3.24)

(24) 登録日 平成22年1月15日(2010.1.15)

(51) Int.Cl.	F 1
HO4N 1/413 (2006.01)	HO4N 1/413 D
GO6T 9/00 (2006.01)	GO6T 9/00
HO4N 1/41 (2006.01)	HO4N 1/41 B

請求項の数 16 (全 35 頁)

(21) 出願番号 特願2004-261570 (P2004-261570)
 (22) 出願日 平成16年9月8日 (2004.9.8)
 (65) 公開番号 特開2006-80794 (P2006-80794A)
 (43) 公開日 平成18年3月23日 (2006.3.23)
 審査請求日 平成19年8月28日 (2007.8.28)

(73) 特許権者 000001007
 キヤノン株式会社
 東京都大田区下丸子3丁目30番2号
 (74) 代理人 100076428
 弁理士 大塚 康徳
 (74) 代理人 100112508
 弁理士 高柳 司郎
 (74) 代理人 100115071
 弁理士 大塚 康弘
 (74) 代理人 100116894
 弁理士 木村 秀二
 (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】 画像符号化装置及び方法、並びに、コンピュータプログラム及びコンピュータ可読記憶媒体

(57) 【特許請求の範囲】

【請求項 1】

画像データを入力し、当該画像データを符号化する画像符号化装置であって、
 第1のパラメータに従って、入力する画像データを所定画素ブロック単位に非可逆符号化し、非可逆符号化データを生成する第1の符号化手段と、

入力する画像データを前記所定画素ブロック単位に可逆符号化し、可逆符号化データを生成する第2の符号化手段と、

前記第1、第2の符号化手段で生成された、それぞれの符号化データのうち、符号長の短い符号化データを所定の第1のメモリに格納する第1の格納手段と、

前記第2の符号化手段で生成された可逆符号化データを所定の第2のメモリに格納する第2の格納手段と、

前記第2のメモリ内の可逆符号化データを復号し、第2のパラメータに従って再符号化して非可逆符号化データを生成すると共に、該再符号化後により得られた非可逆符号化データと該再符号化前の可逆符号化データのうち、符号長の短い符号化データを前記第1のメモリに格納する再符号化手段と、

前記第1のメモリに格納された符号化データ量を監視し、所定データ量を越えたか否かを判断する監視手段と、

該監視手段で前記符号化データ量が所定データ量を超えたと判断した場合、(a)前記第1のメモリ内の符号化データを破棄し、(b)前記第1の符号化手段に設定されていた第1のパラメータを、圧縮率の高いパラメータに更新して画像データの符号化を継続させ

10

20

、(c) 更新後の第1のパラメータと同じパラメータを第2パラメータとして前記再符号化手段に設定して、前記符号化データ量が所定データ量を超える以前に得られた可逆符号化データを再符号化させるパラメータ更新手段と、

1ページ分の符号化処理が完了した場合、前記第1のメモリに格納された各画素ブロック毎の符号化データ中の、周辺の符号化データの種類と異なる種類の孤立符号化データを、前記周辺の符号化データの種類に補正する補正手段と

を備えることを特徴とする画像符号化装置。

【請求項2】

前記第1の符号化手段による符号化にはJPEGを用い、前記第2の符号化手段の符号化にはJPEG-LSを用いることを特徴とする請求項1に記載の画像符号化装置。 10

【請求項3】

前記補正手段は、前記孤立符号化データが可逆符号化データである場合に、非可逆符号化データに補正することを特徴とする請求項1に記載の画像符号化装置。

【請求項4】

前記補正手段は、前記孤立符号化データが可逆符号化データであって、当該符号化データ量が所定閾値以上の場合に非可逆符号化データに補正することを特徴とする請求項3に記載の画像符号化装置。

【請求項5】

前記補正手段は、前記孤立符号化データが非可逆符号化データである場合には、前記第2のメモリの該当する位置の画素ブロックの可逆符号化データを補正結果として出力することを特徴とする請求項3又は4に記載の画像符号化装置。 20

【請求項6】

画像データを入力し、当該画像データを圧縮符号化し出力する画像符号化方法であって、
、
第1のパラメータに従って、入力する画像データを所定画素ブロック単位に非可逆符号化し、非可逆符号化データを生成する第1の符号化工程と、

入力する画像データを前記所定画素ブロック単位に可逆符号化し、可逆符号化データを生成する第2の符号化工程と、

前記第1、第2の符号化工程で生成された、それぞれの符号化データのうち、符号長の短い符号化データを所定の第1のメモリに格納する第1の格納工程と。 30

前記第2の符号化工程で生成された可逆符号化データを所定の第2のメモリに格納する第2の格納工程と、

前記第2のメモリ内の可逆符号化データを復号し、第2のパラメータに従って再符号化して非可逆符号化データを生成すると共に、該再符号化後により得られた非可逆符号化データと該再符号化前の可逆符号化データのうち、符号長の短い符号化データを前記第1のメモリに格納する再符号化工程と、

前記第1のメモリに格納された符号化データ量を監視し、所定データ量を越えたか否かを判断する監視工程と、

該監視工程で前記符号化データ量が所定データ量を超えたと判断した場合、(a)前記第1のメモリ内の符号化データを破棄し、(b)前記第1の符号化工程に設定されていた第1のパラメータを、圧縮率の高いパラメータに更新して画像データの符号化を継続させ、(c)更新後の第1のパラメータと同じパラメータを第2パラメータとして前記再符号化工程に設定して、前記符号化データ量が所定データ量を超える以前に得られた可逆符号化データを再符号化させるパラメータ更新工程と、

1ページ分の符号化処理が完了した場合、前記第1のメモリに格納された各画素ブロック毎の符号化データ中の、周辺の符号化データの種類と異なる種類の孤立符号化データを、前記周辺の符号化データの種類に補正する補正工程と

を備えることを特徴とする画像符号化方法。

【請求項7】

コンピュータが読み込み実行することで、前記コンピュータを、画像データを入力し、

50

当該画像データを圧縮符号化し出力する画像符号化装置として機能させるコンピュータプログラムであって、

第1のパラメータに従って、入力する画像データを所定画素ブロック単位に非可逆符号化し、非可逆符号化データを生成する第1の符号化手段と、

入力する画像データを前記所定画素ブロック単位に可逆符号化し、可逆符号化データを生成する第2の符号化手段と、

前記第1、第2の符号化手段で生成された、それぞれの符号化データのうち、符号長の短い符号化データを所定の第1のメモリに格納する第1の格納手段と、

前記第2の符号化手段で生成された可逆符号化データを所定の第2のメモリに格納する第2の格納手段と、

前記第2のメモリ内の可逆符号化データを復号し、第2のパラメータに従って再符号化して非可逆符号化データを生成すると共に、該再符号化後により得られた非可逆符号化データと該再符号化前の可逆符号化データのうち、符号長の短い符号化データを前記第1のメモリに格納する再符号化手段と、

前記第1のメモリに格納された符号化データ量を監視し、所定データ量を越えたか否かを判断する監視手段と、

該監視手段で前記符号化データ量が所定データ量を超えたと判断した場合、(a)前記第1のメモリ内の符号化データを破棄し、(b)前記第1の符号化手段に設定されていた第1のパラメータを、圧縮率の高いパラメータに更新して画像データの符号化を継続させ、(c)更新後の第1のパラメータと同じパラメータを第2パラメータとして前記再符号化手段に設定して、前記符号化データ量が所定データ量を超える以前に得られた可逆符号化データを再符号化させるパラメータ更新手段と、

1ページ分の符号化処理が完了した場合、前記第1のメモリに格納された各画素ブロック毎の符号化データ中の、周辺の符号化データの種類と異なる種類の孤立符号化データを、前記周辺の符号化データの種類に補正する補正手段

として機能させることを特徴とするコンピュータプログラム。

【請求項8】

請求項7に記載のコンピュータプログラムを格納したことを特徴とするコンピュータ可読記憶媒体。

【請求項9】

画像データを入力し、当該画像データを符号化する画像符号化装置であって、

第1のパラメータに従って、入力する画像データを所定画素ブロック単位に非可逆符号化し、非可逆符号化データを生成する第1の符号化手段と、

前記第1のパラメータよりも高い圧縮率となる第2のパラメータに従って、入力する画像データを前記所定画素ブロック単位に非可逆符号化し、非可逆符号化データを生成する第2の符号化手段と、

入力する画像データを前記所定画素ブロック単位に可逆符号化し、可逆符号化データを生成する第3の符号化手段と、

前記第1、第3の符号化手段で生成された、それぞれの符号化データのうち、符号長の短い符号化データを所定の第1のメモリに格納する第1の格納手段と、

前記第2、第3の符号化手段で生成された、それぞれの符号化データのうち、符号長の短い符号化データを所定の第2のメモリに格納する第2の格納手段と、

前記第3の符号化手段で生成された可逆符号化データを所定の第3のメモリに格納する第3の格納手段と、

前記第3のメモリ内の可逆符号化データを復号し、第3のパラメータに従って再符号化して非可逆符号化データを生成すると共に、該再符号化により得られた非可逆符号化データと該再符号化前の可逆符号化データのうち、符号長の短い符号化データを前記第2のメモリに格納する再符号化手段と、

前記第1のメモリに格納された符号化データ量を監視し、所定データ量を超えたか否かを判断する監視手段と、

10

20

30

40

50

該監視手段で前記符号化データ量が所定データ量を超えたと判断した場合、(a)前記第1のメモリ内の符号化データを破棄し、前記第2のメモリ内の符号化データを前記第1のメモリに転送し、(b)前記第1の符号化手段に設定されていた第1のパラメータを、前記第2の符号化手段に設定されていた第2のパラメータで更新して画像データの符号化を継続させ、(c)前記第2のメモリ内の符号化データを破棄し、前記第2の符号化手段に設定されていた第2のパラメータを圧縮率の高いパラメータに更新して、符号化を継続させ、(d)更新後の第2のパラメータと同じパラメータを第3のパラメータとして前記再符号化手段に設定して、前記符号化データ量が所定データ量を超える以前に得られた可逆符号化データを再符号化させるパラメータ更新手段と、

1ページ分の符号化処理が完了した場合、前記第1のメモリに格納された各画素ブロック毎の符号化データ中の、周辺の符号化データの種類と異なる種類の孤立符号化データを、前記周辺の符号化データの種類に補正する補正手段と

を備えることを特徴とする画像符号化装置。

【請求項10】

前記第1、第2の符号化手段による符号化にはJPEGを用い、前記第3の符号化手段による符号化にはJPEG-LSを用いることを特徴とする請求項9に記載の画像符号化装置。

【請求項11】

前記補正手段は、孤立符号化データが可逆符号化データである場合に、非可逆符号化データに補正することを特徴とする請求項9に記載の画像符号化装置。

【請求項12】

前記補正手段は、孤立符号化データが可逆符号化データであって、当該符号化データ量が所定閾値以上の場合に非可逆符号化データに補正することを特徴とする請求項11に記載の画像符号化装置。

【請求項13】

前記補正手段は、孤立符号化データが非可逆符号化データである場合には、前記第2のメモリの該当する位置の画素ブロックの可逆符号化データを補正結果として出力することを特徴とする請求項11又は12に記載の画像符号化装置。

【請求項14】

画像データを入力し、当該画像データを圧縮符号化し出力する画像符号化方法であって、

第1のパラメータに従って、入力する画像データを所定画素ブロック単位に非可逆符号化し、非可逆符号化データを生成する第1の符号化工程と、

前記第1のパラメータよりも高い圧縮率となる第2のパラメータに従って、入力する画像データを前記所定画素ブロック単位に非可逆符号化し、非可逆符号化データを生成する第2の符号化工程と、

入力する画像データを前記所定画素ブロック単位に可逆符号化し、可逆符号化データを生成する第3の符号化工程と、

前記第1、第3の符号化工程で生成された、それぞれの符号化データのうち、符号長の短い符号化データを所定の第1のメモリに格納する第1の格納工程と、

前記第2、第3の符号化工程で生成された、それぞれの符号化データのうち、符号長の短い符号化データを所定の第2のメモリに格納する第2の格納工程と、

前記第3の符号化工程で生成された可逆符号化データを所定の第3のメモリに格納する第3の格納工程と、

前記第3のメモリ内の可逆符号化データを復号し、第3のパラメータに従って再符号化して非可逆符号化データを生成すると共に、該再符号化により得られた非可逆符号化データと該再符号化前の可逆符号化データのうち、符号長の短い符号化データを前記第2のメモリに格納する再符号化工程と、

前記第1のメモリに格納された符号化データ量を監視し、所定データ量を超えたか否かを判断する監視工程と、

10

20

30

40

50

該監視工程で前記符号化データ量が所定データ量を超えたと判断した場合、(a)前記第1のメモリ内の符号化データを破棄し、前記第2のメモリ内の符号化データを前記第1のメモリに転送し、(b)前記第1の符号化工程に設定されていた第1のパラメータを、前記第2の符号化工程に設定されていた第2のパラメータで更新して画像データの符号化を継続させ、(c)前記第2のメモリ内の符号化データを破棄し、前記第2の符号化工程に設定されていた第2のパラメータを圧縮率の高いパラメータに更新して、符号化を継続させ、(d)更新後の第2のパラメータと同じパラメータを第3のパラメータとして前記再符号化工程に設定して、前記符号化データ量が所定データ量を超える以前に得られた可逆符号化データを再符号化させるパラメータ更新工程と、

1ページ分の符号化処理が完了した場合、前記第1のメモリに格納された各画素ブロック毎の符号化データ中の、周辺の符号化データの種類と異なる種類の孤立符号化データを、前記周辺の符号化データの種類に補正する補正工程と
を備えることを特徴とする画像符号化方法。

【請求項15】

コンピュータが読み込み実行することで、前記コンピュータを、画像データを入力し、当該画像データを圧縮符号化し出力する画像符号化装置として機能させるコンピュータプログラムであって、

第1のパラメータに従って、入力する画像データを所定画素ブロック単位に非可逆符号化し、非可逆符号化データを生成する第1の符号化手段と、

前記第1のパラメータよりも高い圧縮率となる第2のパラメータに従って、入力する画像データを前記所定画素ブロック単位に非可逆符号化し、非可逆符号化データを生成する第2の符号化手段と、

入力する画像データを前記所定画素ブロック単位に可逆符号化し、可逆符号化データを生成する第3の符号化手段と、

前記第1、第3の符号化手段で生成された、それぞれの符号化データのうち、符号長の短い符号化データを所定の第1のメモリに格納する第1の格納手段と、

前記第2、第3の符号化手段で生成された、それぞれの符号化データのうち、符号長の短い符号化データを所定の第2のメモリに格納する第2の格納手段と、

前記第3の符号化手段で生成された可逆符号化データを所定の第3のメモリに格納する第3の格納手段と、

前記第3のメモリ内の可逆符号化データを復号し、第3のパラメータに従って再符号化して非可逆符号化データを生成すると共に、該再符号化により得られた非可逆符号化データと該再符号化前の可逆符号化データのうち、符号長の短い符号化データを前記第2のメモリに格納する再符号化手段と、

前記第1のメモリに格納された符号化データ量を監視し、所定データ量を超えたか否かを判断する監視手段と、

該監視手段で前記符号化データ量が所定データ量を超えたと判断した場合、(a)前記第1のメモリ内の符号化データを破棄し、前記第2のメモリ内の符号化データを前記第1のメモリに転送し、(b)前記第1の符号化手段に設定されていた第1のパラメータを、前記第2の符号化手段に設定されていた第2のパラメータで更新して画像データの符号化を継続させ、(c)前記第2のメモリ内の符号化データを破棄し、前記第2の符号化手段に設定されていた第2のパラメータを圧縮率の高いパラメータに更新して、符号化を継続させ、(d)更新後の第2のパラメータと同じパラメータを第3のパラメータとして前記再符号化手段に設定して、前記符号化データ量が所定データ量を超える以前に得られた可逆符号化データを再符号化させるパラメータ更新手段と、

1ページ分の符号化処理が完了した場合、前記第1のメモリに格納された各画素ブロック毎の符号化データ中の、周辺の符号化データの種類と異なる種類の孤立符号化データを、前記周辺の符号化データの種類に補正する補正手段

として機能させることを特徴とするコンピュータプログラム。

【請求項16】

10

20

30

40

50

請求項 1 5 に記載のコンピュータプログラムを格納したことを特徴とするコンピュータ可読記憶媒体。

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

本発明は、画像データを符号化する技術に関するものである。

【背景技術】

【0 0 0 2】

従来、静止画像データを圧縮するには、離散コサイン変換を利用した方法や、Wavelet 变換を利用した方法が多く使われている。この種の符号化は、可変長符号化であるので、
10 符号化対象の画像毎に符号量が変化する。

【0 0 0 3】

国際標準化方式である J P E G 符号化では、1 つの画像に対して 1 組の量子化マトリクスしか定義できないので、プリスキャン無しには、1 つの画像（原稿）の符号化データを目標とする符号量に収めることは難しい。また、限られたメモリに記憶するシステムで使用する場合においては、メモリオーバーを起こす危険性がある。

【0 0 0 4】

これを防止するために、予定した符号量よりオーバーした場合は、圧縮率パラメータを変更して、同じ原稿の再読み込みを行なう方法や、予めプリスキャンによる符号量見積もりを行ない、符号量を調整するために、量子化パラメータの再設定を行なう方法などがとられていた。

【0 0 0 5】

上記のように、プリスキャン、本スキャンを行うのが一般的であるが、これでは少なくとも 2 回、同じ原稿を読取る必要があり、効率的とは言えない。

【0 0 0 6】

特に複写機等において複数枚（ページ）の原稿を A D F (Auto Document Feeder) から 1 ページずつ連続的に読み取りながら符号化する様な場合には、同じ原稿を 2 回読取ることは処理時間的にも不可能である。

【0 0 0 7】

本願出願人は、上記のようなプリスキャン、本スキャンの 2 回のスキャンを不要とし、かつ 1 つの画像全体に共通の符号化パラメータを用いた符号化を行った上で、目標となる符号化データ量に圧縮する技術を提案した（特許文献 1）。この技術は、1 度（1 ページ分）の画像入力中に、符号化データを 2 つのメモリに順次格納していき、もしその途中で、所定メモリ内の符号化データ量が所定サイズを超えた場合には、その所定メモリ内のデータを破棄するとともに、それまでの符号化パラメータから更に圧縮率を高くする新しい符号化パラメータに更新して未だ符号化されていない部分の画像データの符号化処理を継続する（ここで得られる符号化データを第 1 符号化データとする）。このとき、もう一方のメモリには、圧縮率を上げる以前の符号化により得られた符号化済みデータが格納されているので、この符号化済みデータを更新後のパラメータに従って再符号化することにより、上記新しいパラメータで最初から符号化した場合と同等の符号化データを得ることができる（この再符号化で得られる符号化データを第 2 符号化データとする）。そしてこれら第 1 符号化データと第 2 符号化データを結合することにより、1 つ（1 ページ分）の画像の全体に渡って共通の符号化パラメータ（更新後の符号化パラメータ）で符号化された符号化データ（J P E G 符号化にも準拠）を得ることができ、かつ目標となる符号化データ量にできるというものである。

【0 0 0 8】

一方で、上記特許文献 1 の様な、符号化途中の符号化データ量に応じて再符号化を実行する技術ではないが、画像中の複数の領域において、可逆と非可逆の符号化の何れか一方を選択的に適用しつつ、画像全体を一定の符号量とする技術もある（特許文献 2）。

【特許文献 1】特開 2 0 0 3 - 8 9 0 3 公報

10

20

30

40

50

【特許文献2】特開平10-224640号公報

【発明の開示】

【発明が解決しようとする課題】

【0009】

上記従来技術で用いられる特許文献1で用いられる圧縮符号化は、JPEG等の非可逆圧縮技術のみが用いられている。

【0010】

さて、上記特許文献1の符号量制御技術において、1ページの画像の符号化中に、生成される符号化データ量が所定サイズを超えた場合、そのページ全体を一律圧縮率を上げるのと等価の処理を行うことになるため、部分的に予測以上の画質劣化を招くことがあった。特に、文字線画部分を含むが画像を圧縮する場合にこの傾向が高かった。

10

【0011】

ところで、JPEG-LSという可逆圧縮技術が知られている。この技術は「JPEG」と冠しつつも、その符号化アルゴリズムは通常の非可逆のJPEGとは全く異なるものであり、自然画の圧縮率はJPEGに譲るもの、文字線画やコンピュータグラフィクスについては高い圧縮率で可逆符号化できるものとして知られている。

【0012】

これを考慮すると、1ページに写真画像、文字画像が混在しているの原稿画像を圧縮する場合、出来るだけ文字線画部分には可逆圧縮を適用するべきである。

【0013】

20

従って、このような原稿を圧縮する際に、先に示した特許文献1と同様に、一度の画像データの入力で、つまり、再入力を必要としないで、且つ、目標とするデータ量に収めるために、非可逆圧縮と可逆圧縮技術を併用する技術が望まれることになる。また、画像の一部または全部に対する非可逆符号化の適用において、その画像には共通の符号化パラメータを用いることも必要である。

【0014】

ここで、1ページの所定画素ブロック毎に可逆符号化或いは非可逆符号化を行うことを考察する。

【0015】

先ず、原画像中の所定画素ブロックサイズに対して、非可逆符号化処理、可逆符号化処理のいずれかを行うものとする。

30

【0016】

原稿画像の或る領域が文字線画であり、別な或る領域が自然画（或いは写真画像）である場合、かつ文字線画領域では可逆符号化を行い、自然画領域では非可逆符号化を行う場合においては、比較的大きな領域を単位に、これら符号化処理が切り換わるのであれば、符号化の違いによる画質の差が多少生じたとしても、夫々に適した符号化が行われているからであると理解するならば、その境目はそれほど気にならないであろう。

【0017】

ところが、所定基準に基づいて小ブロック単位で自動的に領域の種類を判別して符号化を切替えるような場合には、非可逆符号化データが孤立して生成される、或いは、可逆符号化データが孤立して生成されることも有り得る。このような状況では、小ブロック（小さい領域）を単位に、画質が変化してしまうので、視覚的ノイズとして認識される心配がある。

40

【0018】

本発明はかかる問題点に鑑み成されたものであり、非可逆符号化と可逆符号化技術を併用しつつ、一度の画像データの入力で目標とするデータ量の符号化データを生成し、且つ、画像の一部または全部に対する非可逆符号化の適用によって、その画像に共通の符号化パラメータを用いる様にし、更には、ブロックノイズの発生も抑制することを目的とする。

【課題を解決するための手段】

50

【0019】

上記課題を解決するため、本発明の画像符号化装置は以下の構成を備える。すなわち、

画像データを入力し、当該画像データを圧縮符号化し出力する画像符号化装置であって

、

画像データを入力し、当該画像データを符号化する画像符号化装置であって、

第1のパラメータに従って、入力する画像データを所定画素ブロック単位に非可逆符号化し、非可逆符号化データを生成する第1の符号化手段と、

入力する画像データを前記所定画素ブロック単位に可逆符号化し、可逆符号化データを生成する第2の符号化手段と、

前記第1、第2の符号化手段で生成された、それぞれの符号化データのうち、符号長の短い符号化データを所定の第1のメモリに格納する第1の格納手段と、

前記第2の符号化手段で生成された可逆符号化データを所定の第2のメモリに格納する第2の格納手段と、

前記第2のメモリ内の可逆符号化データを復号し、第2のパラメータに従って再符号化して非可逆符号化データを生成すると共に、該再符号化後により得られた非可逆符号化データと該再符号化前の可逆符号化データのうち、符号長の短い符号化データを前記第1のメモリに格納する再符号化手段と、

前記第1のメモリに格納された符号化データ量を監視し、所定データ量を越えたか否かを判断する監視手段と、

該監視手段で前記符号化データ量が所定データ量を超えたと判断した場合、(a)前記第1のメモリ内の符号化データを破棄し、(b)前記第1の符号化手段に設定されていた第1のパラメータを、圧縮率の高いパラメータに更新して画像データの符号化を継続させ、(c)更新後の第1のパラメータと同じパラメータを第2パラメータとして前記再符号化手段に設定して、前記符号化データ量が所定データ量を超える以前に得られた可逆符号化データを再符号化させるパラメータ更新手段と、

1ページ分の符号化処理が完了した場合、前記第1のメモリに格納された各画素ブロック毎の符号化データ中の、周辺の符号化データの種類と異なる種類の孤立符号化データを、前記周辺の符号化データの種類に補正する補正手段とを備える。

【発明の効果】

【0020】

本発明によれば、非可逆符号化と可逆符号化技術を併用しつつ、一度の画像データの入力で目標とするデータ量の符号化データを生成し、かつ画像の一部または全部に対する非可逆符号化の適用において、その画像に共通の符号化パラメータを用いることができ、更には、ブロックノイズの発生を抑制することが可能になる。

【0021】

また、非可逆符号化としてJPEG、可逆符号化としてJPEG-LSを用いることと、格別な像域判定回路を設けなくても、文字線画、自然画それぞれに対して適した符号化が選択されることになり、復号した際の画質及び圧縮率ともに優れた符号化データを生成することが可能になる。

【発明を実施するための最良の形態】

【0022】

以下、添付図面に従って本発明に係る実施形態を詳細に説明する。

【0023】

<装置概要の説明>

図22は、実施形態が適用する複写機のブロック構成図である。

【0024】

図中、1は装置全体の制御を司る制御部であり、CPU、ROM、RAM等で構成される。2はLCD表示器や各種スイッチ、ボタン等で構成される操作部である。3は原稿読取部(イメージスキャナ部)であり、ADF(Auto Document Feeder)を搭載している。読取った画像はRGB各色成分毎に8ビット(256階調)のデジタルデータを出力する

10

20

30

40

50

。4は、不図示のインターフェース（ネットワークインターフェースを含む）を介して受信したPDL形式の印刷データに基づき、印刷画像を描画するレンダリング部である。5はセレクタであり、制御部1からの指示に従って、原稿読取部3、又は、レンダリング部4から出力されたビットマップイメージのいずれか一方を選択し、出力する。

【0025】

6は本実施形態の特徴部分である符号化処理部である。この符号化処理部6の詳細な後述するが、画像データの符号化するものである。

【0026】

7は2次記憶装置（実施形態ではハードディスク装置とする）であり、符号化処理部6より出力された符号化データを順に格納記憶する。

10

【0027】

8は、復号処理部であって、2次記憶装置7に格納された圧縮符号化された画像データを、その格納順に読み出し、復号する。9は画像処理部であって、復号処理部8からの復号画像を入力し、RGB色空間を記録色空間であるYMCへの変換、UCR（Under Color Removal）処理を行うと共に、画像データの補正処理を行う。

【0028】

10はプリンタエンジン部である。プリンタエンジン部の印刷機構は、レーザビームプリンタエンジンとするが、インク液を吐出するタイプでも構わず、その種類は問わない。

【0029】

上記構成において、例えば、利用者が操作部2を操作して、複写モードを選択し、原稿を原稿読取部3（のADF）にセットし、複写開始キーを押下すると、原稿読取部3で読み取られた原稿画像データは、ラスター順に、セレクタ5を介して符号化処理部6に転送され、ここで圧縮符号化されて2次記憶装置7に格納していく。

20

【0030】

また、外部より印刷データを受信した場合には、セレクタ5をレンダリング部4を選択するようにし、レンダリング部4が生成した印刷データに基づく画像を圧縮符号化し、2次記憶装置7に格納することになる。

【0031】

復号処理部8は、プリンタエンジン10の印刷速度に応じて、2次記憶装置7から圧縮符号化データを読み出し、復号処理する。そして、画像処理部9で復号した画像データからYMC成分の記録用画像データを生成する。そして、その結果をプリンタエンジン部10に出力し、印刷を行う。

30

【0032】

上記の通り、2次記憶装置7への圧縮符号化データの格納処理と、復号し印刷するための読み出し処理は、非同期である。つまり、2次記憶装置7は画像圧縮処理と、復号処理との間に介在するバッファとして機能することになり、原稿読み取り／符号化処理は、復号／記録処理に非依存なので、多数の原稿を高速に読み取ることができ、次のジョブの原稿読み取りにすばやく移行することができる。

【0033】

図23は、符号化処理部6の処理を説明するためのフローチャートである。

40

【0034】

先ず、ステップS1で画像データを数ライン単位に読み取り、ステップS2にて符号化処理を行う。そして、ステップS3にて、1ページの符号化処理が完了したと判定するまで、画像の入力と符号化処理を行う。1ページの符号化処理が完了すると、ステップS4に進んで、生成した符号化データを補正する補正処理を行い、2次記憶装置7に出力する。そして、ステップS5にて、次ページがあると判定した場合には、ステップS1以降の処理を行うことになる。

【0035】

本実施形態における特徴となる点は、上記のステップS2とステップS4にある。そこで、先ず、実施形態における符号化処理部6の構成と、その構成に基づくステップS2の

50

処理を説明し、その後で、ステップ S 4 の処理を説明することとする。

【 0 0 3 6 】

< 符号化処理部 6 の説明 >

図 1 は、実施形態における符号化処理部 6 のブロック構成図である。以下、同図の各部を簡単に説明する。

【 0 0 3 7 】

入力部 1 0 1 には、複数ライン分のラインバッファーメモリを内蔵し、先に説明したように、セレクタ 5 を介して、原稿読取部 3 もしくはレンダリング部 4 からの画像データをラスター順に入力し、内部のラインバッファーに格納し、 $N \times M$ 画素ブロック（実施形態では 8×8 画素ブロックとした）単位に出力する。

10

【 0 0 3 8 】

第 1 の符号化部 1 0 2 は非可逆符号化部であって、圧縮率に影響を与えるパラメータに従って、入力部 1 0 1 より入力した画素ブロック単位に圧縮符号化処理を行い、その結果（符号化データ）を出力する。但し、符号化データの先頭には、第 1 の符号化部 1 0 2 で符号化されたことを示す識別ビットを付加する。

【 0 0 3 9 】

実施形態における、この画像符号化部 1 0 2 は、JPEG 符号化（非可逆符号化）を適用した例を説明する。つまり、 8×8 画素単位に相当する画像データを直交変換し、後述する量子化ステップを用いた量子化をし、ハフマン符号化処理を行うものである。ここで生成される符号化データ量を左右するのが量子化ステップであり、これが圧縮率に影響を与える符号化パラメータとして符号化シーケンス制御部 1 1 0 により設定される。JPEG 符号化は、自然画に適した技術として知られている。

20

【 0 0 4 0 】

図 2 1 は D C T 変換後の周波数係数を量子化する際に用いる量子化マトリクステーブル Q_0 、 Q_1 、 Q_2 を示している（符号化シーケンス制御部 1 1 0 に記憶保持されている）。ここで、量子化マトリクステーブル内の値 $Q_i(0, 0) \sim Q_i(7, 7)$ ($i = 0, 1, 2 \dots$) が量子化ステップ値を意味する。量子化ステップ値は、概ね、 $Q_0 < Q_1 < Q_2 \dots$ の関係にある。量子化ステップ値が大きくなればなるほど、量子化後の周波数係数値の取り得る範囲が狭くなり、圧縮率が向上する。

30

【 0 0 4 1 】

第 2 の符号化部 1 0 3 は、第 1 の符号化部 1 0 2 と異なり、可逆符号化部である。可逆符号化であるため、その復号結果は符号化する前の画像と同じとなり、原理的に画質の劣化は発生しない。実施形態では、この第 2 の符号化部 1 0 3 に JPEG - L S を用いた。JPEG - L S は「JPEG」を冠するものの、第 1 の符号化部で採用している非可逆符号化 JPEG とはそのアルゴリズムは全く異なる。JPEG - L S 符号化の特徴は、文字線画、コンピュータグラフィックスに適した技術であり、これらの画像を符号化する場合には、非可逆符号化である JPEG に与える量子化ステップが“1”（実質可逆）は勿論こと、“2”、“3”等の比較的小さい量子化ステップ値よりも少ない符号データを生成することができる。

40

【 0 0 4 2 】

また、第 2 の符号化部 1 0 3 は、第 1 の符号化部 1 0 2 と実質的に同じタイミングで、同じ画素ブロックに対して符号化を行い、符号化データを出力する。また、第 2 の符号化部 1 0 3 は、符号化データを出力する際に、その符号化データの先頭に第 2 の符号化部 1 0 2 で符号化されたことを示す識別ビットを付加する。

【 0 0 4 3 】

第 1 の符号長検出部 1 0 8 は、第 1 の符号化部 1 0 2 から出力される 1 画素ブロック分の、識別ビットである 1 ビットを含む符号化データの総長（以下、単に符号化データ長という）を検出し、符号化シーケンス制御部 1 0 8 に出力する。第 2 の符号長検出部 1 0 9 は第 2 の符号化部 1 0 3 から出力される 1 画素ブロック分の符号化データ長（識別ビットである 1 ビットを含む）を検出し、符号化シーケンス制御部 1 0 8 に出力する。

50

【0044】

符号化シーケンス制御部110は実施形態における符号化処理部6の制御を司るものであり、内部には第1カウンタ111が設けられている。このカウンタは、1ページの符号化を開始する際にリセットされ、第1の符号長検出部108からの符号長と第2の符号長検出部109からの符号長データを比較し、短い符号長データを選択し、累積加算するものである。また、短い符号長が判明するので、短い符号長の符号化データを格納するよう第1のメモリ制御部104に要求する。

【0045】

第1のメモリ制御部104は符号化シーケンス制御部110の制御の元で、第1のメモリ105への符号化データの書き込み、消去処理を行う。この第1のメモリ105は、1ページの符号化が完了した際の最終的に確定した符号化データを格納することになり、2次記憶装置7に出力される符号データを格納するものもある。

10

【0046】

第2のメモリ制御部106は符号化シーケンス制御部110の制御の元で、第2符号化部103で生成された符号化データの第2のメモリ107への格納処理、及び、第2のメモリ107より符号化データの読み出しを行う。この第2のメモリ107は、1ページの画像データを符号化する際の作業エリアとして使用されるものである。

【0047】

再符号化部112は、第2のメモリ107に格納された符号化データ（可逆符号化データ）中の、符号化シーケンス部110より指定された範囲の符号化データを再符号化する。この再符号化部112は例えば図20の構成になる。

20

【0048】

第2のメモリ107には第2の符号化部（JPEG-LS符号化部）103で符号化された符号化データが格納されているので、先ず、JPEG-LS復号部112aで原画像の画素ブロックに復号（復元）する。そして、JPEG符号化部112bは、復元された画素ブロックを、符号化シーケンス制御部110より設定された量子化マトリクステーブルに従ってJPEG符号化（非可逆符号化）を行う。そして、セレクタ112cはJPEG符号化部112bから出力された非可逆符号化データと、再符号化する以前の状態の可逆符号化データの小さい方を選択し、出力する。この処理を符号化シーケンス制御部106より指定された範囲について繰り返し実行する。第2のカウンタ113は、再符号化部112の再符号化が開始される際にリセットされ、再符号化部112で生成される符号化データ量を累積カウントする。設定された範囲の再符号化が完了した場合、第2のカウンタに保持されている値（再符号化で生成された符号化データ量）は、符号化シーケンス制御部110に出力される。

30

【0049】

以上、図1の構成について説明したが、全体の処理を更に詳しく説明すると次の通りである。

【0050】

1ページの画像データの符号化を開始する際に、入力される画像サイズに応じた目標データ量が符号化制御部110に制御部1より設定される。符号化シーケンス制御部111は第1の符号化部102に対して初期の量子化マトリクステーブルQ0（最高画質、且つ最低圧縮率に相当する符号化パラメータ）を設定し、第1のカウンタ111をゼロクリアする。そして、第1の符号化部102、第2の符号化部103に対して符号化処理を開始させる。以下は、1ページの画像データの入力と符号化処理の説明である。

40

【0051】

第1の符号長検出部108及び第2の符号長検出部109より、同じ画素ブロックに対する符号化データが出力されると共に、第1、第2の符号長検出部108、109でそれぞれの符号長が得られる。符号化シーケンス制御部110は、小さい方の符号長を選択し、第1のカウンタ111に加算させる。このとき、いずれを選択したのかを示す制御信号を第1のメモリ制御部104に出力する。なお、2つの符号化長が同じであった場合には

50

、可逆符号化データを選択したことを示す制御信号を出力する。

【0052】

第1のメモリ制御部104には、符号化シーケンス制御部110より上記の制御信号を受け取り、符号長の短いと判定された符号化データを選択し、第1のメモリ105に格納する。

【0053】

以上の結果、第1のメモリ105には、画像データの各画素ブロックについて符号長の短い符号化データが格納されていく。つまり、第1のメモリ105には、可逆符号化データと非可逆符号化データが混在して格納されていく。一方、第2のメモリ107には可逆符号化データのみが格納されていくことになる。

10

【0054】

ここで注意したい点は、第1のカウンタ111には第1のメモリ105に格納される符号化データの総符号量の情報が格納されることである。

【0055】

符号化シーケンス制御部110は、1ページ分の画像の符号化処理が進行中、第1のカウンタ111の値、すなわち、第1のメモリ105に格納される符号化データの総量を監視し、目標データ量を越えた（到達も含む）か否かを判定する。そして、第1のカウンタ111に格納された値（総符号量）が、目標データ量を越えたと判定した場合、符号化シーケンス制御部110は次の処理を行う。

1. 第1のメモリ105内の格納済みのデータを廃棄するよう第1のメモリ制御部104に制御信号を出力する。第1のメモリ制御部104は、この制御信号に基づいて、メモリアドレスカウンタをクリアするか、あるいは、符号化データ管理テーブルをクリアすることにより、格納された符号化データを廃棄する。

20

2. 第1のカウンタ111をゼロクリアする（入力部101からの画像の入力は継続している）。

3. 第1の符号化部102に対して、今までより高い圧縮率で符号化を行うようするため、量子化マトリクステーブルを更新する。すなわち、従前にセットされた量子化マトリクステーブルが Q_i であった場合には、 Q_{i+1} をセットする。初期状態では Q_0 がセットされていたので、最初に目標量を越えたと判定した場合には Q_1 がセットされることになる。

30

4. 第2のカウンタ113をゼロクリアし、再符号化部112に量子化マトリクステーブル Q_{i+1} をセットし、第2のメモリ107に格納されている符号化データの再符号化を開始させる。再符号化して得られた符号化データ（可逆・非可逆符号化データが混在している）は、第2のメモリ107に再格納する。なお、第2のメモリ107には、第2の符号化部103からの符号化データも格納されているので、両者は区別して格納される。

5. 再符号化が完了した場合、第2のメモリ107に「再」格納された符号化データを、第1のメモリ105に転送すると共に、第2のメモリより削除する（符号化部103からの符号化データは削除しない）。そして、第2のカウンタ113の値を読み出し、第1のカウンタ111に足し込む。この結果、第1のカウンタ111は再び第1のメモリに格納された符号化データの総量を示すことになる。

40

【0056】

なお、再符号化処理が終了したかどうかは、第2のメモリ制御部106が検出する。すなわち、再符号化処理するために読み出すデータが無くなれば、再符号化処理の終了を符号化シーケンス制御部110に知らせる。実際には、第2のメモリ制御部107の読み出し処理だけでなく、再符号化部111の処理も終了した後に、符号化処理が完了したことになる。

【0057】

そして1ページの画像の入力と符号化が完了する前に再び総符号化データ量が目標データ値を越えたと判断した場合、上記工程1乃至5を実行する。従って、最終的に、第1のメモリ105には目標データ量以下の符号化データが格納されることになる。

50

【0058】

上記、図1の構成における符号化シーケンス制御部110の処理手順を図9のフローチャートに示すが、説明を簡単にするために、簡略化した図3のフローチャートに従ってまず、説明する。

【0059】

図3のフローチャートは、大別すると、下記の3つの処理フェーズに分かれる。

- (1) 符号化フェーズ
- (2) 符号化・再符号化フェーズ
- (3) 転送フェーズ

上記それぞれの処理フェーズにおいて、どのように画像データ、符号化データ等が流れ 10
て処理され、メモリにどのように格納されるのかを視覚的にわかりやすく示したのが図4
乃至図8である。

【0060】

図4は、図3のフローチャートにおけるステップS303とS305に対応する符号化フェーズの初期状態を表す。なお、図4におけるスイッチ40は、第1の符号化部102からの符号化データと第2の符号化部103からの符号化データのうち、符号長の短い方を選択することを示している。このスイッチ40は、符号化シーケンス制御部110及び第1のメモリ制御部104の機能で実現することになる。第1のメモリ105には2つの符号化部の符号データのうち符号長の短い方が格納される。従って、第1のメモリ105に格納されるデータ量Iは、第2のメモリ107に格納されるデータ量I'よりも少ない 20
関係が維持されることになる。

【0061】

図5は、ステップS309で量子化マトリクステーブルを変更した際の状態を示している。図示のように、第1のメモリ105内の符号化データはない。

【0062】

図6は、ステップS311～S315に対応する符号化・再符号化フェーズの処理状態を示し、図7はステップS317に対応する転送フェーズの処理状態を、図8は転送フェーズ後の符号化フェーズの処理状態を表す。以下、各フェーズについて説明する。

【0063】

<< 符号化フェーズ >>

1ページ分の画像データの符号化処理は、符号化パラメータの初期設定処理（ステップS301）から始まる。ここでは、符号化処理する画像サイズ（ページ記述レンダリング等の入力部101から読み取る用紙サイズ）から一意的に定まる符号化データ量の目標データ量や第1の符号化部102に適用する量子化マトリクステーブルQ0を設定する処理である。

【0064】

この後、ステップS303にて、第1の符号化部102及び第2の符号化部103にて符号化処理を開始させる。この結果、第1のメモリ105には、画素ブロック単位に、第1の符号化部102からの符号化データ及び第2の符号化部103からの符号化データのうち符号長の短い一方の符号化データを格納されていく。第1のメモリ105に格納されている符号化データ量は、第1のカウンタで計数する点は既に述べた通りである。一方、第2のメモリ106には、第2の符号化部103からの符号化データが格納されていくことになる。図4は、この状況を示している。第1のメモリ105に格納されるデータ量を示す領域Iは、少なくとも第2のメモリ107に格納されるデータ量を示す領域I'以下のなる。

【0065】

次にステップS305にて、該データ量のカウント値が上記目標値をオーバーしたかどうかを検知し、オーバーしていなければステップS303の第1の符号化と第2の符号化処理を継続する。これが、初期状態の符号化フェーズである。

【0066】

10

20

20

30

40

40

50

<<符号化・再符号化フェーズ>>

符号化処理が進行し、第1のメモリに格納される総符号データ量が目標量をオーバーすると、ステップS307にて第1のメモリ105内の符号化データを廃棄すると共に、ステップS309にて、第1の符号化部102に設定されていた量子化マトリクステーブルQ0を、次の段階のQ1に更新する。総符号化データ量が目標データ量をオーバーすることは、圧縮後のデータ量が目標以内に収まらないことを意味する。よって同じ量子化ステップを用いて符号化処理を継続しても意味がないので、前よりも量子化ステップ幅の大きい量子化ステップQ1に変更するわけである。

【0067】

量子化ステップを変更した後、ステップS311では、第1の符号化部102と第2の符号化部103の符号化処理を再開する。また、再符号化部112に量子化マトリクステーブルQ1（第1の符号化部にセットした更新後の量子化マトリクステーブルと同じ）をセットし、第2のメモリに格納されている符号化データ量の再符号化を開始させ、再符号化結果を再び第2のメモリに格納させる。この状態を示すのが図5である。

10

【0068】

そして、ステップS315で再符号化の処理が完了するまで待つことになる。

【0069】

<転送フェーズ>

ステップS315にて、再符号化が完了したと判断した場合、第1メモリ105及び第2のメモリ107の符号化データの格納状況を示すのが図6である。図示において、領域II、II'は、再符号化処理を行っている最中に、新に入力した画像データに対応する符号化データを示し、領域Iは、領域I'に格納されていた符号化データを再符号化した結果（非可逆符号化データと可逆符号化データが混在している）を示している。

20

【0070】

ステップS317では、図7に示すように、第2のメモリ107に格納された再符号化データ（図示の領域I）を、第1のメモリ105に転送する。この転送が完了すると、第2のメモリ107内の領域Iのデータは破棄（もしくは上書きを許容）する。

【0071】

上記転送フェーズが終了したら、ステップS303、S305の符号化フェーズに戻り、符号化フェーズに戻ることになる。この結果、図8に示すように、新たな入力画像データに対する符号化データII、II'をそれぞれのメモリへの格納処理を行うことになる。この符号化フェーズは、初期状態の符号化フェーズ（図4）と少し異なり、第1の符号化部102で符号化する際の量子化ステップがQ0からQ1に変更されていると共に、第1のメモリ105に格納される符号化データの順序が、画像入力順にならない点である。それらの違いを無視すれば、転送フェーズ直後の符号化フェーズと初期状態の符号化フェーズは、同じと見なせる。なお、図8に示すように、符号化データの順番は必ずしも時系列になるとは限らないが、各フェーズにおける格納アドレスを別途記憶しておき、1ページの符号化処理が完了し、補正処理部120（詳細後述）を介して2次記憶装置7に出力する際には、第1のメモリ105より時系列に符号化データを読み出し、出力するようすれば良い。

30

【0072】

よって、符号化フェーズ、符号化・再符号化フェーズと転送フェーズの3つを繰り返すことで、最終的に1ページの画像データをデータ量設定値以下に圧縮した符号を第1のメモリ105に格納することが出来る。しかも、入力部101は一連の処理が終わるまで、入力を継続するだけである。すなわち、画像を再度最初から入力し直すということが無くなる。

40

【0073】

図3に示したフローチャートは、説明が理解しやすいように、図4乃至図8に示した各フェーズに対応する処理のみを記述した。しかしながら実際には、1ページの画像データの入力はどこかのフェーズで終了する。従って、どのフェーズで終了したかによって、そ

50

れ以降の対応も多少異なる。それを考慮した流れを示したのが図9のフローチャートである。図9のフローチャートは、1ページ分の画像データの入力完了と図3で説明した各種処理との関係を考慮したものであり、ここでは図3のフローチャートに、ステップS801、S803、S805、S807を追加している。

【0074】

ステップS801、S803、S805は、それぞれ、符号化フェーズ、符号化・再符号化フェーズ、転送フェーズにおいて、入力部101からの1ページ分の画像データの入力が終了したことを検知する。

【0075】

符号化フェーズと転送フェーズで1ページ分の画像データの入力が終了したことを検知した場合(ステップS801、S805)、ステップS807へ移り、当該ページの圧縮符号化処理を終了し、次に処理すべき1ページ以上の画像データがあれば、次の1ページ分の画像データの圧縮符号化処理を開始し、無ければ停止状態に入る。

【0076】

一方、符号化・再符号化フェーズで1ページ分の画像データの入力終了を検知した場合(ステップS803)には、第1の符号化部102、第2の符号化部103では再符号化処理する画像データが無くなるまで一旦動作を止める必要があるので、ステップS311の符号化処理をパスし、ステップS313で、今までに第1の符号化部102、第2の符号化部103で符号化済みの画像データを所定の符号化データ量に抑える為の再符号化処理のみを継続して行なう。再符号化処理が全て終了して、その後の転送処理が終わらないと、1ページ分の画像データ全体の符号化データが第1のメモリ上に集まらないため、1ページ分の画像データの入力終了後も再符号化処理及びそれに続く転送処理は継続して行われる必要がある。この場合には、ステップS315にて、再符号化処理が全て終了したことを検知すると、符号化・再符号化フェーズ中に、第2のメモリ107のみに格納された符号化データを第1のメモリに転送し(ステップS317)た後、次のステップS805にて、1ページ分の画像データの入力終了が検知された場合には、ステップS807へ移ることになる(補正処理に以降する)。

【0077】

以上が実施形態における符号化処理部6における符号化処理の内容であり、図9の動作説明もある。

【0078】

以上説明したように、非可逆符号化データを生成する第1の符号化部102、可逆符号化データを生成する第2の符号化部103という異なる符号化技術を用いながらも、1ページの画像を入力を中断することなく、且つ、再入力を行うことなく、目標符号化データ量以下の符号化データを生成することが可能となる。

【0079】

特に、注目したい点は、JPEGとJPEG-LSという2つの技術を併用している点である。JPEG符号化は自然画に対して圧縮効率が高いことが知られている。一方、JPEG-LS符号化は、文字線画に対して高い圧縮率が得られるし、可逆符号化であるため原画像を忠実に再現することが可能である。

【0080】

かかる点から、実施形態における画像符号化処理部6は次のような作用効果が期待できることになる。なお、以下では目標符号化データ量は、1ページ全体が自然画である場合を想定したものとする。

1. 原稿が文字のみで構成される通常原稿の場合、第1のメモリ105に格納される画素ブロック単位の符号化データとしては、第2の符号化部103からのJPEG-LS符号化データ(可逆符号化データ)が選択される可能性が高くなる。また、その際の圧縮率は非常に高いものであり、総符号化データ量が目標符号化データ量をオーバーしないまま符号化が完了することが期待できる。また、可逆符号化データであるので、印刷結果も原画像に忠実なものとなり、印刷結果の画質は符号化が介在しないのと実質的に等価のものとなる。

10

20

30

40

50

2. 原稿が自然画のみである場合、第1のメモリ105に格納される符号化データとして、第1の符号化部102からのJPEG符号化データ（非可逆符号化データ）が選択される可能性が高くなる。また、目標データ量をオーバーする回数も、数回になる可能性がある。しかし、もともと自然画であるため階調性が画質を決めるファクタであるので、実害のない画質で印刷が可能となる。

3. 原稿が自然画と文字が混在している場合、文字線画領域ではJPEG-LS符号化データが選択される可能性が高く、自然画ではJPEG符号化データが選択される可能性が高くなる。文字線画領域では、圧縮効率が高いJPEG-LS符号化が選択される可能性が高いわけであるから、1ページを占める自然画領域と文字線画の比以上に、自然画領域に割り当てる符号化データ量を大きくすることができます。つまり、写真画像の量子化ステップの値は比較的小さなままで、符号化処理が完了することが期待できるので、文字線画は勿論のこと、写真画像領域の画質劣化を抑制することが可能になる。

4. 更に、上記1乃至3を、像域判定回路等の格別な回路構成無しに、実現できることも大きな利点である。

【0081】

ここで、第1カウンタ111（符号化データ量）の時間軸の推移の一例を図13に示す。

【0082】

図示の場合、タイミングT0で、原稿画像の入力を開始し、タイミングT5で原稿画像の入力が完了した場合を示している。入力される原稿のサイズが固定であれば、タイミングT5は固定となる。以下、各タイミングでの処理を説明する。

【0083】

タイミングT0：

画像入力開始（符号化開始）タイミングである。このとき、第1の符号化部102には、初期値として量子化マトリクステーブルQ0をセットし、第1カウンタ111は“0”に初期化される。この後、画像の入力及び符号化が継続すると、画像データの符号化データの符号化が行われ、第1カウンタ111のカウント値は徐々に増えていく。

【0084】

タイミングT1：

画像データの符号化データ量が、目標符号量に到達した場合を示している。このとき、第1のメモリ105の画像データの符号化データは破棄し、第1カウンタ111をゼロクリアし、第1の符号化部102にセットされていた量子化マトリクステーブルをQ1に更新する。また、再符号化部112にも量子化マトリクステーブルQ1をセットし、再符号化処理を開始させる。

【0085】

タイミングT2：

再符号化及び転送処理の完了を示している。再符号化が完了すると、再符号化して得られた符号データが第2のメモリ107から第1のメモリ105に転送されると共に、再符号化データ量を示す第2のカウンタ113の値が、第1のカウンタ111に足し込まれる。この結果、第1のメモリ105及び第2のメモリ107には、1ページの先頭からタイミングT2までの間に入力した画像データに対する量子化マトリクステーブルQ1で符号化したのと等価の符号化データが格納される。

【0086】

タイミングT3：

再び、画像データの符号化データ量が、目標符号量に到達した場合を示している。このとき、第1のメモリ105の画像データの符号化データは破棄し、第1カウンタ111をゼロクリアし、第1の符号化部102の量子化マトリクステーブルをQ2をセットする。また、再符号化部112にも量子化マトリクステーブルQ2をセットし、再符号化処理を開始させる。

【0087】

10

20

30

40

50

タイミング T 4 :

再符号化及び転送処理の完了を示している。再符号化が完了すると、再符号化して得られた符号データが第2のメモリ107から第1のメモリ105に転送されると共に、再符号化データ量を示す第2のカウンタ113の値が、第1のカウンタ111に足し込まれる。この結果、第1のメモリ105及び第2のメモリ107には、1ページの先頭からタイミングT2までの間に投入した画像データに対する量子化マトリクステーブルQ2で符号化したのと等価の符号化データが格納される。

【0088】

タイミング T 5 :

1ページの原稿投入の完了を示している。この場合、第1のメモリ105には、1ページの画像の符号化データが格納されていることになるので、その結果を、補正処理部120を介して2次記憶装置7に出力する。なお、このとき、補正処理部120に対しては、最終的に第1の符号化部102で利用した量子化マトリクステーブルが何であったのかも通知する（理由は後述）。

【0089】

ここで、2枚目の原稿画像を読み取る場合には、上記タイミングT1からの処理を繰り返すことになる。

【0090】

なお、画像によっては、原稿画像の投入が完了したタイミングT5の直前に、第1カウンタ111の値が目標量をオーバーすることも有り得る。この場合、タイミングT5の後に、再符号化及び転送処理が行われることになる。従って、第1のメモリ105に格納された符号化データを2次記憶装置7に出力する条件は、原稿画像の投入が完了し、且つ、符号化（再符号化及び転送）が完了した場合となる。

【0091】

また、再符号化112による再符号化しようとする符号化データ量は、第1カウンタ111の値が目標データ値をオーバーする回数を重ねるほど、多くなる傾向にある。換言すれば、目標値をオーバーする回数が少ないほど、符号化処理に要する時間が少ないとなる。

【0092】

以上、図23におけるステップS2の処理を説明した。次に、図23におけるステップS4の補正処理について説明する。

【0093】

既に説明したように、1ページの符号化処理が完了した場合、第1のメモリ105に格納される各画素ブロック毎の符号化データは、非可逆符号化データか可逆符号化データのいずれかになり、これら2種類の符号化データが混在したものとなる。

【0094】

実施形態では、非可逆符号化としてJPEG、可逆符号化としてJPEG-LSを例にしているので、格別な像域判定回路或いは手段を設けなくても、結果的に、投入する1ページの画像中の自然画領域ではJPEG符号化データが、文字線画領域ではJPEG-LS符号化データが採用される確率が高くなる。

【0095】

しかしながら、これはあくまで確率であって、文字線画領域内の全画素ブロックがJPEG-LS符号化に基づく可逆符号化データになることを保証するものではないし、自然画領域内の全画素ブロックがJPEG符号化に基づく非可逆符号化データになることを保証するものではない。つまり、注目画素ブロックの符号化データが可逆符号化データであり、注目画素ブロックの回りの画素ブロック群が非可逆符号化データとなることも起こり得るし、その逆になる場合も起こり得る。このように、2次元的に符号化種類の孤立した符号化データ（以下、孤立符号化データという）が発生すると、その復号結果の画像では、その画素ブロックの境界で画素の濃度や色が不連続となり、不自然な画像になる。

【0096】

10

20

30

40

50

特に、注目画素ブロックは可逆符号化データであって、その回りが非可逆符号化データである場合に、この問題は顕著になる。

【0097】

そこで、本実施形態では、このような孤立可逆符号化データを検索し、その孤立可逆符号化データをその回りの非可逆符号化データに合わせるべく、注目している孤立可逆符号化データを、符号化シーケンス制御部111より通知された量子化マトリクステーブルで再符号化する。この処理を行うのが補正処理部120であり、図23におけるステップS4の処理である。

【0098】

図24は補正処理部120の処理内容を示すフローチャートである。以下、同図に従つて、補正処理部120の処理手順を説明する。

10

【0099】

先ず、ステップS11で注目符号化データ（1つの画素ブロックの符号化データ）とその回りの8個の計9個の符号化データを読み出す。ただし、注目符号化データがページの最上部、左端、右端、及び、最下部に位置する場合、9個の符号化データを読み出すことはできない。このような場合には、存在しない位置の画素ブロックの符号化データの種類は注目画素ブロックの符号化データの種類と同じとして扱う。

【0100】

ステップS12では、注目符号化データが孤立符号化データであるか否かを判定する。符号化データが可逆符号化データであるか、非可逆符号化データであるかは、その先頭ビットを調べれば判明するのは上記実施形態から明らかである。従って、注目符号化データの回りの8つの符号化データの種類が全て同じであって、注目符号化データの種類が回りと異なれば孤立符号化データであると判定できる。

20

【0101】

孤立符号化データではないと判定した場合には、ステップS15に進んで、注目符号化データをそのまま2次記憶装置7に出力する。

【0102】

一方、孤立符号化データであると判定した場合には、ステップS13に進んで、注目符号化データは可逆符号化データであるか否かを、その先頭ビットに基づいて判定する。

30

【0103】

注目符号化データが可逆符号化データではない（非可逆符号化データである）と判断した場合にはステップS15に進んで、そのまま出力する。

【0104】

また、注目符号化データが可逆符号化データであると判定した場合には、ステップS14にて非可逆符号化データに変換する。可逆符号化データであるから、現画像の画素ブロックのデータに復元できる。従って、その復号結果を、符号化シーケンス制御部110より通知される量子化マトリクステーブルを用いて符号化（JPEG符号化）すればよい。そして、ステップS15に進んで、変換結果である符号化データを出力する。

【0105】

以上の処理を、ステップS16にて、1ページ分の符号化データについて全て行ったと判定するまで繰り返すことになる。

40

【0106】

なお、第1のメモリ105には、2つの符号化部から出力される符号化データのうち、符号長が短い方が格納されている。従って、上記の補正処理部120を介して出力される1ページ分の符号化データ量は、第1のメモリ105に格納されていたデータ量を下回ることはない。換言すれば、補正処理部120から出力される1ページ分の符号化データ量は、符号化処理で設定された目標符号化データ量を越えることが起こり得ることを意味する。

【0107】

しかしながら、この補正処理部120で、符号化種別が変更される符号化データは、孤

50

立可逆符号化データを対象にしており、その数は全画素ブロックの個数の数パーセント程度であり、十分に許容される範囲内にあることが確かめられている。また、実施形態のように、複写機に適用する場合には、仮に目標データ量を越えたとしても、画像形成には影響は発生しないので、実害はない。

【0108】

以上説明したように本実施形態によれば、非可逆圧縮と可逆圧縮技術を併用しつつ、一度のページ画像データの入力で、ブロックノイズの発生を抑制しつつ、目標とするデータ量の符号化データを生成することが可能になる。

【0109】

また、非可逆符号化としてJPEG、可逆符号化としてJPEG-LSを用いることで、格別な像域判定回路を設けなくても、文字線画、自然画それぞれに対して適した符号化が選択されることになり、復号した際の画質及び圧縮率ともに優れた符号化データを生成することが可能になる。

【0110】

以上であるが、実施形態の構成は様々な改良並びに変更しても構わない。以下に説明するのは、実施形態が適用可能な他の符号化処理部6、並びに、他の補正処理部120の例である。

【0111】

<符号化処理部6の変形例1>

図2は、図1に対する変形例である。図1と異なる点は、再符号化部112で再符号化して得られた符号化データの格納先を、第1のメモリ105にした点である。これ以外の構成は図1と同じである。

【0112】

図10は、第1のメモリ105に格納された総符号化データ量が目標データをオーバーし、第1のメモリ105内のデータを破棄し、第2のメモリ107に格納されている符号化データ(図示の領域I')の再符号化を開始する際の状態を示している。

【0113】

図示に示すように、再符号化部112の再符号化の格納先は第1のメモリ105に設定して、再符号化を開始させる。

【0114】

図11は、再符号化が完了した際の2つのメモリの符号化データの格納状態を示している。再符号化が完了すると、図示のように、第1のメモリ105には、領域Iで示す符号化データが格納される。この領域Iの符号化データは、目標データ量をオーバーしたと判定する以前までに入力した画像データを、量子化マトリクステーブルQ1で符号化した場合に相当するものとなる。

【0115】

再符号化が行われている最中にも、画像データの入力及び符号化が行われているので、図示に示すように領域II及びII'が存在する。

【0116】

第1のメモリ105には、空き領域105aが生成されることになるので、図示の領域IIの符号化データを、領域Iの後端位置にまで移動させる。この後、符号化フェーズを再開することになり、この際のメモリへの格納状態は図12に示すようになる。

【0117】

図1に対して図2の構成のメリットは、再符号化後のデータ転送が実質的に無くなる点である。

【0118】

なお、この変形例では、転送フェーズが無くなるだけであるので、先の説明と本変形例は、3つのフェーズを繰り返して処理することに変わりは無い。従って、処理の内容も図3や図9と殆ど同じであるので、説明は不要であろう。なお、図11では空き領域105aを無くすために、第1のメモリ105内の移動を行ったが、各領域の接続関係を管理

10

20

30

40

50

するファイル管理テーブル、或いは、パケット管理テーブル等を設ければ良いので、必ずしも内部転送は必要ではない。

【0119】

<符号化処理部6の変形例2>

上記実施形態及び変形例1では、第1のメモリ105内の符号化データ量が目標データ量をオーバーした際に、第2のメモリ107の符号化データを再符号化部112で再符号化した。換言すれば、再符号化部112は、第1のメモリ105内の符号化データ量が目標データ量未満である期間は実行しないことになる。

【0120】

そこで、期間を有効利用する例を変形例2として説明する。

10

【0121】

以下は、第1のメモリ105内の符号化データ量が目標データ量未満の場合であって、第2のメモリ107に第2の符号化部103からの可逆符号化データが格納されていく最中の処理の説明である点に注意されたい。また、条件として、第1の符号化部102に設定される量子化マトリクステーブルが Q_i であるものとして説明する。

【0122】

第2のメモリ107に第2の符号化部103より可逆符号化データを格納していくのは、これまでの実施形態と同じである。但し、再符号化部112は、格納されている可逆符号化データから順に読み出し、量子化マトリクス Q_{i+1} で再符号化して非可逆符号化データを生成し、その結果を第2のメモリ107に格納していく処理を前倒し的に行う。

20

【0123】

この結果、第1のメモリ105内の符号化データが目標データ量をオーバーしたとき、第2のメモリ107には可逆符号化データは勿論のこと、ある程度の非可逆符号化データが存在することが約束されるので、同一画素ブロック位置であれば小さいほうを第1のメモリに転送し、比べる対象となる非可逆符号化データが存在しない画素ブロックについてだけ、図20による処理を行えば良いことになる。

【0124】

また、第2のメモリ107から第1のメモリ105への転送が完了した場合、第2のメモリ内に格納されていた非可逆符号化データを破棄し、今度は、量子化マトリクス Q_{i+2} を設定し、再符号化することを開始すればよい。以上の結果、再符号化に係る処理を更に短縮させることが可能になる。

30

【0125】

<符号化処理部6の変形例3>

先に説明した第1の実施形態では、1ページの画像入力中に符号化データ量（第1のカウンタ111の値）が目標値をオーバーした際に一旦ゼロクリアした。そして、第1のカウンタ110が再び正確な総符号量を保持するのは、再符号化部112による再符号化が完了した場合になる。つまり、図13におけるタイミングT1-T2間、及び、T3-T4間では、一時的に、符号化データ量が不明な状態になる。また、タイミングT5で原稿画像の入力が終了したとしても、たまたまタイミングT5の直前で、総符号量が目標値をオーバーした場合には、再符号化部112による再符号化が完了しない限り、次の原稿画像の入力が行えない。

40

【0126】

本変形例3では、上記の問題を解決する。本変形例3における符号化処理部6の構成を図14に示す。

【0127】

図1との違いは、JPEG（非可逆）符号化部としての第3の符号化部114、第3の符号化部114で生成される画素ブロックの符号長を検出する第3の符号長検出部115、第3のカウンタ116、第3のメモリ制御部117及び第3のメモリ118を設けた点にある。また、第3の符号化部114は、生成する符号化データの先頭にJPEG符号化（非可逆符号化）であることを示すビットを付加する点は第1の実施形態と同じである。

50

【0128】

また、図1では非可逆（JPEG）符号化を行うJPEG符号化部が1つであったのに対し、本変形例3では図14に示す如く、第1の符号化部102、第3の符号化部202の2つになり、これら2つの非可逆符号化部が並行して符号化処理を行う。また、第1の符号化部102と第3の符号化部114との違いは、設定するパラメータにある。つまり、第1の符号化部102に設定されていた量子化マトリクステーブルを Q_i としたとき、第3の符号化部114には常に1段階大きな量子化マトリクステーブル Q_{i+1} を設定する。1ページの符号化開始の初期段階では、第1の符号化部102にセットする量子化マトリクステーブルは Q_0 であり、第3の符号化部114にセットする量子化マトリクステーブルは Q_1 となる。

10

【0129】

また、第1の符号化部102、第2の符号化部103、第3の符号化部114はそれぞれ同じタイミングで同じ画素ブロックについて符号化処理を行い、ほぼ同時に符号化データを生成し、出力するものとする。

【0130】

さらにまた、第2の符号化部103から出力される可逆符号化データは第3のメモリ制御部117を介して無条件に第3のメモリ118に格納する。また、1ページの符号化が完了した場合、第1のメモリ105に格納されている符号化データが最終的に求める符号化データとなるのは第1の実施形態と同様である。

【0131】

20

以下、図14の構成における符号化シーケンス制御部110の処理概要を説明する。

【0132】

本変形例3における符号化シーケンス制御部110における画素ブロック単位の基本処理は次の通りである。

1. 第1の符号化部102にセットする量子化マトリクステーブルを Q_i としたとき（初期段階では Q_0 ）、第3の符号化部114には常に1段階大きな量子化マトリクステーブル Q_{i+1} （初期段階では Q_1 ）をセットする。

2. 第1の符号化部102から出力される非可逆の符号化データと第2の符号化部103から出力される可逆の符号化データの小さい方が第1のメモリ105に格納されるよう、第1のメモリ制御部に制御信号を出力する。また、第1のカウンタ111には、2つの符号化データの符号長のうち短い方を累積加算する。つまり、第1のカウンタ111には、第1の実施形態と同様、第1のメモリ105に格納されている符号化データ量を示す情報を格納することになる。

30

3. 第3の符号化部114から出力される非可逆符号化データと第2の符号化部103から出力される可逆符号化データの小さい方を第2のメモリ107に格納するよう制御信号を出力する。また、第3のカウンタ116には、2つの符号化データの符号長のうち短い方を累積加算する。

【0133】

次に、1ページの符号化処理中の符号化シーケンス制御部110の処理の概要を説明する。

40

【0134】

符号化シーケンス制御部110は、第1のカウンタ111の値（第1のメモリ105に格納された符号化データ量）が目標データ量をオーバーしたか否かを判定する。目標データ量をオーバーしないと判定し、1ページの符号化処理が完了した場合には、第1のメモリ105内に格納された符号化データを、符号化処理部6の符号化結果として補正処理部120を介して2次記憶装置7に出力する。

【0135】

一方、第1のカウンタ111の値が、目標データ量をオーバーしたと判定した場合、符号化シーケンス制御部110は次の各工程を行う。

工程1：第1のメモリ105内の画像データの符号化データを破棄するよう第1のメモリ

50

制御部 104 に要求する。

工程 2：第 2 のメモリ 107 に格納された符号化データ（量子化マトリクステーブル Q_{i+1} による非可逆符号化データと可逆符号化データが混在したデータ）を第 1 のメモリ 105 に転送する。これに伴い、第 1 のカウンタ 111 の値を第 3 のカウンタ 116 の値で更新する（第 1 のカウンタ 111 に第 3 のカウンタ 116 の値を上書きする）。

工程 3：第 2 のメモリ 107 から第 1 のメモリ 105 への転送が完了した場合、第 2 のメモリ 107 内のデータを破棄するよう第 2 のメモリ 106 に制御信号を出力する。

工程 4：第 1 の符号化部 102 への量子化マトリクステーブルを Q_{i+1} に更新し、第 3 の符号化部 114 への量子化マトリクステーブルを Q_{i+2} に更新し、符号化を継続させる。

工程 5：再符号化部 112 に量子化マトリクステーブル Q_{i+2} （更新後の第 3 の符号化部 114 の量子化マトリクステーブルと同じ）をセットし、第 3 のメモリ 118 に格納された可逆符号化データを再符号化させる。再符号化部 112 は、第 1 の実施形態と同じ構成であるので、その説明は省略する。再符号化処理で得られた符号化データは、第 3 のメモリに再格納する。このとき、再符号化処理で生成された符号化データ量は第 2 のカウンタ 113 に格納される。

工程 6：工程 5 の再符号化が完了したら、その再符号化処理して得られた符号化データを第 2 のメモリに転送し、第 2 のカウンタ 113 の値を、第 3 のカウンタ 116 に加算する。

【 0136 】

上記工程 1 乃至 6 において、工程 1 乃至 5 は、入力部 101 による画像データの入力に対して十分に高速に処理可能である。従って、図 9 におけるタイミング $T_1 - T_2$ 間、或いは、 $T_3 - T_4$ 間のように、符号化データ量が不明な期間は短いものとすることができる。特に、工程 2 での第 1 のカウンタ 111 の更新処理は、転送処理が完了する以前に行つても構わない。この場合には、符号化データ量が不明な期間は実質的に無いものとすることが可能になる。

【 0137 】

なお、図 9 のタイミング T_5 の直前において、第 1 のカウンタ 111 で示される符号化データ量が目標値をオーバーした場合、上記工程 1 から処理が開始されるものの、工程 2 まで完了すれば、求める符号化データを第 1 のメモリ 105 に格納したことになるので、2 次記憶装置 7 に書き込みを行い、工程 3 以降の処理は行わず、次ページの読み込みと符号化処理を開始して構わない。

【 0138 】

以上、図 14 の構成における動作概要を説明した。本変形例 3 のように、2 つの非可逆符号化部と、1 つの可逆符号部を並行して設ける場合は、図 15 に示すフローチャートに基づいて 1 ページ分の画像データの符号化を行うことになる。

【 0139 】

なお、図 15 の処理は、その大半の処理が図 9 と類似しているので、3 つのフェーズで処理を説明する様にし、図 9 と異なる点を主に説明することとする。

【 0140 】

上述した図 9 のフローと変形例 3 のフローとの一番大きな違いは、ステップ S317 における第 2 のメモリ 107 から第 1 のメモリ 105 への転送処理が、ステップ S307 とステップ S309 の間に移動している点、この転送処理が終わった際に、第 2 のメモリ 107 内のデータを破棄する処理（ステップ S320）が追加された点、並びに、再符号化処理が完了した場合に第 3 のメモリから第 2 のメモリへの転送処理（ステップ S321）が追加された点である。

【 0141 】

ステップ S301 の符号化パラメータの初期設定では、第 1 の画像符号化部 102 に設定すべき初期量子化マトリクステーブル Q_0 を決定し、第 3 の画像符号化部 114 はそれより 1 段階大きな Q_2 を設定する。

【 0142 】

10

20

30

40

50

符号化フェーズでは、ステップ S 8 0 1、S 3 0 3、S 3 0 5 を繰り返し実行する。図 16 は、符号化フェーズにおける第 1 乃至 3 のメモリの符号化データの格納状態を示している。図示において、スイッチ 4 0 は第 1 の実施形態と同じ意味である。スイッチ 4 1 は第 2 のメモリ制御部 1 0 6 及び符号化シーケンス制御部 1 1 0 によって構成されるものである。すなわち、スイッチ 4 1 は、画素ブロック単位に、第 1 の符号化部 1 0 2 及び第 2 の符号化部 1 0 3 それぞれより出力される符号長の短い符号化データを選択し、第 1 のメモリ 1 0 5 に格納する。スイッチ 4 1 は、画素ブロック単位に、第 3 の符号化部 1 1 4 及び第 2 の符号化部 1 0 3 それぞれより出力される符号長の短い符号化データを選択し、第 2 のメモリ 1 0 7 に格納する。そして、第 3 のメモリ 1 1 8 には第 2 の符号化部 1 0 3 からの符号化データを無条件に格納する。

10

【 0 1 4 3 】

今、図 16 において、第 1 のメモリ 1 0 5 に格納された符号化データ量（図示の領域 I ）が、目標値をオーバーした場合を考察する。つまり、ステップ S 3 0 5 にて、目標データ量をオーバーしたと判定された場合である。

【 0 1 4 4 】

このとき、第 1 のメモリ 1 0 5 に保持していた符号化データを廃棄し（ステップ S 3 0 7 ）、第 2 のメモリ 1 0 7 で保持している圧縮率の高い符号化データ（領域 I' ）を、第 1 のメモリ 2 0 4 へ転送すると共に、第 3 のカウンタ 1 1 6 の値を第 1 のカウンタ 1 1 1 に書き込む（ステップ S 3 1 7 ）。この転送後、第 2 のメモリ 1 0 7 に格納されている符号化データを破棄し、第 3 カウンタの値をゼロクリアする（ステップ S 3 2 0 ）。そして、第 1 の符号化部 1 0 2 に対して新たに量子化マトリクステーブル Q 1 を設定し、第 2 の符号化部 1 0 3 に対しても新たに量子化マトリクステーブル Q 2 を設定する（ステップ S 3 0 9 ）。

20

【 0 1 4 5 】

図 17 は、このステップ S 3 0 9 の直後の各メモリの格納状況を示している。処理はステップ S 3 1 1 にて、第 1 の符号化部 1 0 2 、第 3 の符号化部 1 1 4 、第 2 の符号化部 1 0 3 の処理を継続させることになる。つまり、第 1 の符号化部 1 0 2 は、新に設定された量子化マトリクステーブル Q 1 で符号化を続行しすることになる。また、第 2 のメモリ 1 0 7 には、目標データ量をオーバーしたと判定された以降に入力される画像データの符号化データが格納されるものの、それ以前の符号化データは存在しないことになる。

30

【 0 1 4 6 】

そこで、第 3 のメモリ 1 1 8 に格納されている図 16 の領域 I" の符号化データ（可逆符号化データ）について再符号化部 1 1 2 で再符号化を開始させる。再符号化部 1 1 2 の構成は第 1 の実施形態で説明したものと同じであるが、設定する量子化マトリクステーブルは第 3 の符号化部 1 1 4 にセットした量子化マトリクステーブルと同じ Q 2 にする。再符号化処理で得られた符号化データは、再び第 3 のメモリ 1 1 8 に格納する。

【 0 1 4 7 】

図 18 は、再符号化処理が完了したと判定した直後の各メモリの格納状態を示している。図示のように、第 3 のメモリ 1 1 8 には、同図の領域 I" に相当する再符号化データ（図示の領域 I' ）が格納されている。再符号化処理には、多少の時間が必要であるため、図示のように領域 I I 、 I I' 、 I I" のように符号化データが新に格納されることになる。

40

【 0 1 4 8 】

さて、再符号化が完了すると、第 3 のメモリ 1 1 8 に格納された領域 I' の符号化データを、第 2 のメモリ 1 0 7 に転送すると共に、再符号化の符号データ量を示す第 2 のカウンタ 1 1 3 の値を、第 3 のカウンタ 1 1 6 に足し込む（ステップ S 3 2 1 ）、そして、ステップ S 8 0 1 以降の符号化シーケンスに戻ることになる。図 19 は、符号化シーケンスに戻り、ある程度の符号化が進んだ状態を示している（図示の領域 I I I 、 I I I' 、 I I I" が増えている）。図 19 において、第 2 のメモリ 1 0 7 内の領域 I' と I I' との間に空き領域が存在するが、メモリ 1 0 7 内での転送を行うことで空き領域がないように

50

しても構わない。但し、第2のメモリ107は、補助的なものである点、及び、次回、再び目標データ量を越えた場合には、図示の領域I'、II'、III'の符号化データを第1のメモリ105にその順番に転送すればよいので、問題は発生しない。

【0149】

以上変形例3を説明した。本変形例3によれば、第1のメモリ105内の符号化データ量が目標データ量をオーバーした場合に、オーバーする以前については、第2のメモリ107に格納された1段階高い圧縮率の高い符号化データそのまま使うことにより、第1のカウンタ111が不明な期間は実質的に無い、或いは短くでき、且つ、再符号化処理は、再び目標データ量をオーバーする以前であれば良いので、再符号化部112の処理は第1の実施形態ほど高速である必要もない。更に、図9のタイミングT5の直前に、第1のカウンタ111の値が目標データ量をオーバーし、タイミングT5になったとしても、第2のメモリ107から第1のメモリ105への転送が完了していれば、それ以降の処理、すなわち再符号化処理は不要であり、すぐさま次のページの符号化が行えるようになる(図15のステップS1201がYESの場合に相当する)。

【0150】

なお、上記変形例3では、再符号化部112での再符号化結果を第3のメモリ118に格納したが、変形例1と同様、再符号化結果を第2のメモリ107に格納するようにしてもよい。

【0151】

以上、符号化処理部の3つの変形例を説明したが、それぞれにおいて、補正処理部120が同様の処理を行うことで、第1の実施形態と同様の作用効果を奏することができるものは明らかである。

【0152】

<第2の実施形態>

先に説明した第1の実施形態における補正処理部120は、孤立した可逆符号化データを無条件に非可逆符号化データに変換するものであった。

【0153】

ここで、注目したい点は、実施形態では、可逆符号化としてJPEG-LSを採用していることである。その関係から、孤立可逆符号化データの符号量が、所定閾値以下の符号化データ量である場合、その画素ブロックは文字線画等である確率が非常に高いことを意味している。このような状況は、自然画中に文字(撮影者の名前、日時)等が合成されている場合を考えると分かりやすい。

【0154】

このような状況にある場合、孤立可逆符号化データをJPEG符号化で非可逆符号化データに変換してしまうと、文字線画としての先鋭度が失われ、逆に画質が劣化してしまうことになりかねない。そこで、本第2の実施形態では、孤立可逆符号化データを条件付きで非可逆符号化データに変換する例を説明する。

【0155】

符号化処理部6のブロック構成は、図1、図2、図14のいずれでも構わない。従って、以下では、補正処理部120の処理についてのみ説明することとする。

【0156】

図25は、第2の実施形態における補正処理部120の処理手順を示すフローチャートである。

【0157】

図24と異なる点は、ステップS17が追加された点であり、それ以外は図24と同じである。すなわち、図示のステップS17を追加することで、孤立している可逆符号化データであって、その符号化データ量が予め設定された閾値以下の場合には変換を行わず、閾値を越えた場合にのみ、非可逆符号化データに変換するものである。

【0158】

以上の結果、第1の実施形態に加えて、自然画中に文字線画等が存在した場合には、そ

10

20

30

40

50

の部分の先鋭度が維持され、視認性を維持するという作用効果を奏することが可能になる。

【0159】

＜第3の実施形態＞

上記第2の実施形態では、条件つきで、孤立可逆符号化データを非可逆符号化データに変換するものであった。本第3の実施形態では、孤立非可逆符号化データについても考慮する例を説明する。

【0160】

図26は第3の実施形態における符号化処理部6の構成を示している。図1と異なる点は、補正処理部120が第2のメモリ107をも参照対象とする点である。なお、本第3の実施形態では、図1をその基本としているが、図2、図14をその基礎にしても構わない。要するに、補正処理部120が、可逆符号化データを格納しているメモリ（図2では第2のメモリ107、図14では第3のメモリ118）をアクセスできるようにしていればよい。

10

【0161】

先の第2の実施形態で説明したように、自然画中に文字線画が存在することは、良くあることであるが、その逆の状態、すなわち、符号化単位である画素ブロック（実施形態では高々8×8画素サイズ）のサイズの自然画が、文字線画中に孤立して存在することは、普通はあり得ない。

【0162】

20

したがって、このような孤立非可逆符号化データは、無条件に可逆符号化データに変換しても構ないことになる。

【0163】

図27は、第3の実施形態における補正処理部120の処理内容を示すフローチャートである。図25（第2の実施形態）との違いは、ステップS18が追加された点のみである。

【0164】

ステップS13にて、可逆符号化データではない、すなわち、孤立した非可逆符号化データであると判定した場合、処理はステップS18に進む。ステップS18では、注目している画素ブロックと同一位置にある可逆符号化データを第2のメモリ107より読み出し、その読み出した符号化データを、変換結果としてステップS15で出力する。

30

【0165】

以上の結果、第3の実施形態によれば、第2の実施形態の作用効果に加えて、何らかの原因で孤立した非可逆符号化データを、可逆符号化データに置換えて出力することで、ブロックノイズを抑制した画像を形成することが可能になる。

【0166】

以上、本発明に係る第1乃至第3の実施形態を説明したが、第1のメモリ105、第2のメモリ107、及び、第3のメモリ118は物理的に別のメモリであるとして説明をしてきた。本発明においてはこれらのメモリを独立して設けることは十分に1つの特徴となる。しかしながら、これらのメモリを物理的に別のメモリとしない場合も本発明の範疇に含まれる。特に、メモリの転送スピードが、十分に高速な場合には物理的に1つのメモリ上に、第1のメモリ、第2のメモリ（及び第3のメモリ）に相当する2つ（或いは3つ）の領域を確保して、第1のメモリを第1のメモリ領域、第2のメモリを第2のメモリ領域と言い直して、これまでの説明を読み直せば、1つのメモリでも実現できることが分かる。

40

【0167】

また、1つのメモリで上記各実施形態を実現する場合には、前記転送フェーズで説明したデータ転送処理のいくつかは不要となる。その詳細はその都度容易に想像できるので説明は省略するが、前記2つの領域を厳密に別けて使用する場合、物理的に2つのメモリを持つ時と同じようにデータ転送処理が必要であるが、2つの領域間で同じデータを共有す

50

ることになれば、データ転送処理が不要になるだけでなく記憶容量の削減も図れる。

【0168】

例えは、第2のメモリ領域で保持していた符号化データを、第1のメモリ領域へ転送する際、該符号化データが格納されている先頭アドレスとデータサイズの2つの情報を第2のメモリ制御部から第1のメモリ制御部へ転送するだけで、前記符号化データを転送したのと同じ効果が得られる。

【0169】

前記符号化データを、ファイル形式やパケット形式で格納している場合は、メモリ制御部の間で転送する情報は少し増え、該符号化データに関連する管理テーブル情報を転送する必要がある。

10

【0170】

なお、実施形態では、符号化対象を 8×8 画素ブロックサイズとして説明したが、このサイズも本発明を限定するものではない。要するに、2つ（或いはそれ以上でも良い）の異なる符号化技術を使って、同一の画像領域に対して生成された2種類の符号化データ量を比較できるようにすれば良い。例えば、JPEG符号化では $N \times M$ 画素ブロック単位に符号化を行い、JPEG-LSが $2N \times 2M$ の画素ブロックを符号化対象とし得るのであれば、4つのJPEG符号化データと1つのJPEG-LS符号化データを比較するようにも構わない。

【0171】

また、第3の実施形態では、孤立可逆符号化データを条件つきで、非可逆符号化データに変換する例を説明したが、これは望ましい形態であるものの、この条件を除いたとしても、十分な画質の画像を形成することが確認されているので、その条件をつけることが本発明の必須とはならない。すなわち、図28に示すように、ステップS17を除外しても構わない。

20

【0172】

さらにまた、実施形態では、孤立符号化データとは、注目している1画素ブロックの符号化データの種類（可逆か非可逆かの種類）が、回りの8つの符号化データの種類と異なることを前提にして説明した。しかしながら、同じ種類の符号化データが1以上N以下で連続している場合にまで拡張しても構わない。この場合のNは、読み取り解像度にも依存して決定すれば良いが、微小領域を示すことに変わりなく、最大でも1桁台にすることが望ましい。

30

【0173】

また、実施形態では、2種類の符号化技術として、非可逆符号化であるJPEG、可逆符号化であるJPEG-LSを用いる例を説明した。この2つの符号化技術は、可逆、非可逆の違いがあると同時に、JPEGは自然画に適した符号化技術であり、JPEG-LSは文字線画やコンピュータグラフィックスに適したものであり、符号化効率の面でも相異なるものである。このように、可逆符号化／非可逆符号化、及び、文字線画／自然画の両方に対して相異なる性質を持つ2つの符号化技術を用いると、本願発明は有利に作用することがわかるであろう。

【0174】

40

また、上記実施形態では、図22に示す複写機に適用した例を説明したが、例えばパソコン用コンピュータ等の汎用情報処理装置にイメージスキャナ等の画像入力装置を接続して符号化する場合にも適用できるのは明らかである。この場合、図23（図3（又は図9）或いは図15に示す処理を含む）に係るプログラムを実行すれば良いので、本願発明はかかるコンピュータプログラムをもその範疇とするのは明らかである。また、通常、コンピュータプログラムはCD-ROM等のコンピュータ可読記憶媒体をそのコンピュータにセットし、システムにコピーもしくはインストールすることで実行可能になるわけであるから、当然、そのようなコンピュータ可読記憶媒体も本発明の範疇に含まれる。

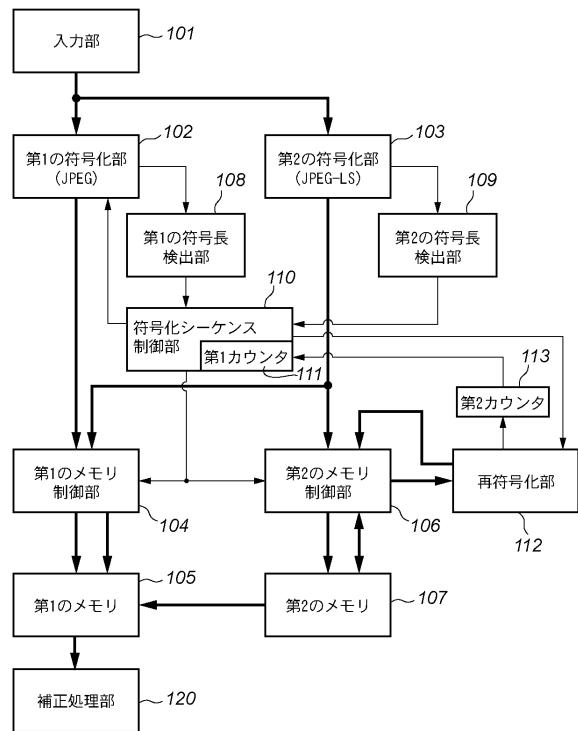
【図面の簡単な説明】

【0175】

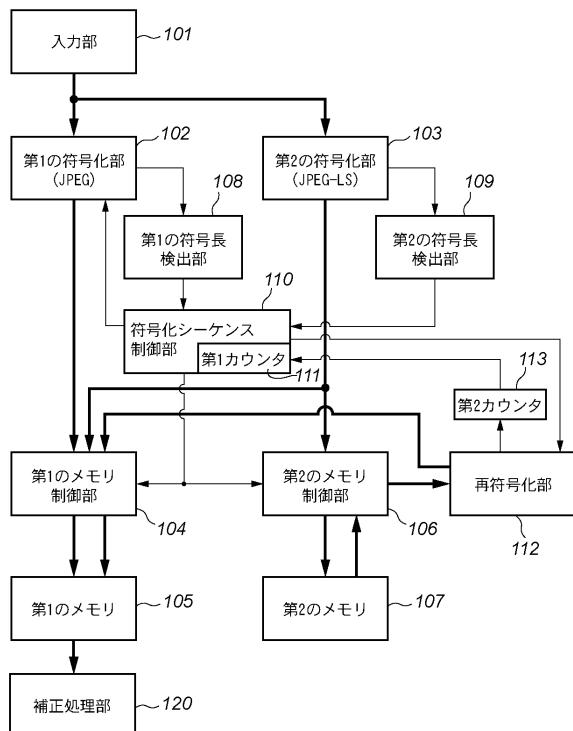
50

- 【図1】第1の実施形態における符号化処理部のブロック構成図である。
- 【図2】第1の実施形態における符号化処理部の変形例を示すブロック構成図である。
- 【図3】第1の実施形態における処理を簡略化して示したフローチャートである。
- 【図4】第1の実施形態における初期状態の符号化フェーズにおけるデータフローとメモリ内容を表わす図である。
- 【図5】第1の実施形態における符号化・再符号化フェーズの開始時のデータフローとメモリ内容を表わす図である。
- 【図6】第1の実施形態における符号化・再符号化フェーズの終了時のデータフローとメモリ内容を表わす図である。
- 【図7】第1の実施形態における転送フェーズにおけるデータフローとメモリ内容を表わす図である。 10
- 【図8】第1の実施形態における転送フェーズ後の符号化フェーズにおけるデータフローとメモリ内容を表わす図である。
- 【図9】第1の実施形態における処理の詳細を示すフローチャートである。
- 【図10】図2の構成における符号化・再符号化フェーズの開始時におけるデータフローとメモリ内容を表わす図である。
- 【図11】図2の構成における転送フェーズにおけるデータフローとメモリ内容を表わす図である。
- 【図12】図2の構成における転送フェーズ後の符号化フェーズにおけるデータフローとメモリ内容を表わす図である。 20
- 【図13】第1の実施形態における第1カウンタの値の推移の一例を示す図である。
- 【図14】第1の実施形態における符号化処理部の他の変形例のブロック構成図である。
- 【図15】図14の構成における処理手順を示すフローチャートである。
- 【図16】図14の構成における初期状態の符号化フェーズにおけるデータフローとメモリ内容を表わす図である。
- 【図17】図14の構成における目標データ量オーバー時におけるデータフローとメモリ内容を表わす図である。
- 【図18】図14の構成における符号化・再符号化フェーズの完了時におけるデータフローとメモリ内容を表わす図である。
- 【図19】図14の構成における符号化・再符号化フェーズ後の符号化フェーズにおけるデータフローとメモリ内容を表わす図である。 30
- 【図20】実施形態における再符号化部のブロック構成図である。
- 【図21】実施形態で用いる量子化マトリクステーブルを示す図である。
- 【図22】実施形態が適用する複写機の構成図である。
- 【図23】実施形態における符号化処理部の処理手順を示すフローチャートである。
- 【図24】第1の実施形態における補正処理の処理手順を示すフローチャートである。
- 【図25】第2の実施形態における補正処理の処理手順を示すフローチャートである。
- 【図26】第3の実施形態における符号化処理部のブロック構成図である。
- 【図27】第3の実施形態における補正処理の処理手順を示すフローチャートである。
- 【図28】第3の実施形態における他の補正処理の処理手順を示すフローチャートである。 40
- 。

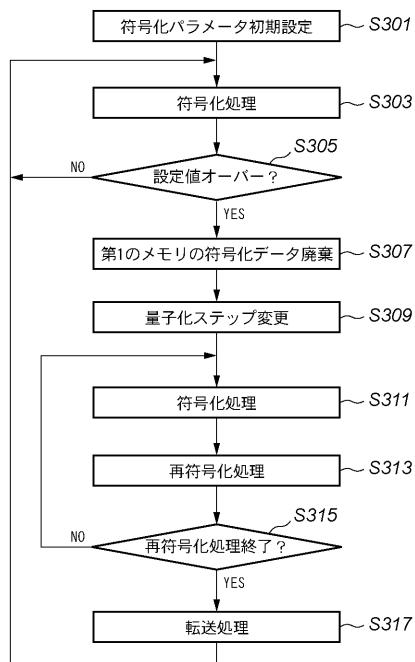
【図1】



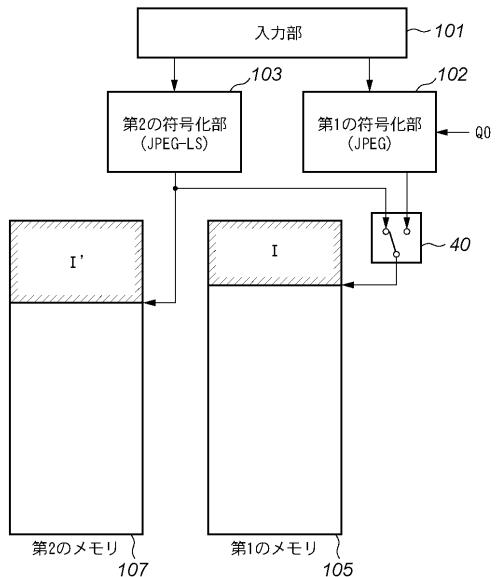
【図2】



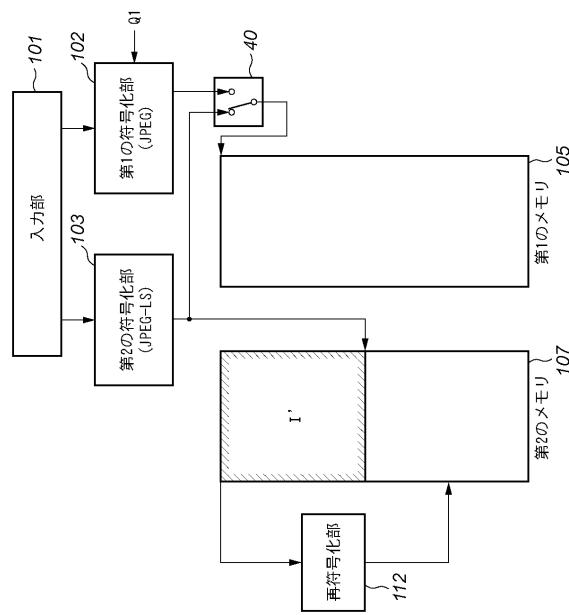
【図3】



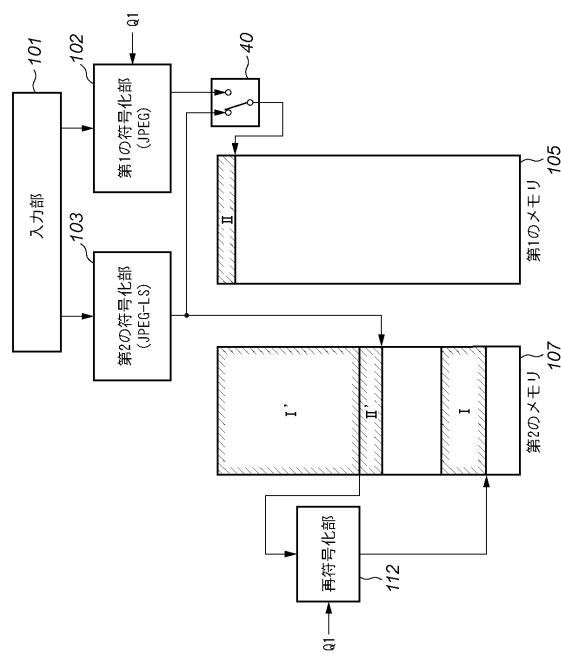
【図4】



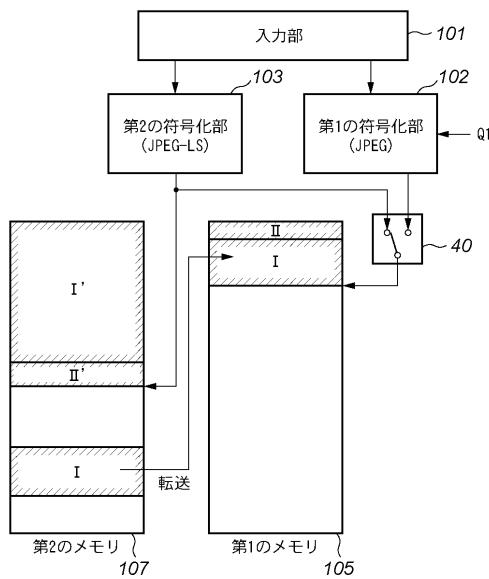
【図5】



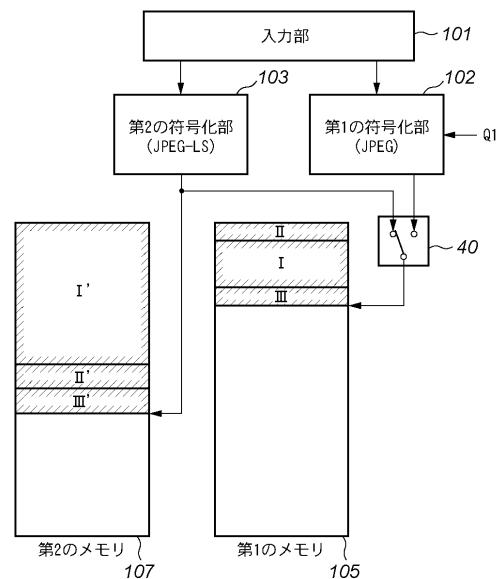
【図6】



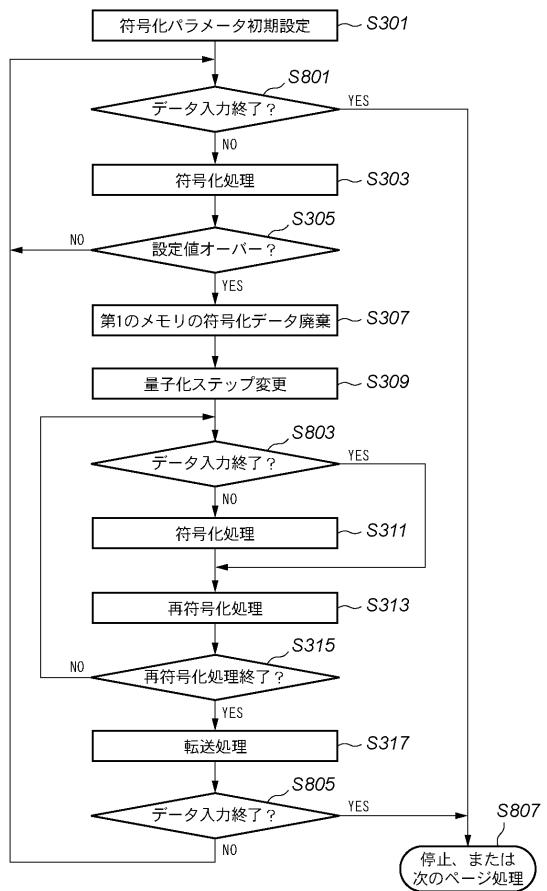
【図7】



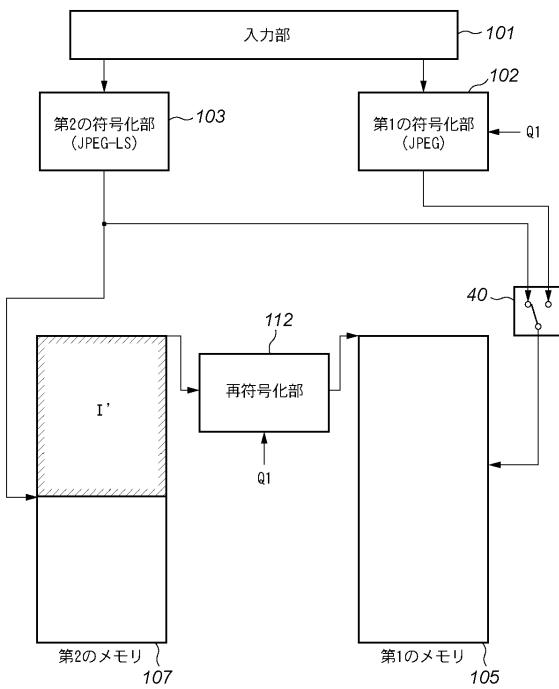
【図8】



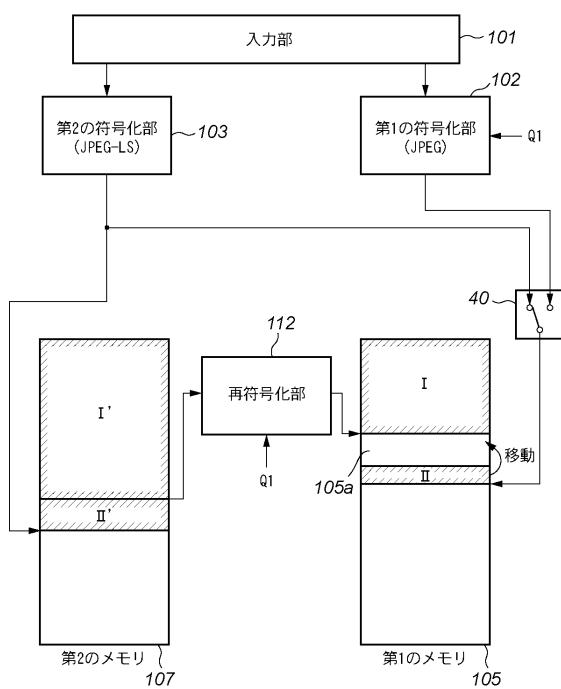
【図9】



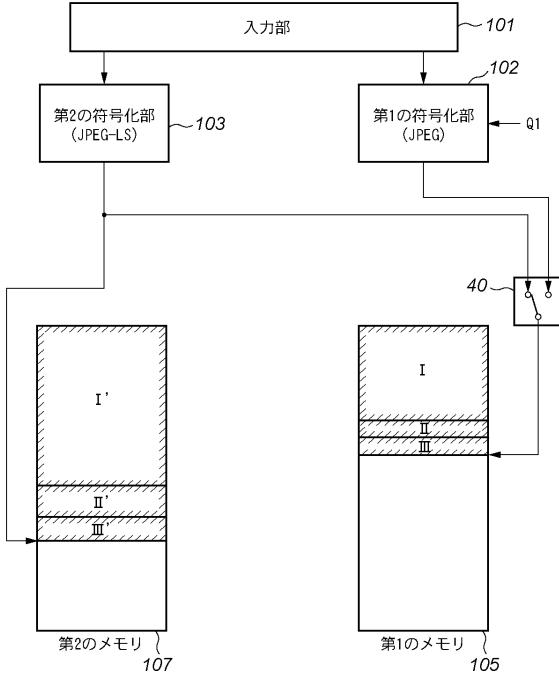
【図10】



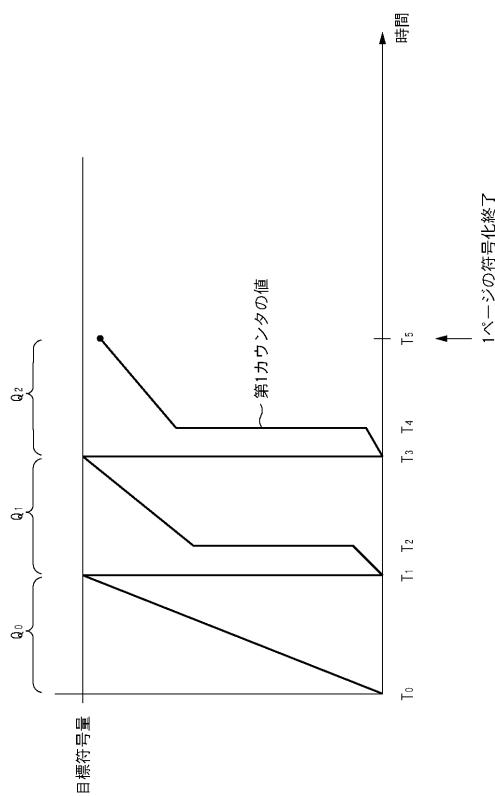
【図11】



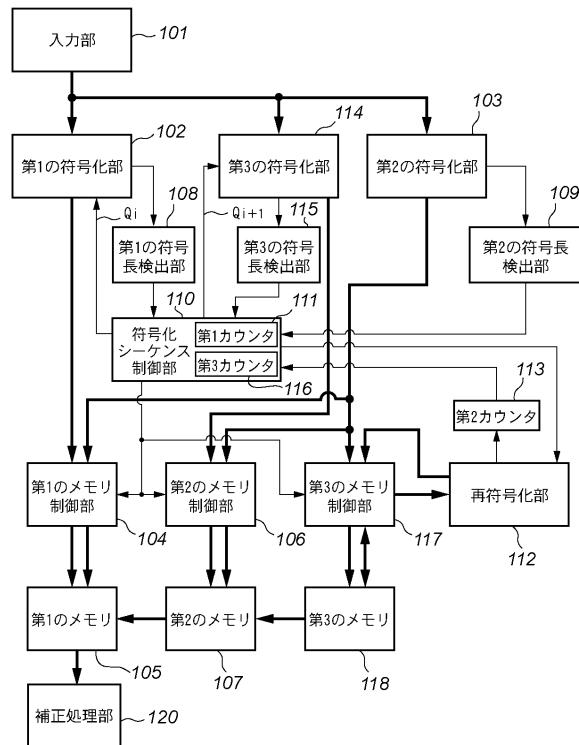
【図12】



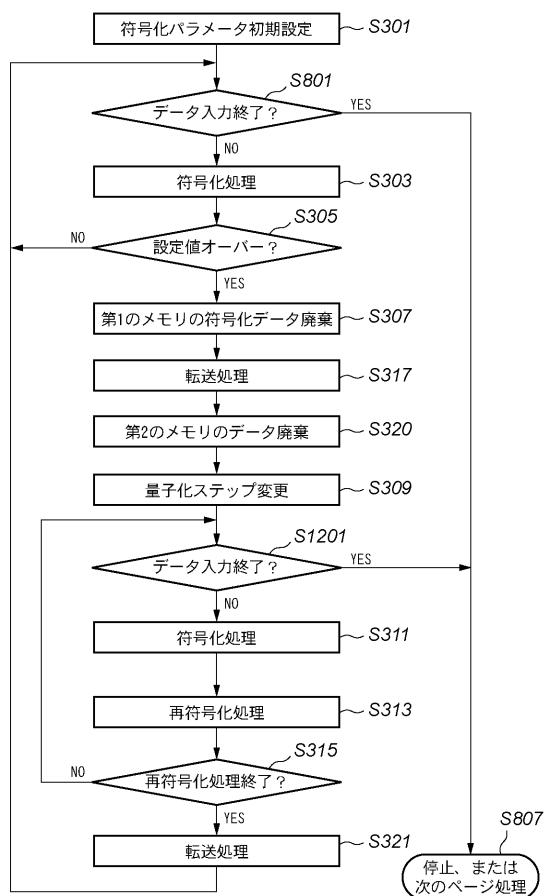
【図13】



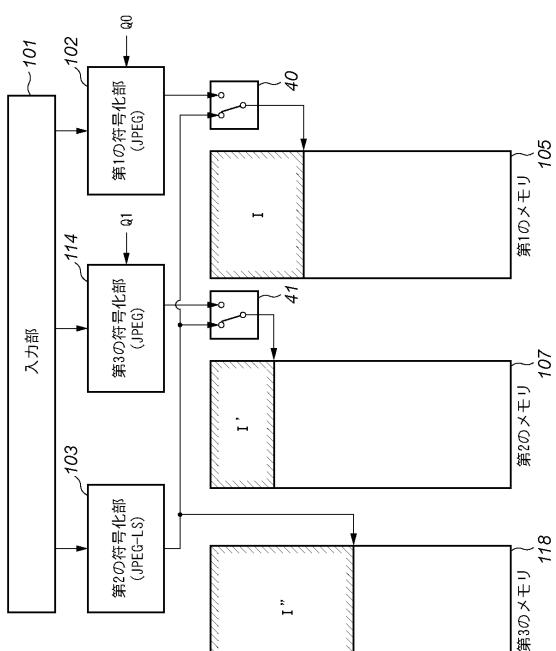
【図14】



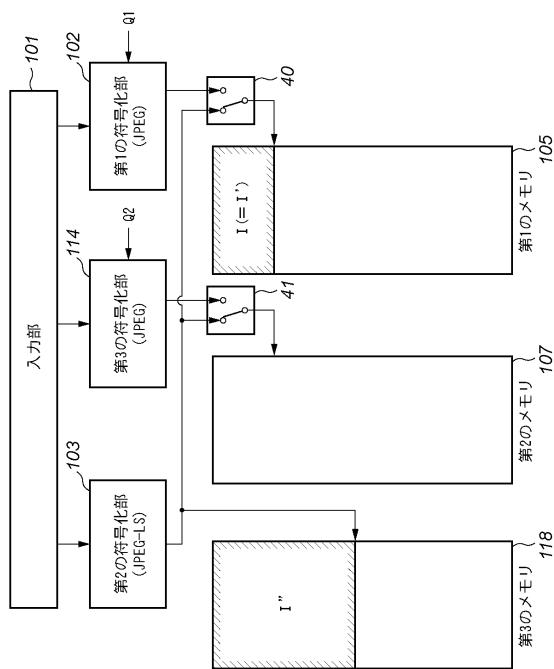
【図15】



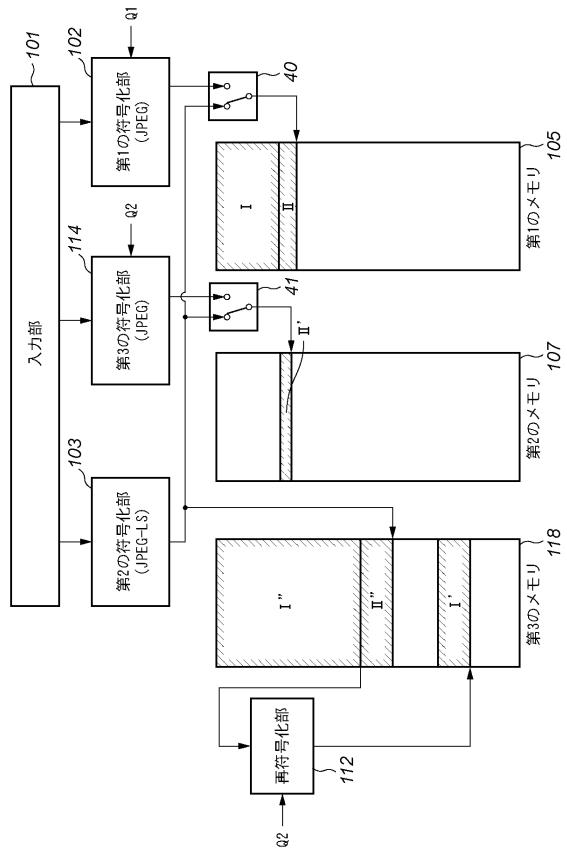
【図16】



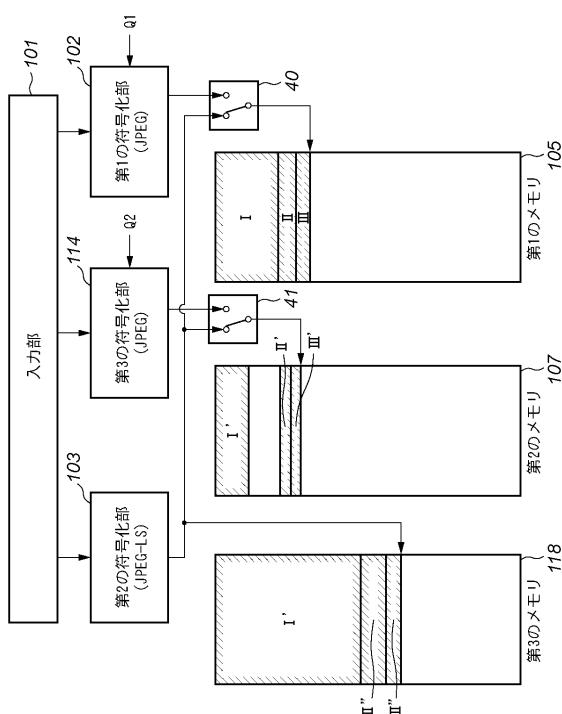
【図17】



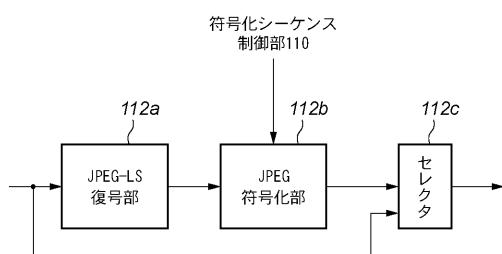
【図18】



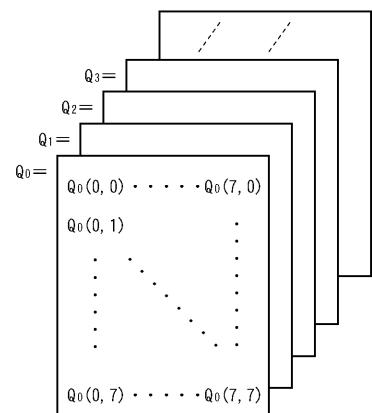
【図19】



【図20】

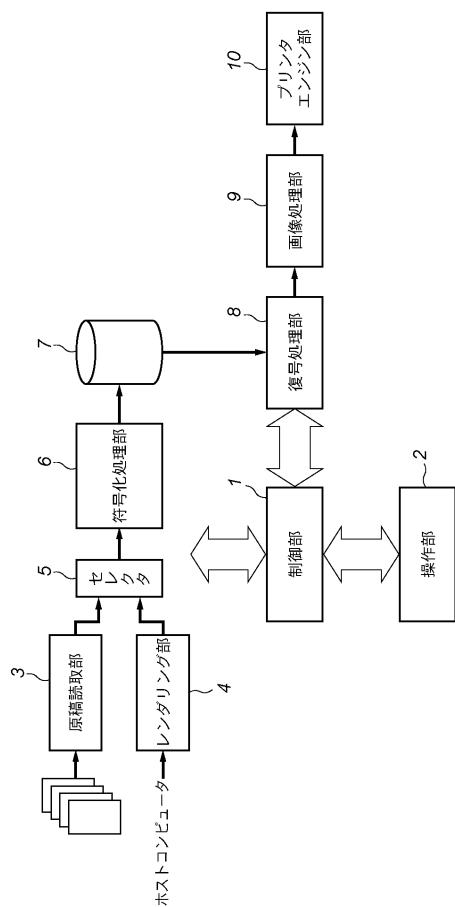


【図21】

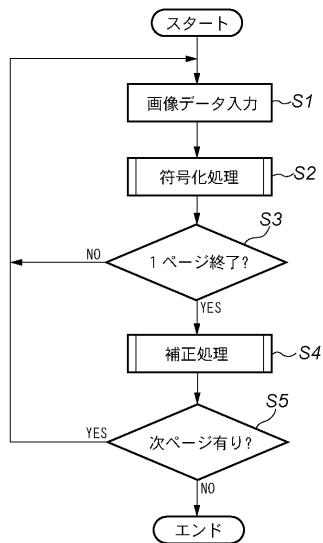


$$Q_0 < Q_1 < Q_2 < Q_3 \dots$$

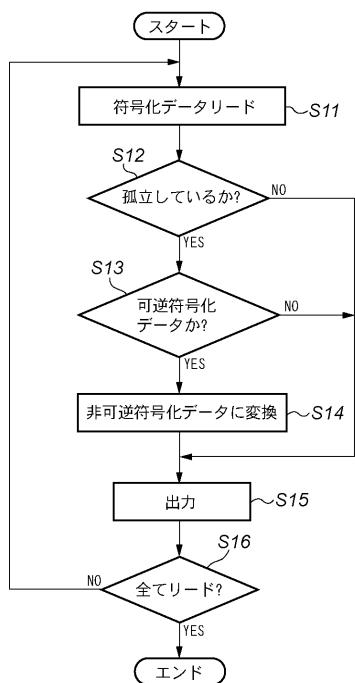
【図22】



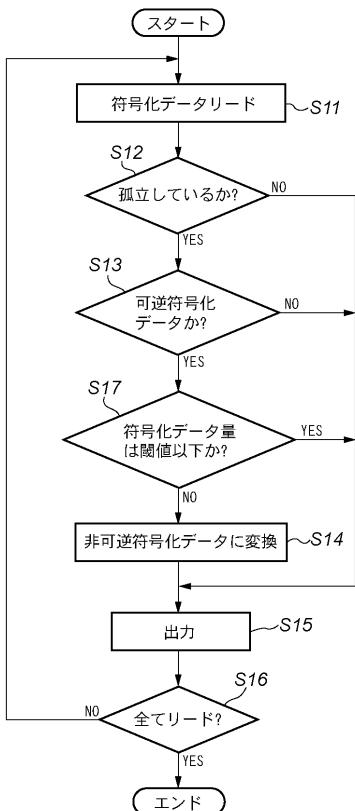
【図23】



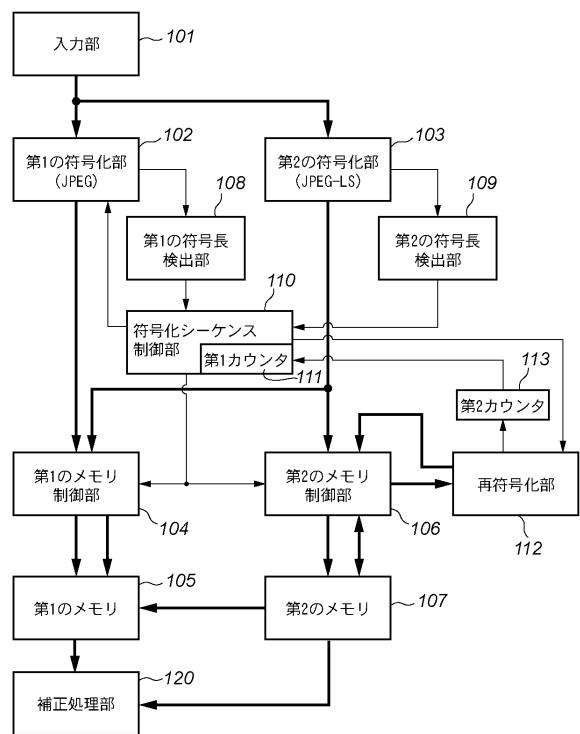
【図24】



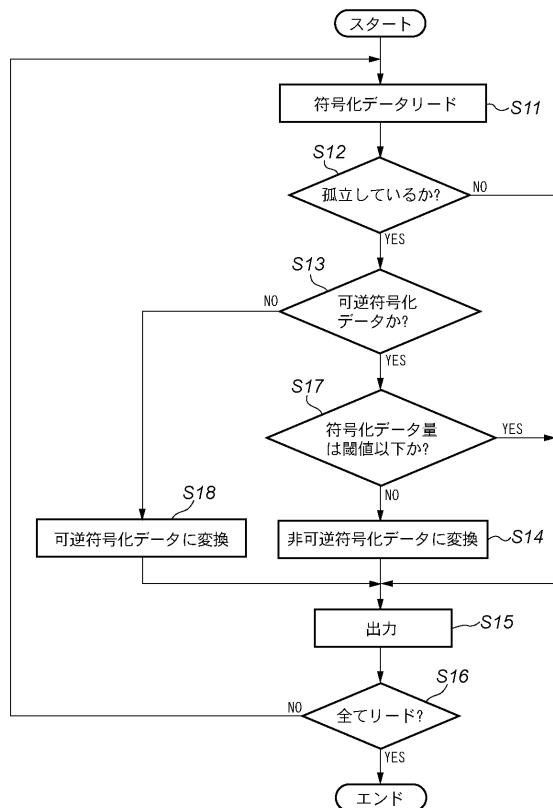
【図25】



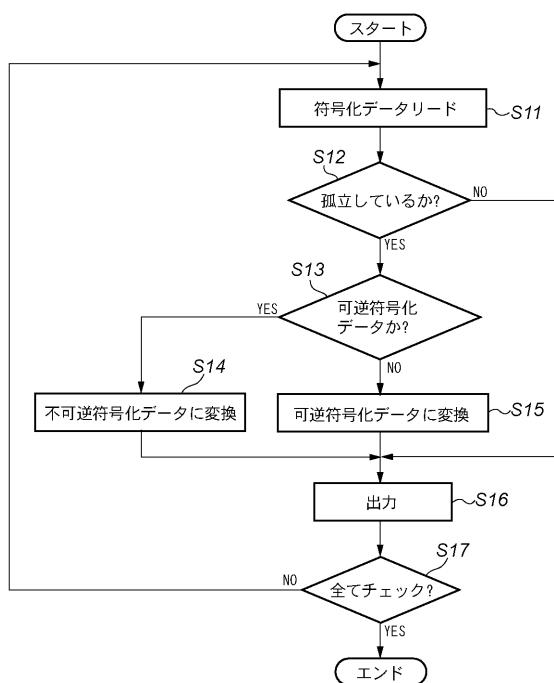
【図26】



【図27】



【図28】



フロントページの続き

(72)発明者 伊藤 直樹

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

(72)発明者 田村 宏和

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

審査官 堀井 啓明

(56)参考文献 特開2003-209697(JP, A)

国際公開第03/084243(WO, A1)

特開平07-007621(JP, A)

特開2000-350040(JP, A)

(58)調査した分野(Int.Cl., DB名)

H04N1/41-1/419

H04N7/12-7/137