

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4612797号
(P4612797)

(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.

H04N 7/32 (2006.01)

F I

H04N 7/137

Z

請求項の数 6 (全 49 頁)

(21) 出願番号 特願2004-69486 (P2004-69486)
 (22) 出願日 平成16年3月11日 (2004. 3. 11)
 (65) 公開番号 特開2005-260588 (P2005-260588A)
 (43) 公開日 平成17年9月22日 (2005. 9. 22)
 審査請求日 平成19年3月12日 (2007. 3. 12)

(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 康德
 (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
 (72) 発明者 前田 充
 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

最終頁に続く

(54) 【発明の名称】 符号化装置、符号化方法

(57) 【特許請求の範囲】

【請求項 1】

動画像を構成する各フレームの画像を動き補償によって符号化する符号化装置であって、

各フレームの画像を入力する入力手段と、

前記入力手段が入力した第1のフレームの画像から該入力手段による画像入力順に各フレームの画像を参照し、参照した第2のフレームの画像と当該第2のフレームの次のフレームである第3のフレームの画像との間の相違度が予め設定された閾値以上であると判断した場合には、前記第1のフレームから前記第2のフレームまでを1つの区間として設定することにより、前記各フレームを複数の区間に分割する区間分割手段と、

前記区間分割手段が分割したそれぞれの区間毎に、区間内の他フレームの画像群との相違度の総和値が最も小さい自フレームの画像を代表画像として設定する代表画像設定手段と、

前記それぞれの区間毎に設定した代表画像から、注目フレームの画像を符号化するために参照する代表画像を参照画像として選択する参照画像選択手段とを備え、

前記参照画像選択手段が選択した参照画像、若しくは当該参照画像を含む区間内の各フレームの画像を用いて、前記注目フレームの画像を動き補償により符号化することを特徴とする符号化装置。

【請求項 2】

動画像を構成する各フレームの画像を動き補償によって符号化する符号化装置であって

10

20

、
各フレームの画像を入力する入力手段と、
前記入力手段が入力した第1のフレームの画像から該入力手段による画像入力順に各フレームの画像を参照し、参照した第2のフレームの画像と当該第2のフレームの次のフレームである第3のフレームの画像との間の相違度が予め設定された閾値以上であると判断した場合には、前記第1のフレームから前記第2のフレームまでを1つの区間として設定することにより、前記各フレームを複数の区間に分割する区間分割手段と、

前記区間分割手段が分割したそれぞれの区間毎に、区間内の各フレームの画像を代表する代表画像を1つ設定する代表画像設定手段と、

前記それぞれの区間毎に設定した代表画像毎に、符号化対象のフレームの画像との動き補償による予測誤差を求め、当該予測誤差が最も小さくなる代表画像を、前記符号化対象のフレームの画像を符号化するために参照する参照画像として選択する参照画像選択手段とを備え、

前記参照画像選択手段が選択した参照画像、若しくは当該参照画像を含む区間内の各フレームの画像を用いて、前記符号化対象のフレームの画像を動き補償により符号化することを特徴とする符号化装置。

【請求項3】

前記相違度は、2つの画像においてそれぞれ対応する画素の画素値の差分を、当該画像を構成する全て又は一部の画素について加算した総和値であることを特徴とする請求項1又は2に記載の符号化装置。

【請求項4】

動画像を構成する各フレームの画像を動き補償によって符号化する符号化方法であって、

各フレームの画像を入力する入力工程と、

前記入力工程で入力した第1のフレームの画像から該入力工程による画像入力順に各フレームの画像を参照し、参照した第2のフレームの画像と当該第2のフレームの次のフレームである第3のフレームの画像との間の相違度が予め設定された閾値以上であると判断した場合には、前記第1のフレームから前記第2のフレームまでを1つの区間として設定することにより、前記各フレームを複数の区間に分割する区間分割工程と、

前記区間分割工程で分割したそれぞれの区間毎に、区間内の他フレームの画像群との相違度の総和値が最も小さい自フレームの画像を代表画像として設定する代表画像設定工程と、

前記それぞれの区間毎に設定した代表画像から、注目フレームの画像を符号化するために参照する代表画像を参照画像として選択する参照画像選択工程とを備え、

前記参照画像選択工程で選択した参照画像、若しくは当該参照画像を含む区間内の各フレームの画像を用いて、前記注目フレームの画像を動き補償により符号化することを特徴とする符号化方法。

【請求項5】

動画像を構成する各フレームの画像を動き補償によって符号化する符号化方法であって、

各フレームの画像を入力する入力工程と、

前記入力工程で入力した第1のフレームの画像から該入力工程による画像入力順に各フレームの画像を参照し、参照した第2のフレームの画像と当該第2のフレームの次のフレームである第3のフレームの画像との間の相違度が予め設定された閾値以上であると判断した場合には、前記第1のフレームから前記第2のフレームまでを1つの区間として設定することにより、前記各フレームを複数の区間に分割する区間分割工程と、

前記区間分割工程で分割したそれぞれの区間毎に、区間内の各フレームの画像を代表する代表画像を1つ設定する代表画像設定工程と、

前記それぞれの区間毎に設定した代表画像毎に、符号化対象のフレームの画像との動き補償による予測誤差を求め、当該予測誤差が最も小さくなる代表画像を、前記符号化対象

10

20

30

40

50

のフレームの画像を符号化するために参照する参照画像として選択する参照画像選択工程とを備え、

前記参照画像選択工程で選択した参照画像、若しくは当該参照画像を含む区間内の各フレームの画像を用いて、前記符号化対象のフレームの画像を動き補償により符号化することを特徴とする符号化方法。

【請求項 6】

コンピュータを、請求項 1 乃至 3 の何れか 1 項に記載の符号化装置が有する各手段として機能させることで、前記コンピュータを前記符号化装置として機能させるためのコンピュータプログラム。

【発明の詳細な説明】

10

【技術分野】

【0001】

本発明は、動画像を構成する各フレームを動き補償により符号化するための技術に関するものである。

【背景技術】

【0002】

近年、新しい動画像の符号化方式として H.264 符号化方式が注目されている。本符号化方式は ITU-T と ISO が共同で開発した符号化方式である。

【0003】

この新しい符号化方式の特徴は従来の MPEG-1、2、4 符号化方式と異なり、4 × 4 整数変換を用い、イントラ予測も複数用意されている。また、ループないフィルタが用いられ、動き補償も 7 種類のサブブロックで行なわれている。また、その動き補償の画素精度も MPEG-4 符号化方式と同様に 1/4 画素精度の動き補償を行なうことができる。さらにエントロピー符号化としてユニバーサル可変長符号化やコンテキスト適応可変長符号化が用いられている。

20

【0004】

さらに大きな特徴として、MPEG-1、2、4 では前後 2 枚の参照画像を用いて動き補償を行っていたが、より多くの参照画像を用いることが可能になった。ビットストリーム先のヘッダに含まれる num_ref_frames 符号は最大 16 の値をとることができる。すなわち、前後 16 フレームを参照することが可能である。符号化対象となるマクロブロックは前述のとおり、最大 16 フレームの画像に対して、7 種類のサブブロックについて 1/4 画素精度で予測誤差を求め、予測誤差が最小になるマクロブロックを選択することで符号化効率を大幅に改善することが可能となる。

30

【0005】

図 11 に H.264 の符号化器の構成を示す。この符号化器には画像データがマクロブロック単位で入力される。

【0006】

切り替え器 1000 はイントラ符号化するか否かを切り替える。イントラ符号化の場合はイントラ予測器 1001 に画像データが入力され、9 つのモードで予測を行ない、予測誤差を算出する。イントラ符号化以外の場合は画像データは差分器 1002 に入力され、予測画像との差分を求め、予測誤差とする。

40

【0007】

変換/量子化器 1003 は得られた予測誤差を 4 × 4 画素ブロックの整数変換を行ない、各係数を量子化する。量子化された結果はエントロピー符号化器 1004 で可変長符号化されて出力器 1014 に出力される。同時に量子化された結果は逆量子化/逆変換器 1005 に入力され、予測誤差を再生し、加算器 1006 で予測画像に加算する。その結果はフレームメモリ 1007 ~ 1010 に適宜格納される。

【0008】

動き推定器 1011 は、フレームメモリ 1007 ~ 1010 に格納された復号画像と入力画像を比較し、各サブブロック単位で 1/4 画素精度で動きベクトルを求める。動きベ

50

クトルと選択したフレームの番号は動き補償器 1 0 1 2 に入力され、該当するフレームメモリから参照画像を読み込み、予測誤差の最も小さいものを選択して予測画像として差分器 1 0 0 2 に出力する。また、動きベクトルと選択されたフレームの番号は動き符号化器 1 0 1 3 に入力され符号化され、動きベクトルはmvd_idx_l0符号等に、参照フレームの情報はref_idx_l0符号等になり、出力器 1 0 1 4 に出力される。出力器 1 0 1 4 は符号化データを書式に従って整形して出力する。

【 0 0 0 9 】

図 2 2 に H . 2 6 4 の復号器の構成を示す。符号化データが入力器 5 1 0 2 0 に入力されると、符号は解釈され、それぞれの復号器に分配される。エントロピー復号器 5 1 0 2 1 は可変長の復号を行ない、 4×4 の変換係数を量子化した結果を得る。量子化結果は逆量子化 / 逆変換器 5 1 0 2 2 に入力され、予測誤差を再生する。イントラ符号化の場合はイントラ予測器 5 1 0 2 3 に画像データが入力され、周囲の画素から予測を行ない、画素データを再生して出力する。イントラ符号化以外の場合は、加算器 5 1 0 2 4 で予測画像に加算し画素データを再生して出力する。同時にその結果はフレームメモリ 5 1 0 2 5 ~ 5 1 0 2 8 に適宜格納される。動き復号器 5 1 0 2 9 では動きベクトルを表すmvd_idx_l0符号等、参照フレームの情報を表すref_idx_l0符号等を復号し、動き補償器 5 1 0 3 0 に入力する。動き補償器 5 1 0 3 0 は、該当するフレームメモリから参照画像を読み込み、予測画像として加算器 5 1 0 2 4 に出力する。

【 0 0 1 0 】

このように複数フレームを参照することにより、物体の陰に 1 度なって、しばらく後に現れるような場合のように時間的に隔たったフレームを参照することで符号化効率を改善することが可能になる。

【非特許文献 1】ISO / IEC14496-10 Information technology - Coding of audio-visual objects - Part 10: Advanced video coding

【発明の開示】

【発明が解決しようとする課題】

【 0 0 1 1 】

前述の符号化装置では、複数フレームを参照して符号化するが、多くのフレームを見ることによって予測誤差を小さくしようとしている。しかしながら、参照するフレーム数が大きくなると、動き補償のための動きベクトル探索に非常に大きな処理が必要になる。

【 0 0 1 2 】

本発明は以上の問題に鑑みてなされたものであり、より高速に動き保証を行なうことを目的とする。

【 0 0 1 3 】

また、前述の符号化装置では、大きな変化がない状況では全ての参照フレームが非常に似通っており、少ないフレーム数では、長時間を参照することができず、変化が生じたときに大きな符号量を発生させることになる。例えば、線路の状態を監視しているモニタでは電車が走っていない時間が殆どであり、電車がフレームに入った瞬間に大きな符号量を発生させる。これでは複数フレームを参照しても符号化効率を向上させる効果が低くなるといった問題が生じる。また、長い時間に対応する場合は保持するフレーム数が非常に多くなり、回路規模が膨大になり、処理も多くなるため、処理時間がかかるといった問題点が生じる。

【 0 0 1 4 】

また、符号化時にノイズが多い画像を参照画像として選択すると符号量が大きくなるといった欠点がある。

【 0 0 1 5 】

また、一般的にフレーム内符号化された画像はフレーム間符号化された画像よりも劣化が小さいため、参照フレームには最適であるが、符号量が大きくなるといった欠点がある。

【 0 0 1 6 】

本発明は以上の問題に鑑みてなされたものであり、参照するフレームの数を効果的に減らし、より高速な動き補償による符号化処理を行なうことを目的とする。

【課題を解決するための手段】

【0017】

本発明の目的を達成するために、例えば本発明の符号化装置は以下の構成を備える。

【0018】

即ち、動画像を構成する各フレームの画像を動き補償によって符号化する符号化装置であって、

各フレームの画像を入力する入力手段と、

前記入力手段が入力した第1のフレームの画像から該入力手段による画像入力順に各フレームの画像を参照し、参照した第2のフレームの画像と当該第2のフレームの次のフレームである第3のフレームの画像との間の相違度が予め設定された閾値以上であると判断した場合には、前記第1のフレームから前記第2のフレームまでを1つの区間として設定することにより、前記各フレームを複数の区間に分割する区間分割手段と、

前記区間分割手段が分割したそれぞれの区間毎に、区間内の他フレームの画像群との相違度の総和値が最も小さい自フレームの画像を代表画像として設定する代表画像設定手段と、

前記それぞれの区間毎に設定した代表画像から、注目フレームの画像を符号化するために参照する代表画像を参照画像として選択する参照画像選択手段とを備え、

前記参照画像選択手段が選択した参照画像、若しくは当該参照画像を含む区間内の各フレームの画像を用いて、前記注目フレームの画像を動き補償により符号化することを特徴とする。

【発明の効果】

【0049】

本発明の構成により、参照するフレームの数を効果的に減らすことができ、より高速な動き補償による符号化処理を行なうことができる。

【発明を実施するための最良の形態】

【0050】

以下添付図面を参照して、本発明を好適な実施形態に従って詳細に説明する。

【0051】

[第1の実施形態]

図1は本実施形態に係る動画像符号化装置の構成を示す図である。本実施形態では、H.264符号化方式をベースにとって説明するが、H.264と異なり、参照フレームは時間的に不連続のものを参照することになる。但し、ベースはH.264に限定されない。また、説明を簡単にするため、過去のフレームを参照する前方向予測を例に取るが、これに限定されず、両方向予測に適用してももちろん構わない。また、以下の説明では、フレームの画像データをフレームデータと呼称する場合もある。また、このフレームデータ(フレームの画像)には、このフレームのフレーム番号を示すデータも含まれているものとする。

【0052】

図1において、19~22は外部から入力された動画像の各フレームの画像(入力画像)を格納するためのフレームメモリである。26は入力先を選択して出力するセレクトである。1はフレーム内符号化/フレーム間符号化のモードに従って出力先を選択するセレクトである。

【0053】

2はH.264符号化方式によるイントラ予測を行なうイントラ予測器であり、3は動き予測誤差を求める差分器である。4は整数型直交変換を行ない、入力された係数を量子化する変換/量子化器であり、7はその逆の作用を施す逆量子化/逆変換器である。8は動き予測画像を加算する加算器である。5は変換/量子化器4による量子化結果を符号化するエントロピー符号化器であり、6は発生した複数フレーム分の符号を格納するバッフ

アである。

【 0 0 5 4 】

9、10、11、12はフレームメモリであり、局所復号された画像データとフレームの番号をフレーム単位で格納する。13は入力画像と復号された画像データから最適な動きベクトルを該当するフレームから抽出する動き推定器であり、14は動き推定器13で算出された動きベクトルと該当するフレームの情報から予測画像を生成する動き補償器である。15は算出された動きベクトルと該当するフレームの情報から動きの情報を符号化する動き符号化器である。

【 0 0 5 5 】

17はフレームメモリ9～12の入出力を制御するフレームメモリ制御器であり、16はバッファ6に格納された符号化データを出力する出力器である。25は入力画像を比較して類似度を算出して区間を設定する区間設定器である。18は区間を代表する代表フレームを判定する代表フレーム判定器である。26は符号化を行なうフレームを選択するセクタである。27は設定された区間を格納しておく区間メモリであり、28は判定された代表フレームを格納しておく代表フレームメモリである。29は参照する区間を判定する参照フレーム判定器である。

10

【 0 0 5 6 】

上記のように構成された動画像符号化装置における動画像の符号化動作を以下、説明する。

【 0 0 5 7 】

符号化に先立ち、各フレームメモリ、区間メモリ27、代表フレームメモリ28を初期化する。初期化としては例えば各メモリ内を0のデータで埋める。

20

【 0 0 5 8 】

また、区間設定器25には、区間の先頭を表す始点値と終点を表す終点値として第1フレームを表す1を設定する。また、出力器16は参照するフレームの最大数を表すヘッダ情報を生成して出力する。

【 0 0 5 9 】

本動画像符号化装置に入力されたフレームの画像データはフレームメモリ19～23に順に格納される。また、入力された画像データはそれぞれ、直前に入力されたフレームの画像データと共に区間設定器25に入力される。すなわち、第Nフレーム目の画像と第(N-1)フレーム目の画像とがセットになって区間設定器25に入力される。

30

【 0 0 6 0 】

区間設定器25では入力された2つのフレームデータ(フレームの画像データ)の相違度Rを求める。ここでは説明を簡単にするため、2つの画像で対応する画素同士の画素値の差分の絶対値を全ての画素について加算した総和値を相違度Rとする。

【 0 0 6 1 】

例えば、フレームメモリ19に格納されている第Nフレーム目の画像 X_N の画素値を $X_N(i, j)$ (i, j は画像 X_N における画素の位置)、フレームメモリ20に格納されている第(N-1)フレーム目の画像 X_{N-1} の画素値を $X_{N-1}(i, j)$ とする。

【 0 0 6 2 】

この場合、この2つの画像(第Nフレーム目の画像と第(N-1)フレーム目の画像)との間の相違度 R_N は以下の式に従って求めることができる。

40

【 0 0 6 3 】

$$R_N = \sum (| X_N(i, j) - X_{N-1}(i, j) |) \quad (1)$$

ここで「 \sum 」は全ての i, j について加算することを示している。

【 0 0 6 4 】

そして区間設定器25は、求めた相違度 R_N が予め設定された閾値 T_h よりも小さければ終点値に1を加え、Nを1つインクリメントして同様の処理を行なう。

【 0 0 6 5 】

一方、相違度 R_N が予め設定された閾値 T_h よりも大きかった場合、または、符号化し

50

ていないフレームデータでフレームメモリ 19 ~ 22 がいっぱいになってしまった場合、区間判定器 25 は区間の始点値と終点値を区間メモリ 27 の空いた領域または、空いた領域が無ければ最も古い区間を表すデータに上書きして格納する。また、これらの値は代表フレーム判定器 18 に入力される。

【0066】

代表フレーム判定器 18 は区間を表すこれらの値（始点値、終点値）とフレームメモリ 19 ~ 22 の中でこの区間内の各フレームの画像データを読み込み、相違度 Q を求める。本実施形態では、区間の始点値を S 、終点値を E としたとき、第 N フレームとその他のフレーム（第 N フレーム目をのぞく第 S フレーム目から第 E フレーム目までの各フレーム）との相違度 Q_N は

$$Q_N = \left(\sum_{j=S}^E |X_N(i, j) - X_M(i, j)| \right) \quad (M = S \sim E) \quad (2)$$

すなわち、全ての i, j について $|X_N(i, j) - X_M(i, j)|$ の総和値を求め、更にその結果を全ての $M (S \sim E)$ について加算する。特に $N = M$ の場合は計算を省略することができる。

【0067】

すなわち本実施形態に係る相違値 Q_N は、第 N フレーム目の画像と第 M フレーム目の画像とで対応する画素同士の画素値の差分の絶対値を全ての画素について加算した総和値を全ての $M (M = S, S + 1, \dots, E - 1, E)$ について求め、その総和を計算することにより求められるものである。

【0068】

代表フレーム判定器 18 は求めた各相違度 Q_N のうち、最も小さい相違度 Q のフレームを選択する。選択したフレームの番号を代表フレームメモリ 28 に入力する。

【0069】

そして、区間設定器 25 は終点値に 1 を加えたものを始点値と終点値にセットする。

【0070】

続いて、セクタ 26 は区間メモリ 27 に最近格納した始点値、終点値を参照し、フレームメモリ 19 ~ 23 のそれぞれが格納するフレームデータにおいて始点値から終点値の間のフレームデータを順に読み出し、セクタ 1 に入力する。

【0071】

セクタ 1 は一定間隔、例えば 15 フレーム毎にフレームをフレーム内符号化するために、出力にイントラ予測器 2 を選択し、そうでない場合は、フレーム間符号化するために出力に差分器 3 を選択する。

【0072】

イントラ予測器 2 が選択された場合、入力されたフレームデータはブロック毎に予測が行なわれる。一方、差分器 3 が選択された場合は、入力された画像データはスライス単位で動き推定器 13 に入力される。

【0073】

まず代表フレームメモリ 28 に保持されている「過去に設定した各区間の代表フレーム番号」を参照して、各区間の代表フレームのデータをフレームメモリ 9 ~ 12 から読み出し、動き推定器 13、動き補償器 14 により、それぞれの代表フレームの入力画像に対する予測画像を生成する。そしてそれぞれの代表フレームの予測画像と入力画像との誤差（予測誤差）を求め、入力画像との予測誤差が最も小さい代表フレームを特定する。

【0074】

そして特定した代表フレームのフレーム番号を含む区間の両端を示す情報である始点値、終点値を代表フレームメモリ 28 から読み出し、読み出した始点値、終点値でもって規定される区間内の各フレームの画像をフレームメモリ 9 ~ 12 から読み出し、読み出した画像群と入力画像とを用いて、動き推定器 13、補償器 14 により、周知の動き補償による符号化処理を行なう。

【0075】

10

20

30

40

50

そして、入力画像に対する予測画像を動き補償器 14 により生成するので、これを差分器 3 に入力する。差分器 3 は入力画像と予測画像から予測誤差を求める。また、動きベクトルは動き符号化器 15 で符号化され、バッファ 6 に格納される。イントラ予測器 2 または差分器 3 で得られた結果は変換 / 量子化器 4 に入力され、整数型直交変換を行ない、その係数を量子化する。その結果はエントロピー符号化器 5 と逆量子化 / 逆変換器 7 に入力される。エントロピー符号化器 5 では、入力された量子化結果をエントロピー符号化し、バッファ 6 にフレーム単位で格納する。

【 0 0 7 6 】

一方、イントラ予測器 2、差分器 3 の何れにおいても予測誤差が出力されるので、この予測誤差が変換 / 量子化器 4、逆量子化 / 逆変換器 7 を介して加算器 8 で、動き補償器 14 から入力される「入力画像に対する予測画像」と加算される。例えば予測誤差が差分器 3 から出力された場合、この予測誤差とは、セクタ 26 から入力された入力画像と、この入力画像に対する予測画像との差分であるので、加算器 8 がこの差分と動き補償器 14 から入力される「入力画像に対する予測画像」と加算することにより、復号画像が復元される。よって、フレームメモリ 9 ~ 12 には、フレームメモリ 19 ~ 22 に格納されているフレームデータが復元されて格納されることになる。

【 0 0 7 7 】

以上の動き補償による符号化処理は、参照する画像（上記説明では、特定した代表フレームのフレーム番号を含む区間内の各フレームの画像）を特定する処理以外は周知の動き補償による符号化処理である。

【 0 0 7 8 】

このように本実施形態に係る符号化処理では、連続するフレーム間で画像にさほど差がない（隣接するフレーム間での相違度 R_n が閾値 T_h 以下）ようなフレーム列の区間を設定し、更にこの区間内の各フレームにおいてその他のフレームとの差が最も少ないフレームを参照画像のフレームとして設定するので、区間内の各フレームを符号化する際に参照する画像は 1 つであり、参照すべき画像の数を軽減することができ、より高速に符号化処理を行なうことができる。

【 0 0 7 9 】

なお、逆量子化 / 逆変換器 7 では、入力された量子化結果から予測誤差（入力画像と予測画像との差）を得、加算器 8 はこの予測誤差と動き補償器 14 からの予測画像とを加算することにより、入力画像を復元する。そして復元した入力画像をフレームメモリ 9 ~ 12 のうち、データが入っていないフレームメモリか、最も古いフレームの画像データのフレームメモリに格納する。

【 0 0 8 0 】

最後に、バッファ 6 に格納された符号化データは出力器 16 により外部に出力する。また、符号化が終了したフレームの画像データはフレームメモリ 19 ~ 22 のうちから消去される。

【 0 0 8 1 】

以上の処理を本動画像符号化装置に入力される全てのフレームを符号化するまで行なう。

【 0 0 8 2 】

図 2 は、以上説明した符号化処理のフローチャートである。

【 0 0 8 3 】

先ず、上述の通り、図 1 に示した各部の初期化を行ない、更に、出力器 16 は、参照するフレームの最大数を表すヘッダ情報を生成して出力する（ステップ S1）。次に、符号化すべき全てのフレームについて符号化処理を行ったか否かを判断する（ステップ S2）。符号化すべきフレームがまだ残っている場合には処理をステップ S3 に進め、符号化していないフレームの画像データでフレームメモリ 19 ~ 22 がいっぱいになってしまったか否かを判定する（ステップ S4）。フレームメモリに余裕があれば処理をステップ S4 に進め、フレームメモリに余裕がなく満杯であれば処理をステップ S8 に進める。

【 0 0 8 4 】

フレームメモリに余裕がある場合、フレームメモリ 19 ~ 22 に符号化対象のフレームデータを順に入力する（ステップ S 4）。そして区間設定器 25 により、始点値で示される位置のフレームから順に参照するのであるが、現在参照しているフレームと、この次のフレームとの相違度 R を上記式（1）に従って求める（ステップ S 5）。そして求めた相違度 R が予め決まっている閾値 T_h よりも小さいか否かを判断する（ステップ S 6）。 $R < T_h$ の場合、区間設定器 25 により、現在参照しているフレームの画像と、この次のフレームの画像とがさほど差がないと判断され、終点値に 1 を足して更新し（ステップ S 7）、ステップ S 3 以降の処理を繰り返す。このようにして、現在参照しているフレームの画像と、この次のフレームの画像とがさほど差がないと判断した場合には、終了点を 1 つずつ増やして更新していき、始点値と終点値とで規定される区間を更新する。

10

【 0 0 8 5 】

一方、ステップ S 6 において相違度 R が閾値 T_h 以上の場合、もしくはステップ S 3 においてフレームメモリに余裕がなく満杯である場合、区間設定器 25 により、現在参照しているフレームの画像と、この次のフレームの画像とに差があると判断され、始点値と終点値とで規定される区間を以降の処理で参照すべく、この始点値、終点値を区間メモリ 27 に格納する（ステップ S 8）。

【 0 0 8 6 】

そして、代表フレーム判定器 18 は、区間を表すこれらの値（始点値、終点値）とフレームメモリ 19 ~ 22 の中でこの区間内の各フレームの画像データを読み込み、この各フレームについて式（2）に従って相違度 Q を求める処理を行なう（ステップ S 9）。そして各フレームについて求めた相違度 Q のうち、最も小さい相違度 Q となったフレームを代表フレームとして設定するため、最も小さい相違度 Q となったフレームのフレーム番号を代表フレームメモリ 28 に入力し、格納する（ステップ S 10）。

20

【 0 0 8 7 】

そして以降の処理により、この区間内の各フレームについて動きベクトルを求め、符号化する処理を行なう。

【 0 0 8 8 】

まず、区間内の各フレームについて符号化を行ったか否かを判断し（ステップ S 11）、行っていない場合には処理をステップ S 12 に進め、まだ符号化していないフレームの画像を 1 つずつ符号化する。

30

【 0 0 8 9 】

以下に 1 つのフレームの画像を符号化するための処理について説明する。

【 0 0 9 0 】

まず、1 つのフレームの画像は符号化単位（ここではスライスと呼称する）毎に符号化を行なうのであるが、1 つの画像の全ての符号化単位について符号化を行ったか否かを判断する（ステップ S 12）。

【 0 0 9 1 】

まだ符号化を行っていない符号化単位があった場合には、処理をステップ S 13 に進め、動き補償器 14 は、代表フレームメモリ 28 に保持されている「過去に設定した各区間毎の代表フレーム番号」に基づいて、各区間の代表フレームのデータをフレームメモリ 9 ~ 12 から読み出し、それぞれの代表フレームの符号化対象フレームの画像に対する予測画像を生成し、それぞれの代表フレームに対する予測画像と符号化対象フレームの画像との誤差（予測誤差）を求める（ステップ S 13）。

40

【 0 0 9 2 】

そして、符号化対象フレームとの予測誤差が最も小さい代表フレームを特定し、特定した代表フレームのフレーム番号を含む区間の両端を示す情報である始点値、終点値を代表フレームメモリ 28 から読み出すことで、符号化しようとする符号化単位を符号化するために参照する区間を設定することができる（ステップ S 14）。

【 0 0 9 3 】

50

そしてステップS 1 4で設定した区間内の各フレームの画像をフレームメモリ1 9 ~ 2 2から読み出し、読み出した画像群と符号化対象の符号化単位とを用いて、この符号化単位に対して符号化処理を行なう(ステップS 1 5)。

【0 0 9 4】

ステップS 1 5における処理をより詳細に説明すると、フレーム間符号化のモードが設定されている場合には、動き推定器1 3は、まだ符号化を行っていない符号化単位について、ステップS 1 4で設定した区間内の各フレームの画像の該当部分と比較して周知の技術により動きベクトルを求め、動き符号化器1 5はこの動きベクトルを符号化してその符号化結果をバッファ6に出力する。

【0 0 9 5】

また、求めた動きベクトルから動き補償器1 4はこの符号化単位に対する予測画像を生成し、差分器3はこの予測画像と符号化単位との差分を予測誤差として求める。そしてこの予測誤差を変換/量子化器4、エントロピー符号化器5により符号化してバッファ6に出力する。これにより、バッファ6には、この符号化単位に対する符号化結果として、動きベクトルの符号化データ、予測誤差の符号化データが格納されることになる。

【0 0 9 6】

一方、フレーム内符号化のモードが設定されている場合には、この符号化単位をそのまま符号化する。

【0 0 9 7】

そして次の符号化単位に対する符号化処理を行なうべく、処理をステップS 1 2に戻し、以降の処理を繰り返す。以上のステップS 1 2 ~ S 1 5の符号化処理は周知のものであるが、本実施形態では、符号化時に参照する画像の選択方法に以上のような特徴がある。

【0 0 9 8】

一方、ステップS 1 2において、1つのフレームの画像を構成する符号化単位全てについて符号化処理を行った場合には処理をステップS 1 6に進め、ステップS 1 2 ~ S 1 5で符号化した対象のフレームの画像をフレームメモリ9 ~ 1 2の内で画像データを未格納のフレームメモリ、またはフレームメモリ9 ~ 1 2の内で最も古い番号のフレームを格納しているフレームメモリに格納する(ステップS 1 6)。

【0 0 9 9】

そして出力器1 6はバッファ6に出力された符号化データを外部に出力し(ステップS 1 7)、区間内の次のフレームを符号化すべく、処理をステップS 1 1に戻し、以降の処理を繰り返す。

【0 1 0 0】

そしてステップS 1 1で区間内の全てのフレームについて符号化を行った場合には処理をステップS 1 8に進め、区間設定器2 5は終点値に1を加えたものを始点値と終点値にセットすることで始点値、終点値を更新し(ステップS 1 8)、処理をステップS 2に戻し、以降の処理を繰り返す。

【0 1 0 1】

以上の説明により、本実施形態に係る動画像符号化装置は、より少ないフレーム数の参照で効率よく最適な動き補償を行なうことが可能になり、高速に符号化することが可能になるといった効果がある。なお、本実施形態では相違度は画像間での画素毎の差分であったが、画像間の類似度を示すものであれば、特に限定するものではない。

【0 1 0 2】

また、本実施形態では画素毎の差分をフレームの全ての画素について行なったが、サブサンプリングした画素に対して行なっても構わない。

【0 1 0 3】

また本実施形態に係る動画像符号化装置は、図1に示した構成を有するハードウェアとしたが、これに限定するものではなく、他にも例えば、図1に示した各部の機能をソフトウェアでもって実装し、このプログラムをPC(パーソナルコンピュータ)やWS(ワークステーション)などのコンピュータにインストールし、このコンピュータのCPUがこ

10

20

30

40

50

れを実行することにより、このコンピュータは上述した動画像符号化処理を実行することになる。よってこの場合、このコンピュータをもってして本実施形態に係る動画像符号化装置とすることができる。

【 0 1 0 4 】

このように、ソフトウェアでもって実装可能であるのは本実施形態だけではなく、以降の実施形態においても同様である。

【 0 1 0 5 】

[第 2 の実施形態]

図 3 は本実施形態に係る動画像符号化装置の構成を示す図である。同図において図 1 と同じ部分については同じ番号をつけており、その説明を省略する。

10

【 0 1 0 6 】

1 1 7 はフレームメモリ 9 ~ 1 2 の入出力を制御するフレームメモリ制御器であり、1 2 5 は入力画像を比較して類似度を算出して区間を設定する区間設定器である。1 1 8 は区間を代表する代表フレームを判定する代表フレーム判定器である。

【 0 1 0 7 】

以下に本実施形態に係る動画像符号化装置が行なう動画像符号化処理について説明する。

【 0 1 0 8 】

第 1 の実施形態と同様に、符号化に先立ち、各フレームメモリ、区間メモリ 2 7、代表フレームメモリ 2 8 を初期化する。初期化としては例えば各メモリ内を 0 のデータで埋める。

20

【 0 1 0 9 】

また、区間設定器 1 2 5 には、区間の先頭を表す始点値と終点を表す終点値として第 1 フレームを表す 1 を設定する。また、出力器 1 6 は参照するフレームの最大数を表すヘッダ情報を生成して出力する。

【 0 1 1 0 】

動画像を構成する各フレームは順次フレームメモリ 1 9 に入力される。セレクタ 1 は、フレーム内符号化を行なうモードが設定されている場合には、フレームメモリ 1 9 に入力され、格納されているフレームデータをマクロブロック単位で順次イントラ予測器 2 に入力する。そしてイントラ予測器 2 はブロック毎に予測を行ない、予測結果である予測誤差は変換 / 量子化器 4 で整数型直交変換と量子化が行なわれ、その結果はエントロピー符号化器 5 に入力される。エントロピー符号化器 5 では、入力された量子化結果をエントロピー符号化し、バッファ 6 にフレーム単位で格納する。

30

【 0 1 1 1 】

逆量子化 / 逆変換器 7 では、入力された量子化結果から復号画像を得る。フレームメモリ制御器 1 1 7 の指示に従って、フレームメモリ 9 ~ 1 2 のうち、空いているフレームメモリ、または最も古い代表フレームの画像データのフレームメモリに復号画像データを格納する。バッファ 6 に格納された符号化データは出力器 1 6 から出力される。

【 0 1 1 2 】

このように、フレーム内符号化を行なう場合については第 1 の実施形態と同様である。

40

【 0 1 1 3 】

一方、フレーム間符号化のモードが設定されている場合について説明する。

【 0 1 1 4 】

基本的には第 1 の実施形態と同様にして符号化処理を行なうのであるが、第 1 の実施形態と異なるのは、予め区間、及び代表フレームを決めておくのではなく、符号化を行っていくのと並行して、この区間、代表フレームを適宜更新する点にある。以下ではその更新処理について説明する。

【 0 1 1 5 】

第 1 の実施形態と同様に、差分器 3 は、入力画像と予測画像との差分を予測誤差として出力するのであるが、この予測誤差は加算器 8 でこの予測画像に加算され、復号画像を復

50

元し、この復号画像をフレームメモリ9～12のいずれかに格納される。

【0116】

そしてフレームメモリ9～12に格納され、入力された画像データはそれぞれ、直前に入力されたフレームの画像データと共に区間設定器125に入力される。すなわち、第Nフレーム目の画像と第(N-1)フレーム目の画像とがセットになって区間設定器125に入力される。

【0117】

区間設定器125では入力された2つのフレームデータの相違度Rを式(1)に従って求める。なお、本実施形態においても、相違度Rの求め方はこれに限定するものではない。そして区間設定器125は、求めた相違度Rが予め設定された閾値 T_h よりも小さければ終点値に1を加え、終点値を更新する。

10

【0118】

次に、区間設定器125により更新後の終点値が示すフレームが第Nフレーム目である場合、代表フレーム判定器116は、現在始点値が示すフレームから第(N-1)フレームまでの各フレームの画像と第Nフレーム目の画像とを用いて、式(2)に従って、相違度 Q_N を求める(その場合、「現在始点値が示すフレームから第(N-1)フレームまでの各フレームの画像」が式(2)における $X_M(i, j)$ に相当し、「第Nフレーム目の画像」が $X_N(i, j)$ に相当することになる)。

【0119】

また、区間設定器125は、第Nフレーム目の画像と、現在の区間の代表フレームとして設定されているフレームの画像データとの相違度 S_N を式(1)に従って求める。

20

【0120】

そして代表フレーム判定器29は、相違度 S_N と第(N-1)フレームまでを符号化した際の代表フレームとの相違度の累積 Q_D (相違度 Q_D の初期値は0)と加算し、新しい累積相違度 Q_D' を算出する。この累積相違度 Q_D' と相違度 Q_N との大小比較を行なう。

【0121】

比較の結果、相違度 Q_N が累積相違度 Q_D' よりも小さい場合、第Nフレーム目を新しい代表フレームとするため、第Nフレームのフレーム番号(すなわちN)を代表フレームメモリ28に設定する。その後、相違度 Q_D の値として相違度 Q_N の値を代入する。

30

【0122】

一方、相違度 Q_N が相違度 Q_D' 以上であれば、代表フレームの更新は行わず、その後、相違度 Q_D の値として相違度 Q_D' の値を代入する。

【0123】

以上の処理により、代表フレーム、及び区間の終点値を、次のフレームの符号化の直前に設定することができる。

【0124】

図4は、以上説明した符号化処理のフローチャートである。同図において図2と同じ処理ステップについては同じステップ番号をつけており、説明を簡略化する。また本実施形態では、フレーム単位で代表フレームを選択する例をとって説明するが、これに限定されず、第1の実施形態のようにスライスやマクロブロックを単位にしてももちろん構わない。

40

【0125】

先ず、上述の通り、図3に示した各部の初期化を行ない、更に、出力器16は、参照するフレームの最大数を表すヘッダ情報を生成して出力する(ステップS1)。次に、符号化すべき全てのフレームについて符号化処理を行ったか否かを判断する(ステップS2)。符号化すべきフレームがまだ残っている場合には処理をステップS4に進め、フレームメモリ19に符号化対象のフレームデータを順に入力する(ステップS4)。

【0126】

次に、フレーム内符号化、フレーム間符号化の何れのモードが設定されているのかを判

50

断する（ステップS 1 0 1）。ここでフレーム内符号化を行なう場合には処理をステップS 1 0 2に進め、セクタ1はフレームメモリ19からの入力画像をイントラ予測器2に出力し、イントラ予測器2は入力された画像に対して周知のイントラ符号化を行なう（ステップS 1 0 2）。

【0127】

一方、フレーム間符号化を行なう場合には処理をステップS 1 0 4に進め、動き補償器14は、代表フレームメモリ28に保持されている「過去に設定した各区間毎の代表フレーム番号」に基づいて、各区間毎の代表フレームのデータをフレームメモリ9～12から読み出し、それぞれの代表フレームの符号化対象フレームに対する予測画像を生成し、そしてそれぞれの代表フレームの予測画像と符号化対象フレームの画像との誤差（予測誤差）を求める（ステップS 1 0 4）。 10

【0128】

そして、符号化対象フレームとの予測誤差が最も小さい代表フレームを特定し、特定した代表フレームのフレーム番号を含む区間の両端を示す情報である始点値、終点値を代表フレームメモリ28から読み出すことで、符号化しようとする符号化対象フレームを符号化するために参照する区間を設定することができる（ステップS 1 0 5）。

【0129】

そしてステップS 1 0 5で設定した区間内の各フレームの画像をフレームメモリ9～12から読み出し、読み出した画像群と符号化対象フレームとを用いて、この符号化対象フレームに対して周知の動き補償による符号化処理を行なう（ステップS 1 0 6）。 20

【0130】

ステップS 1 0 4からステップS 1 0 6までの各ステップにおける処理はそれぞれ、符号化単位が異なる以外は、ステップS 1 3からステップS 1 5までの各ステップにおける処理と同じである。

【0131】

そして以上の処理により、第1の実施形態と同様に、バッファ6には符号化対象の符号化データが格納されるので、出力器16は、バッファ6に格納されたこの符号化データを外部に出力し、更に、ステップS 1 6と同様にして、符号化したフレームの復元画像をフレームメモリ19～22のうち画像データを未格納のフレームメモリ、またはフレームメモリ9～12の中で最も古い番号のフレームを格納しているフレームメモリに格納する（ステップS 1 0 7）。 30

【0132】

次に、以降の処理では、次のフレームの符号化のために参照され得る情報、すなわち、代表フレームの位置、及び区間を設定する処理を行なう。

【0133】

先ず代表フレーム判定器118は、ステップS 1 0 6で符号化されたフレームの復元画像（ステップS 1 0 7で復元）と、このフレームの直前のフレームの画像（同様に復元されてフレームメモリ9～12のいずれかに格納されている）とを比較し、式（1）に基づいて、相違度Rを求める（ステップS 1 0 8）。 40

【0134】

そして求めた相違度Rと閾値Thを比較し（ステップS 1 0 9）、相違度Rが閾値Th以上であれば処理をステップS 1 1 0に進め、そうでなければ処理をステップS 1 1 2に進める。

【0135】

相違度Rが閾値Th以上である場合、処理をステップS 1 1 0に進め、ステップS 1 0 6で符号化したフレームは直前の区間内のフレームには属さない、換言すれば、直前の区間内のどのフレームとも大きな差があるフレームであると判断するので、ステップS 1 0 6で符号化したフレームを次の新たな区間に属するフレームであると判断し、次の区間の開始点、終了点、及びこの区間における代表フレームの設定処理を行なう。

【0136】

10

20

30

40

50

まず区間設定器 125 は、区間メモリ 27 中に新規の区間の開始点、及び終了点としてステップ S 106 で符号化したフレームの番号を設定する（ステップ S 110）。次に、新たに設定する区間における代表フレームとして、ステップ S 106 で符号化したフレームとすべく、このフレームの番号を代表フレームメモリ 28 に格納する（ステップ S 111）。

【0137】

そして次のフレームの符号化処理を行なうべく、処理をステップ S 2 に戻し、以降の処理を行なう。

【0138】

一方、ステップ S 109 において相違度 R が閾値 T_h よりも小さい場合には処理をステップ S 112 に進め、現在区間メモリ 27 に格納されている終了点に 1 を加えて、ステップ S 106 で符号化したフレームを直前の区間を含める処理を行なう（ステップ S 112）。

【0139】

次に、区間設定器 125 は上記式（1）に従って上記相違度 Q_N 、 S_N を求める（ステップ S 113）。より具体的には、更新後の終点値が示すフレーム（ここではステップ S 106 で符号化したフレーム）が第 N フレーム目である場合、代表フレーム判定器 116 は、現在始点値が示すフレームから第 $(N - 1)$ フレームまでの各フレームの画像と第 N フレーム目の画像とを用いて、式（2）に従って、相違度 Q_N を求める（その場合、「現在始点値が示すフレームから第 $(N - 1)$ フレームまでの各フレームの画像」が式（2）における $X_M(i, j)$ に相当し、「第 N フレーム目の画像」が $X_N(i, j)$ に相当することになる）。

【0140】

また、第 N フレーム目の画像と、現在代表フレームとして設定されているフレームの画像データとの相違度 S_N を式（1）に従って求める。

【0141】

そして代表フレーム判定器 29 は、相違度 S_N と第 $(N - 1)$ フレームまでを符号化した際の代表フレームとの相違度の累積 Q_D （相違度 Q_D の初期値は 0）と加算し、新しい累積相違度 Q_D' を算出する（ステップ S 114）。更新後の Q_D を Q_D' と表す。

【0142】

そして、累積相違度 Q_D' と相違度 Q_N との大小比較を行なう（ステップ S 115）。

【0143】

比較の結果、累積相違度 Q_N が相違度 Q_D' よりも小さい場合、処理をステップ S 116 に進め、第 N フレーム目（すなわちステップ S 106 で符号化したフレーム）を新たな代表フレームとするため、第 N フレームのフレーム番号（すなわち N ）を代表フレームメモリ 28 に設定する（ステップ S 116）。その後、累積相違度 Q_D の値として相違度 Q_N の値を代入する（ステップ S 117）。

【0144】

一方、相違度 Q_N が累積相違度 Q_D' 以上であれば処理をステップ S 115 からステップ S 118 に進め、累積相違度 Q_D の値として累積相違度 Q_D' の値を代入する（ステップ S 118）。

【0145】

そしてステップ S 117、S 118 何れの場合であってもその後、次のフレームを符号化するために処理をステップ S 2 に戻し、以降処理を繰り返す。

【0146】

以上の処理により、次のフレームを符号化する際に参照画像として参照すべきフレーム（代表フレーム）、及び区間の終点値を、次のフレームの符号化の直前に設定することができる。

【0147】

このような構成と一連の選択動作により、復号画像で区間の設定と代表フレームの判定

10

20

30

40

50

を行なうため、フレームメモリとして第1の実施形態よりも少ないメモリ容量（本実施形態の場合、入力画像を保持しておくフレームメモリは第1の実施形態が複数なのに比べて単数である）でも高速に動き補償を行なうことが可能になるといった効果がある。

【0148】

〔第3の実施形態〕

図5は、本実施形態に係る動画像符号化装置の基本構成を示すブロック図である。

【0149】

同図において、300はメモリ301にロードされたプログラムやデータを用いて、本装置全体の制御を行なうとともに、後述する各処理を実行するCPUである。301はメモリで、記憶装置304、305からロードされたプログラムやデータを一時的に格納する
10

【0150】

303は、本動画像符号化装置に対する各種の設定や、各種の指示などを入力するための端末装置である。304は、CPU300に各種の処理を実行させるためのソフトウェアのプログラムやデータを記憶するための記憶装置で、これらのソフトウェアの一部、もしくは全部はCPU300の制御によりメモリ301にロードされ、CPU300による
20

【0151】

305は、動画像のストリームのデータを記憶する記憶装置で、このストリームの一部、もしくは全部はCPU300の制御によりメモリ301にロードされ、CPU300による
20

【0152】

なお、同図では記憶装置304、305はそれぞれ別個の装置としているが、これに限定するものではなく、1つの記憶装置内に、記憶装置304、305それぞれが記憶するプログラムやデータを記憶させても良い。また、記憶装置304、305はそれぞれリムーバブルな記憶媒体としてもよく、その場合、本動画像符号化装置にはこれら記憶媒体に記録されたプログラムやデータを読み出す装置を付加する必要がある。

【0153】

306は、動画像を撮像するためのカメラで、撮像された動画像のデータ（各フレームの画像データ）は、記憶装置305内に記録される。307はCRTや液晶画面などにより構成される表示装置で、各種の画像や文字などを表示する。308は通信I/F（インターフェース）で、本動画像符号化装置はこの通信I/F308、そしてLAN、公衆回線、無線回線、放送電波等のネットワーク309を介して外部とデータ通信を行なう。
30

【0154】

302は、上述の各部を繋ぐバスやネットワークを含む通信路である。

【0155】

次に、上記構成を備える動画像符号化装置が行なう、カメラ306により撮像され、記憶装置305に記録された動画像を符号化し、符号化結果を通信I/F308を介してネットワーク309に送出する処理について説明する。なお、符号化方式としてH.264
40

【0156】

端末装置303に備わっている不図示のキーボードやマウスを用いて符号化指示を入力すると、この指示はCPU300に通知され、CPU300は各部の初期化を行なう。またCPU300は、CPU300に後述の動画像符号化処理を実行させるためのソフトウェア、CPU300に後述の通信処理を行わせるためのソフトウェア等のソフトウェアのプログラムやデータを記憶装置305から読み出してメモリ301にロードする。
50

【0157】

図6はメモリ301内のメモリマップの一例を示す図である。同図に示すように、メモリ301には、動画像符号化ソフトウェア、通信ソフトウェアを記憶するためのエリアが設けられているとともに、これらソフトウェアを制御するなどの処理をCPU300に実行させるためのOS（オペレーティングシステム）を記憶するためのエリアも備えている。また、後述する画像エリア（1～9）、符号エリア、ワーキングエリアも備える。

【0158】

図7は、CPU300が、この動画像符号化ソフトウェア、通信ソフトウェアを実行することによってなされる処理のフローチャートである。

【0159】

10
先ず、メモリ301上の各画像エリア、符号エリアをクリアし、ヘッダ情報を生成して符号エリアに格納する（ステップS201）。このとき、ワーキングエリア上に区間を表すテーブルの領域を確保する。テーブルの内容（構成例）を図9に示す。

【0160】

各レコードは有効フラグ、区間始点値、区間終点値、代表フレーム番号からなり、レコードの数は最大、参照画像を格納する画像エリアの個数になる。本実施形態では、画像エリア1は入力画像を格納するので参照画像は8フレームとなる。レコードxの有効フラグ、区間始点値、区間終点値、代表フレーム番号をそれぞれF(x)、S(x)、E(x)、D(x)とする。

【0161】

20
図7に戻って、次に、全てのフレームの符号化が終了したか否かの判定を行なう（ステップS202）。全てのフレームの符号化処理を行っていない場合には処理をステップS203に進め、記憶装置305から、カメラ306が撮像した動画像のうち、1フレーム分のデータを読み出して、メモリ301の画像エリアに格納する（ステップS203）。ステップS203における処理が最初の場合には、画像エリア1に格納する。なお、格納するフレームのデータには、このフレームのフレーム番号も含まれている。

【0162】

30
符号化するフレームがフレーム内符号化を行なうフレームであれば、処理をステップS214に進め、そうでなければ処理をステップS205に進め、ステップS203で画像エリアに格納したフレームの画像と、代表フレームである画像データとの動き補償を行なう（ステップS250）。

【0163】

フレーム内符号化を行なう場合、イントラ予測を行ないながらフレーム内符号化を行なう。（ステップS210）次に、通信ソフトウェアを実行することにより、メモリ301上の符号エリアの符号化データを通信インターフェース308を介して通信回路309に出力し、その後に、メモリ301上の符号エリアの該当する部分をクリアする（ステップS211）。

【0164】

図8は、ステップS205における処理の詳細を示すフローチャートである。

【0165】

40
先ず、レコード数をカウントするための変数xに0を代入して初期化する（ステップS250）。次に、変数xを参照し、変数xの値が、最大のレコード数に達したか否かを判断することで、全レコードについて動き補償処理が終了したか否かを判定し（ステップS251）、終了していれば処理を図7のステップS206に進め、終了していなければ処理をステップS252に進める。

【0166】

全レコードについて動き補償処理が終了していない場合、レコードxに有効なデータが格納されているか否かを示す有効フラグF(x)を参照し、0か否かを判断する（ステップS252）。

【0167】

$F(x) = 0$ の場合、すなわちレコード x に有効なデータが格納されていない場合（例えば、 $F(x) = S(x) = E(x) = D(x) = 0$ の場合）、処理をステップ $S253$ に進め、変数 $R(x)$ に最大数 MAX を代入する（ステップ $S253$ ）。そして変数 x の値を 1 つインクリメントし（ステップ $S256$ ）、処理をステップ $S251$ に戻す。

【0168】

一方、 $F(x) = 0$ の場合、すなわち、レコード x に有効なデータが格納されている場合、処理をステップ $S252$ からステップ $S254$ に進め、変数 $R(x)$ に 0 を代入し、初期化する（ステップ $S254$ ）。そして次に、ステップ $S257 \sim$ ステップ $S259$ の処理を、ステップ $S203$ で画像エリアに格納したフレームの画像を構成する全てのマクロブロックについて行ったか否かを判断する（ステップ $S255$ ）。

10

【0169】

ステップ $S203$ で画像エリアに格納したフレームの画像を構成する全てのマクロブロックについてステップ $S257 \sim$ ステップ $S259$ の処理を行った場合には処理をステップ $S256$ に進める。

【0170】

一方、ステップ $S203$ で画像エリアに格納したフレームの画像を構成する全てのマクロブロックについてステップ $S257 \sim$ ステップ $S259$ の処理を行っていない場合には処理をステップ $S255$ からステップ $S257$ に進め、図 9 に示したテーブルを参照し、代表フレーム番号 $D(x)$ のフレームデータを読み出し、ステップ $S203$ で画像メモリに格納したフレームの画像と比較しながら周知の動きベクトルの探索処理を行なう（ステップ $S257$ ）。

20

【0171】

そして、ステップ $S203$ で画像エリアに格納した画像と、この動きベクトルに基づいて生成されるこの画像の予測画像との予測誤差を求める（ステップ $S258$ ）。予測誤差を求めるための処理については上記実施形態と同様、周知の技術である。そして求めた予測誤差を変数 $R(x)$ に加算する（ステップ $S259$ ）。そして処理をステップ $S255$ に戻し、全てのマクロブロックについてステップ $S257 \sim$ ステップ $S259$ の処理を繰り返す。これにより、代表フレーム番号 $D(x)$ のフレームの画像と、ステップ $S203$ で画像エリアに格納された画像とに基づいた予測誤差を求めることができる。

【0172】

30

また、以上の処理を、 $F(x) = 0$ である全てのレコードについて行なうことにより、各代表画像毎に、ステップ $S203$ で画像メモリに格納された画像の予測画像との予測誤差を求めることができる。

【0173】

図 7 に戻って、各代表画像毎に、ステップ $S203$ で画像エリアに格納された画像の予測画像との予測誤差を求めると、次に、全ての x （本実施形態では 0 から 7）についての変数 $R(x)$ の値を参照し、変数 $R(x)$ の値を最小にする変数 x の値（ x_m ）を特定し、図 9 のテーブルから $S(x_m)$ 、 $E(x_m)$ のデータを読み出すことで、 $S(x_m)$ を始点値、 $E(x_m)$ を終点値とする区間を、動き補償を行なう区間として選択する（ステップ $S206$ ）。

40

【0174】

そして、ステップ $S203$ で画像エリアに格納された画像を、ステップ $S206$ で選択された区間内の各フレームの画像との間で上記周知の動き補償による符号化処理であるフレーム間符号化を行なう（ステップ $S207$ ）。符号化結果のデータはメモリ 301 上の符号エリアに格納される。また、局所復号された画像データは最も古いフレーム番号の画像エリアに格納される。また、このときの動きベクトル探索を行なった際に算出される予測誤差をマクロブロックごとに求め、その総和値をとして求めておく。

【0175】

次に、通信ソフトウェアを実行することにより、メモリ 301 上の符号エリアの符号化データを通信インターフェース 308 を介して通信回路 309 に出力し、その後、メモ

50

リ 3 0 1 上の符号エリアの該当する部分をクリアする（ステップ S 2 0 8）。

【 0 1 7 6 】

次に、ステップ S 2 0 7 で求めた上記 の値と、メモリ 3 0 1 の所定のエリアに記憶されている閾値 T_h とを比較し、 T_h であるか否かを判断する（ステップ S 2 0 9）。

【 0 1 7 7 】

T_h の場合、ステップ S 2 1 2 にて、有効フラグが 0 のレコードを検出し、そのレコードの有効フラグを 1 にし、その始点値と終点値に符号化したフレームの番号を格納する。もし有効フラグが全て 1 であれば、各レコードの始点値を比較し、最も小さい（古い）区間にレコードに上書きする。ステップ S 2 1 3 にて、代表フレーム番号に符号化したフレームの番号を格納する。

10

【 0 1 7 8 】

ステップ S 2 1 4 にて、図 9 のテーブルで、復号画像を上書きしたフレームの属するレコードの終点値を 1 つ繰り上げる。また、最新の区間（符号化したフレームの直前の区間）の終点値を符号化したフレームの番号に変更する。

次のフレームの処理を行なうためにステップ S 2 0 2 に戻る。

【 0 1 7 9 】

このような一連の選択動作により、参照するフレーム数を減らし、高速に動き補償処理を実現しながら予測誤差の増大を抑えることができる。

【 0 1 8 0 】

また、符号化時に計算される動き補償時の誤差によって区間の更新を行なうかどうか判断することで、新たな計算を生じることがないため、処理規模の増大を最小限に抑える効果もある。

20

【 0 1 8 1 】

なお、本実施形態では参照するフレームを 8 フレームとしたが、これに限定されず、画像エリア等を必要な量だけ確保することでその他の枚数の参照にも対応できることは明らかである。

【 0 1 8 2 】

また本実施形態に係る上記処理を第 1、2 の実施形態のように、専用のハードウェアをもって実現するようにしても良い。

【 0 1 8 3 】

また本実施形態では、符号化結果は通信インターフェース 3 0 8 を介して外部に出力したが、これに限定するものではなく、例えば、記憶装置 3 0 5 に記録するようにしても良い。

30

【 0 1 8 4 】

また本実施形態では、カメラ 3 0 6 で撮像された画像データは記憶装置 3 0 5 に一旦記録する例を取ったが、これに限定されず、直接メモリ 3 0 1 の画像エリアに格納してももちろん構わない。

【 0 1 8 5 】

[第 4 の実施形態]

第 3 の実施形態とは異なる符号化処理について説明する。

40

【 0 1 8 6 】

図 1 0 は、本実施形態に係る動画像符号化処理のフローチャートである。なお、本実施形態に係る動画像符号化装置は第 3 の実施形態と同様の構成を備えるものであるとする。従って、図 1 0 に示すフローチャートに従った処理は、CPU 3 0 0 が実行する。

【 0 1 8 7 】

また、図 1 0 において、図 7 と同じ処理ステップ部分には同じステップ番号をつけており、その説明を省略する。

【 0 1 8 8 】

フレーム間符号化を行なう場合、まず、区間の参照回数を記録するための変数 $T(x)$ に 0 を代入し、初期化する（ステップ S 3 0 0）。次に、全てのスライスの符号化が終了

50

したか否かを判断し（ステップS301）、終了していなければ処理をステップS302に進め、図8に示したフローチャートに従った処理をスライス単位で行ない、変数R（x）の値を求める（ステップS302）。

【0189】

そして、ステップS206と同様にして（本実施形態の場合にはスライス単位に）区間を設定する（ステップS303）。そして設定した区間に対する参照回数を示す変数T（x）の値を1つインクリメントする（ステップS304）。このときの変数xの値は、ステップS303で設定した区間（始点値、終点値）を格納するレコードxの値である。

【0190】

そして符号化対象のスライスに対して上記実施形態と同様な処理により符号化処理を行なう（ステップS305）。 10

【0191】

一方、全てのスライスに対する符号化処理が完了している場合、処理をステップ208に進め、第3の実施形態と同様に、符号化データを出力する（ステップS208）。次に、変数T（x）の値、及び変数xの値を参照し、変数T（x）の値を最大にする変数xの値は最大値であるか否かを判断する（ステップS306）。すなわち、最もよく参照された区間が最新区間か否かの判断を行なう。

【0192】

最もよく参照された区間が最新区間ではない場合、処理をステップS212に進め、第3の実施形態と同様にしてステップS212からステップS213の処理を行なう。 20

【0193】

一方、最もよく参照された区間が最新区間である場合、処理をステップS214に進め、第3の実施形態と同様にして区間変更処理を行ない（ステップS214）、更新された区間の代表フレームを更新する（ステップS307）。

【0194】

ステップS307における処理をより詳細に説明すると、先ず更新された区間のレコードから区間の始点値S（x）、終点値E（x）を読み出す。新たな区間の代表フレーム番号D（x）は次式にしたがって算出して更新する。

【0195】

$$D(x) = (S(x) + E(x)) / 2 \quad (3)$$

30

このような一連の選択動作により、参照するフレーム数を減らし、高速に動く補償処理を実現しながら、予測誤差の増大を抑えることができる。

【0196】

また、代表フレームを区間の更新と共に行なうので、常に、最適な代表フレームを選択できる。代表フレームを区間の中央とすることによって、時間的に区間をよりよく代表することが可能にもなる。

【0197】

また、代表フレームを区間の中央としたが、これに限定されず、区間の始点、終点の両フレームとしても構わないし、区間内に複数枚にしてももちろん構わない。

【0198】

40

なお、本実施形態においては参照するフレームを8フレームとしたが、これに限定されず、画像エリアなどを必要な量を確保することでその他の枚数の参照にも対応できることは明らかである。

【0199】

〔第5の実施形態〕

図12は、本実施形態に係る動画像符号化装置の構成を示すブロック図である。本実施形態では、H.264符号化方式をベースにとって説明するが、これに限定されない。また、説明を容易にするため、過去のフレームを参照する前方向予測を例に取るが、これに限定されず、両方向予測に適用してももちろん構わない。また、説明を容易にするため、参照フレームを3フレームとして説明するが、これに限定されない。 50

【0200】

同図において519～523は動画像を構成する各フレームの画像（入力画像）を格納するフレームメモリである。

【0201】

525は入力先を選択して出力するセクタである。51はフレーム内符号化／フレーム間符号化のモードで出力を選択するセクタ、52はH.264符号化方式によるイントラ予測を行なうイントラ予測器、53は動き予測誤差を求める差分器である。

【0202】

54は整数型直交変換を行ない、係数を量子化する変換／量子化器、57はその逆の作用を施す逆量子化／逆変換器である。58は動き予測誤差を加算する加算器である。55は量子化結果を符号化するエントロピー符号化器、56は発生した複数フレーム分の符号を格納するバッファである。

10

【0203】

59、510、511、512はフレームメモリで、局所復号された画像データとフレームの番号をフレーム単位で格納する。513は入力画像と復号された画像データから最適な動きベクトルを該当するフレームから抽出する動き推定器、514は動き推定器513で算出された動きベクトルと該当するフレームの情報から予測画像を生成する動き補償器である。

【0204】

515は算出された動きベクトルと該当するフレームの情報から動きの情報を符号化する動き符号化器である。517はフレームメモリ59～512の入出力を制御するフレームメモリ制御器、516はバッファ56に格納された符号化データと参照情報を合成して出力する出力器である。518は入力画像を比較して代表フレームを判定する代表フレーム判定器である。524は代表フレーム判定器518が代表フレームであると判定した際に、この代表フレームのフレーム番号を出力器516に出力する代表フレームフラグ発生器である。

20

【0205】

上記構成を備える本実施形態に係る動画像符号化装置が行なう処理について以下、説明する。

【0206】

符号化に先立ち、本符号化データが代表フレームを参照する方式であることを示す符号を代表フレームフラグ発生器524が生成して出力器516に出力する。この符号はH.264符号化方式と互換性を保つために、ユーザデータとして符号化されることとするがこれに限定されず、電子的な透かしとして埋め込みを行なってももちろん構わない。

30

【0207】

さらに、参照に用いることができるフレーム数などを含むヘッダ情報を出力器516が生成して出力する。このとき、参照に用いることのできるフレーム数は3としておく。また、各フレームメモリは0にリセットされる。

【0208】

本装置に入力された各フレームの画像データはフレームメモリ519～523に順に格納される。本実施形態に係る動画像符号化装置に備わっている入力画像格納用フレームメモリの数は5つであるので、一度に格納しておく入力画像の数は5つであるが、入力画像格納用フレームメモリの数はこれに限定するものではないので、当然、一度に格納しておく入力画像の数もこれに限定するものではない。

40

【0209】

フレームメモリ519～523のそれぞれに入力画像が格納されると、代表フレーム判定器518は各フレーム間相違度Rを求める。ここでは説明を容易にするため、第1の実施形態と同様に、それぞれの画像において位置的に対応する画素間の画素値の差分の絶対値の総和を相違度Rとする。

【0210】

50

すなわち、フレームメモリ 5 1 9 に格納されている第 N フレームの画素の値を $X_N(i, j)$ (i, j は画素の位置)、フレームメモリ 5 2 0 に格納されている第 (N + 1) フレームの画素の値を $X_{N+1}(i, j)$ 、フレームメモリ 5 2 1 に格納されている第 (N + 2) フレームの画素の値を $X_{N+2}(i, j)$ 、フレームメモリ 5 2 2 に格納されている第 (N + 3) フレームの画素の値を $X_{N+3}(i, j)$ 、フレームメモリ 5 2 3 に格納されている第 (N + 4) フレームの画素の値を $X_{N+4}(i, j)$ とする。

【0211】

このとき、第 N フレームとその他のフレーム (第 (N + 1) フレーム目 ~ 第 (N + 4) フレーム目の各フレーム) との相違度 R_N は、以下の式でもって求められる。

【0212】

$$R_N = \begin{aligned} & (|X_N(i, j) - X_{N+1}(i, j)|) \\ & + (|X_N(i, j) - X_{N+2}(i, j)|) \\ & + (|X_N(i, j) - X_{N+3}(i, j)|) \\ & + (|X_N(i, j) - X_{N+4}(i, j)|) \end{aligned} \quad (4)$$

ここで「 \sum 」は全ての i, j について加算することを示している。

【0213】

同様に、第 (N + 1) フレームとその他のフレームとの相違度 R_{N+1} は、以下の式でもって求められる。

【0214】

$$R_{N+1} = \begin{aligned} & (|X_{N+1}(i, j) - X_N(i, j)|) \\ & + (|X_{N+1}(i, j) - X_{N+2}(i, j)|) \\ & + (|X_{N+1}(i, j) - X_{N+3}(i, j)|) \\ & + (|X_{N+1}(i, j) - X_{N+4}(i, j)|) \end{aligned} \quad (5)$$

以下、同様に相違度 R_{N+2} 、 R_{N+3} 、 R_{N+4} を求める。

【0215】

代表フレーム判定器 5 1 8 はこれら相違度 $R_N \sim R_{N+4}$ を参照し、最も小さい相違度となったフレームを代表フレームとして選択する。例えば、相違度 $R_N \sim R_{N+4}$ のうち、 R_{N+2} が最も小さい場合、第 (N + 2) フレーム目を代表フレームとして選択する。

【0216】

そして代表フレーム判定器 5 1 8 は、最も小さい相違度となったフレームのフレーム番号を保持しておく。

【0217】

続いて、セクタ 5 2 5 はフレーム 5 1 9 ~ 5 2 3 のフレームデータを順に読み出し、セクタ 5 1 に入力する。セクタ 5 1 は一定間隔、例えば 1 5 フレーム毎にフレームをフレーム内符号化するために、出力にイントラ予測器 5 2 を選択し、そうでない場合は、フレーム間符号化するために出力に差分器 5 3 を選択する。イントラ予測器 5 2 が選択された場合、入力されたフレームデータはブロック毎に予測が行なわれる。

【0218】

一方、差分器 5 3 が選択された場合は、入力された画像データはマクロブロック単位で動き推定器 5 1 3 に入力される。

【0219】

そして、動き推定器 5 1 3、動き補償器 5 1 4、動き符号化器 5 1 5 は図 1 の動き推定器 1 3、動き補償器 1 4、動き符号化器 1 5 と同様の処理を行ない、これら 5 フレームの各フレームに対する動き補償による符号化処理を行なう。なお、これら 5 フレームの各フレームに対する動き補償を行なう際に参照する画像は以下のようにして生成される画像である。

【0220】

すなわち、フレームメモリ 5 9 ~ 5 1 2 に保持しておいた各代表フレームのデータの符号化対象フレームに対する予測画像を生成し、そしてそれぞれの代表フレームの予測画像

10

20

30

40

50

と符号化対象フレームの画像との誤差（予測誤差）を求め、そして、符号化対象フレームとの予測誤差が最も小さい代表フレームを参照画像とする。

【0221】

ここで本実施形態に係る特徴は、本実施形態のように一度に5フレーム分しか符号化できなくても、後で5フレーム分の符号化を行なう際にはフレームメモリ59～512には代表フレームの画像が記憶されているので、一度に5フレーム以上の符号化を行なうことに等しくなる点にある。

【0222】

また、代表フレーム判定器518は、フレームメモリ519～523から読み出す各フレームデータのうち、保持された代表フレームの番号によって代表フレームを検知すると、そのフレーム番号を代表フレームフラグ発生器524に出力する。代表フレームフラグ発生器524はこのフレーム番号を出力器516に出力する。なお、代表フレームフラグ発生器524は、通知された最新のフレーム番号を保持しており、新たに通知されるまで、現在保持しているフレーム番号を出力する。なお、代表フレームが特定できるのであれば、フレーム番号に限定するものではない。

【0223】

最後に、バッファ56に格納された5フレーム分の各フレームの符号化データは出力器516により、代表フレームフラグ発生器524からの「代表フレームのフレーム番号」を付加して外部に出力される。なお、代表フレームが特定できるのであれば、フレーム番号に限定するものではない。

【0224】

そして、イントラ予測器52、差分器53の何れにおいても注目フレームデータに対する予測誤差が出力されるので、この予測誤差が変換／量子化器54、逆量子化／逆変換器57を介してそのままの状態に加算器58で、後述する動き補償器514から入力される「注目フレームデータに対する予測画像」と加算される。例えば予測誤差が差分器53から出力された場合、この予測誤差とは、セクタ525から入力された注目フレームデータと、この注目フレームデータに対する予測画像との差分であるので、加算器58がこの差分と動き補償器514から入力される「注目フレームデータに対する予測画像」と加算することにより、注目フレームデータが復元される。フレームメモリ制御器517は代表フレーム判定器518で打表フレームと判定された場合に、復元されたフレームデータがフレームメモリ59～512に格納するように制御する。

【0225】

図13は、以上説明した本実施形態に係る符号化処理のフローチャートである。

【0226】

まず、出力器516は、以下行なう符号化方式が代表フレームを用いる符号化方式であることを示す情報や、参照するフレームの最大数情報を含むヘッダ情報を生成して出力する（ステップS51）。

【0227】

次に、符号化処理すべき対象のフレーム全てについて以下の符号化処理を行っていない場合には（ステップS52）、処理をステップS53に進め、以降の処理を行なう。代表フレーム判定器518は、フレームメモリ519～523の各フレームメモリにフレームデータが格納される（すなわち5フレーム分のフレームデータがフレームメモリ519～523のそれぞれに用意される）まで処理を待つ（ステップS53）。従ってその間、各フレームメモリ519～523にはそれぞれ1フレームずつフレームデータが格納される（ステップS54）。

【0228】

フレームメモリ519～523の各フレームメモリにフレームデータが格納される（すなわち5フレーム分のフレームデータがフレームメモリ519～523のそれぞれに用意される）と、処理をステップS55に進め、代表フレーム判定器518は、上記処理により、この5フレーム分の各フレームについて、相違度 R_N を求める（ステップS55）。

そして求めた5つの相違度のうち、最小の相違度となったフレームを代表フレームとして決定する(ステップS56)。その場合、代表フレーム判定器518は、代表フレームのフレーム番号を代表フレームフラグ発生器524に出力する。

【0229】

そして以下、この5フレーム分の各フレームを符号化する。

【0230】

まず、この5フレーム分全てについて符号化処理を行ったかを判断し(ステップS57)、行った場合には処理をステップS52に戻し、以降の処理を繰り返す。一方、この5フレーム分全てについて符号化処理を行っていない場合には処理をステップS58に進め、符号化しようとするフレームが代表フレームであるかを判断する。

10

【0231】

符号化しようとするフレームが代表フレームであるか否かは、符号化しようとするフレームのフレーム番号が代表フレーム判定器518により得られるフレーム番号と一致するか否かでもって判定可能であるので(一致する場合には代表フレームであると判定)、この判定を行ない(ステップS58)、符号化対象フレームが代表フレームである場合には処理をステップS59に進め、代表フレームフラグ発生器524は、代表フレームのフレーム番号を出力器516に出力する(ステップS59)。そして符号化対象フレームに対して動き補償による符号化処理を行なうのであるが、この符号化処理は上述の通り、第1の実施形態と同じである。

【0232】

20

そして符号化した代表フレームの画像を第1の実施形態と同様にして復号し、復号画像をフレームメモリ59~512の内、最も古いフレーム番号のフレームを格納しているフレームメモリに格納する(ステップS511)。

【0233】

一方、符号化しようとするフレームが代表フレームではない場合、処理をステップS58からステップS512に進め、ステップS510と同様の処理でもって符号化を行なう(ステップS512)。

【0234】

以上の処理により、バッファ56には、5フレーム分の各フレームの符号化データが格納されるので、出力器516はバッファ56からこれら各フレームの符号化データに、代表フレームフラグ発生器524から得られる「代表フレームのフレーム番号」を付加して外部に出力する(ステップS513)。

30

【0235】

このような一連の選択動作により、少ないフレーム数で長い時間のフレームを参照することが可能になり、符号化時にメモリ容量を抑制するといった効果がある。また、代表フレームをその他のフレームとの類似度の高いフレームを選択することで、符号化効率を向上させることができる。

【0236】

また、代表フレームの符号化データを、ユーザデータなどの復号時に無視することができる符号化データとすることで既存のH.264と互換性を保つことも可能となる。

40

【0237】

また、本実施形態では参照の対象をフレームとしたが、これに限定されず、スライス等のフレームを分割した単位で行なってももちろん構わない。

【0238】

また、代表フレームの符号化方式をフレーム内符号化としてもよく、その場合、高い画質のフレームを参照画像データとすることで画質が向上するという効果がある。

【0239】

[第6の実施形態]

本実施形態は、第5の実施形態に係る動画像符号化装置が出力した符号化結果を復号する復号装置について説明する。

50

【 0 2 4 0 】

図 1 4 は、本実施形態に係る復号装置の構成を示す図である。

【 0 2 4 1 】

5 1 0 0 は符号化データを入力し、後段に分離して出力する入力器である。この符号化データには第 5 の実施形態に係る説明の通り、代表フレームのフレーム番号が付加されている。

【 0 2 4 2 】

5 1 0 1 は符号化データを復号して量子化結果を得るエントロピー復号器である。5 1 0 2 は量子化結果を逆量子化し、整数型直交変換の逆変換を行なう逆量子化 / 逆変換器である。5 1 0 3 は H . 2 6 4 符号化方式によるイントラ予測を行なうイントラ予測器であり、5 1 0 4 は逆量子化 / 逆変換器 5 1 0 2 からの動き予測誤差を、動き補償器 5 1 1 0 からの予測画像に加算する加算器である。5 1 1 3 は復号した画像を一時格納するフレームメモリである。

10

【 0 2 4 3 】

5 1 0 5、5 1 0 6、5 1 0 7、5 1 0 8 はフレームメモリであり、復号された画像データとそのフレーム番号をフレーム単位で格納する。本実施形態では第 5 の実施形態に係る動画像符号化装置が出力した符号化結果を復号するためにフレームメモリの数を 4 としたが、これに限定するものではない。

【 0 2 4 4 】

5 1 0 9 は動きベクトルと該当するフレームの情報を復号する動き復号器であり、5 1 1 0 は動き復号器 5 1 0 9 で復号された動きベクトルと該当するフレームの情報から予測画像を生成する動き補償器である。5 1 1 1 は入力器 5 1 0 0 が入力した符号化データに付加されていた「代表フレームのフレーム番号」を参照し、後段のフレームメモリ制御器 5 1 1 2 に通知する代表フレーム解析器、5 1 1 2 はフレームメモリ 5 1 0 5 ~ 5 1 0 8 の入出力を制御するフレームメモリ制御器である。

20

【 0 2 4 5 】

以上の構成を備える、本実施形態に係る復号装置が行なう復号処理について、以下説明する。

【 0 2 4 6 】

先ず、入力器 5 1 0 0 は代表フレームを参照する方式である符号参照に用いることができるフレーム数などを含むヘッダ情報を入力し、これを解釈し、後段の各部を初期化する。

30

【 0 2 4 7 】

そして入力器 5 1 0 0 は、各フレームの符号化データ（各フレームの代表フレームのフレーム番号が付加されている）を読み込み、この符号化データに付加されている「代表フレームのフレーム番号」のデータを代表フレーム解析器 5 1 1 1 に出力する。

【 0 2 4 8 】

また入力器 5 1 0 0 は、この符号化データのうち、動きベクトルに関する符号化データを動き復号器 5 1 0 9 に、画像に関する符号化データをエントロピー復号器 5 1 0 1 に出力する。

40

【 0 2 4 9 】

フレームの符号化モードがフレーム内符号化であれば、エントロピー復号器 5 1 0 1 は入力された符号化データを復号し、量子化結果を再生する。逆量子化 / 逆変換器 5 1 0 2 はこれを逆量子化し、整数型直交変換の逆変換を施して画像データを再生する。

【 0 2 5 0 】

再生された画像はイントラ予測器 5 1 0 3 でブロックごとの予測を行なって再生画像を得る。復号された画像データはフレームメモリ 5 1 1 3 に出力される。

【 0 2 5 1 】

フレームメモリ制御器 5 1 1 2 は、代表フレーム解析器 5 1 1 1 から通知されたフレーム番号と同じフレーム番号の復号データ（すなわち、代表フレームの復号画像）がイント

50

ラ予測器 5 1 0 3、もしくは加算器 5 1 0 4 から出力されると、この復号画像データを、フレームメモリ 5 1 0 5 ~ 5 1 0 8 のうち最も古いフレーム番号のフレームを格納しているフレームメモリに格納する。

【 0 2 5 2 】

なお、イントラ予測器 5 1 0 3、もしくは加算器 5 1 0 4 から出力された復号画像が代表フレームの復号画像ではない場合には、フレームメモリ 5 1 0 5 ~ 5 1 0 8 への復号画像の書き込みは行なわない。

【 0 2 5 3 】

そしてフレームメモリ 5 1 1 3 に格納された復号画像のデータは所定のタイミング（例えば所定フレーム数毎、もしくは所定時間毎）でもって外部に出力する。

10

【 0 2 5 4 】

一方、フレームの符号化モードがフレーム間符号化であれば、動き復号器 5 1 0 9 は動きベクトルと参照するフレームの情報を復号し、これに従って動き補償器 5 1 1 0 がフレームメモリ 5 1 0 5 ~ 5 1 0 8 を参照して予測誤差を算出し、加算器 5 1 0 4 に出力する。エントロピー復号器 5 1 0 1 は入力された符号化データを復号し、量子化結果を再生する。逆量子化 / 逆変換器 5 1 0 2 ではこれを逆量子化し整数型直交変換の逆変換を施して予測誤差を再生する。再生された予測誤差は加算器 5 1 0 4 でブロックごとに加算を行なって再生画像を得て、フレームメモリ 5 1 1 3 に格納する。この場合、フレームメモリ 5 1 0 5 ~ 5 1 0 8 への書き込みは行なわれない。最後に、フレームメモリ 5 1 1 3 に格納された画像データは外部に出力される。

20

【 0 2 5 5 】

すなわち、上記復号処理は、代表フレームの復号画像をフレームメモリ 5 1 0 5 ~ 5 1 0 8 に格納する処理、及び、参照する画像が代表フレームの画像であること以外は、動き補償による符号化処理の符号化結果を復号する周知の処理である。

【 0 2 5 6 】

図 1 5 は、以上説明した本実施形態に係る復号処理のフローチャートである。

【 0 2 5 7 】

先ず入力器 5 1 0 0 は、符号化データに先立って入力される上記ヘッダ情報を参照し、解析する（ステップ S 5 1 0 1）。特に参照可能なフレーム数を解析して保持しておく。

【 0 2 5 8 】

30

次に、全てのフレームの復号が終了したか否かを判断し（ステップ S 5 1 0 2）、終了していなければ処理をステップ S 5 1 0 3 に進め、入力器 5 1 0 0 は各フレームの符号化データを入力する（ステップ S 5 1 0 3）。次に、この符号化データに付加されている、このフレームの代表フレームのフレーム番号を、この符号化データから分離して代表フレーム解析器 5 1 1 1 に入力し、代表フレーム解析器 5 1 1 1 は、このフレーム番号のデータをフレームメモリ制御器 5 1 1 2 に出力する（ステップ S 5 1 0 4）。

【 0 2 5 9 】

そして、動き復号器 5 1 0 9、動き補償器 5 1 1 0、エントロピー復号器 5 1 0 1、逆量子化 / 逆変換器 5 1 0 2、イントラ予測器 5 1 0 3 もしくは加算器 5 1 0 4 により、入力された符号化データを H . 2 6 4 符号化方式の復号方法に従って復号する（ステップ S 5 1 0 5）。復号結果のデータ、すなわち、復号画像のデータはフレームメモリ 5 1 1 3 に出力される。

40

【 0 2 6 0 】

フレームメモリ制御器 5 1 1 2 には、現在復号されているフレームの代表フレームのフレーム番号がステップ S 5 1 0 4 で通知されているので、現在復号されているフレームが代表フレームであるか否かを、現在復号されているフレームのフレーム番号を参照することで判断することができる。

【 0 2 6 1 】

よってこの判断により、現在復号されているフレームが代表フレームである場合には処理をステップ S 5 1 0 7 に進め、フレームメモリ 5 1 0 5 ~ 5 1 0 8 のうち、最も古いフ

50

フレーム番号のフレームを格納しているフレームメモリを特定し（ステップS5107）、特定したフレームメモリに、復号した代表フレームの復号画像データを格納する（ステップS5108）。

【0262】

そして、フレームメモリ5113に出力された復号画像のデータを外部に出力する（ステップS5109）。

【0263】

このような一連の選択動作により、代表フレームの符号化データのみを動き補償時に参照することが可能になり、参照画像を格納するメモリを少なくできるといった効果がある。

10

【0264】

また本実施形態に係る復号装置は、図14に示した構成を有するハードウェアとしたが、これに限定するものではなく、他にも例えば、図14に示した各部の機能をソフトウェアでもって実装し、このプログラムをPC（パーソナルコンピュータ）やWS（ワークステーション）などのコンピュータにインストールし、このコンピュータのCPUがこれを実行することにより、このコンピュータは上述した復号処理を実行することになる。よってこの場合、このコンピュータをもってして本実施形態に係る復号装置とすることができ

【0265】

〔第7の実施形態〕

20

図16は、本実施形態に係る動画像符号化装置の構成を示す図である。なお同図において図12と同じ部分については同じ番号をつけており、その説明を省略する。

【0266】

5200は符号化しようとする動画像を構成する各フレームの画像（入力画像）を保持するフレームメモリである。5213は動き推定器であり、入力画像データとフレームメモリ59～512の画像データとを比較して動きベクトルとその参照したフレーム番号を出力する。また、そのときの誤差も出力する。

【0267】

5202はフレームカウンタであり、フレームメモリ5200から1フレーム出力されるたびに1つカウントアップする。すなわち、フレームメモリ5200から出力されたフレームの数をカウントする。

30

【0268】

5201はセクタで、フレームカウンタ5202によるカウント値を整数Mで割った余りがN（ $N < M$ ）の時に、出力先をイントラ予測器52に切り替え、それ以外では出力先を差分器53に切り替える。なお、このM、Nの値については特に限定するものではなく、要は、セクタ5201は通常は出力先を差分器53とし、周期的に1回だけ出力先をイントラ予測器52に切り替える。よってこの処理が達成されるのであれば、その構成は特に限定するものではない。以下の説明では例として $M = 5$ 、 $N = 1$ とする。

【0269】

5218は代表フレームを判定する代表フレーム判定器である。5224は代表フレーム判定器5218が代表フレームであると判定した際に、この代表フレームのフレーム番号を出力器516に出力する代表フレームフラグ発生器である。なお、代表フレームフラグ発生器5224は、通知された最新のフレーム番号を保持しており、新たに通知されるまで、現在保持しているフレーム番号を出力する。

40

【0270】

以上の構成を備える、本実施形態に係る動画像符号化装置が行なう処理について以下、説明する。

【0271】

第5の実施形態と同様に、符号化に先立ち、本符号化データが代表フレームを参照する方式であることを示す符号を代表フレーム符号化器5224が生成して出力器516に出

50

力する。さらに、参照に用いることができるフレーム数などを含むヘッダ情報を出力器 5 1 6 が生成して出力する。このとき、参照に用いることのできるフレーム数は 3 としておく。また、各フレームメモリは 0 にリセットされる。また、フレームカウンタ 5 2 0 2 が保持するカウント値を 0 にリセットする。

【 0 2 7 2 】

そしてフレームメモリ 5 2 0 0 には、各フレームの画像データが入力され、後段のセクタ 5 2 0 1 に出力するのであるが、フレームカウンタ 5 2 0 2 はこの出力毎にカウント値を 1 つカウントアップする。代表フレーム判定器 5 2 1 8 はこのカウント値を常に参照し、カウント値を 5 で割った余りが 1 である時にフレームメモリ 5 2 0 0 から出力されたフレームを代表フレーム、カウンタ値がそれ以外の時にフレームメモリ 5 2 0 0 から出力されたフレームを非代表フレームと判定する。

10

【 0 2 7 3 】

そして上述の通り、フレームメモリ 5 2 0 0 から出力されたフレームデータが代表フレーム、非代表フレームの何れであるかに応じてイントラ予測器 5 2、もしくは差分器 5 3 を介して変換 / 量子化器 5 4 に予測誤差として出力されるのであるが、この予測差分が代表フレームの予測差分である（代表フレーム判定器 5 2 1 8 が代表フレームであると判定した）場合、第 5 の実施形態と同様に、逆量子化 / 逆変換器 5 7、加算器 5 8 を動作させて代表フレームの画像を復号し、復号した画像をバッファ 5 6 に出力するとともに、復号した画像をフレームメモリ 5 9 ~ 5 1 2 のうち、最も古いデータに画像データとフレーム番号上書きするようにフレームメモリ制御器 5 1 7 に指示を出す。

20

【 0 2 7 4 】

更に代表フレームフラグ発生器 5 2 2 4 は、代表フレーム判定器 5 2 1 8 が代表フレームであると判定した際に代表フレーム判定器 5 2 1 8 から送出されたこの代表フレームのフレーム番号を出力器 5 1 6 に出力する。

【 0 2 7 5 】

一方、予測差分が代表フレームの予測差分ではない（代表フレーム判定器 5 2 1 8 が代表フレームではないと判定した）場合、第 5 の実施形態と同様に、逆量子化 / 逆変換器 5 7、加算器 5 8 を動作させて代表フレームの画像を復号し、復号した画像をバッファ 5 6 に出力するのであるが、復号した画像をフレームメモリ 5 9 ~ 5 1 2 には格納しないよう、フレームメモリ制御器 5 1 7 に指示を出す。

30

【 0 2 7 6 】

また、第 5 の実施形態と同様に、セクタ 5 2 0 1 によって一定の間隔でフレーム内符号化を行なう場合、入力されたフレームデータはマクロブロック単位でイントラ予測器 5 2 に入力され、ブロック毎に予測が行なわれる。その結果は変換 / 量子化器 5 4 で整数型直交変換と量子化が行なわれ、その結果はエントロピー符号化器 5 5 に入力される。

【 0 2 7 7 】

エントロピー符号化器 5 5 では、入力された量子化結果をエントロピー符号化し、バッファ 5 6 にフレーム単位で格納する。代表フレームであれば、逆量子化 / 逆変換器 5 7 では、入力された量子化結果から復号画像を得る。

【 0 2 7 8 】

40

そしてフレームメモリ制御器 5 1 7 の指示に従って、フレームメモリ 5 9 ~ 5 1 2 のうち、空いているフレームメモリ、または最も古い代表フレームの画像データのフレームメモリに復号画像データとフレーム番号を格納する。非代表フレームであれば、フレームメモリ 5 9 ~ 5 1 2 への格納は行なわない。

【 0 2 7 9 】

そしてバッファ 5 6 に格納された符号化データは出力器 5 1 6 で代表フレームフラグ発生器 5 2 2 4 が出力する代表フレームを付加して出力される。

【 0 2 8 0 】

一方、フレーム間符号化を行なう場合についても第 5 の実施形態と同じである。すなわち、セクタ 2 0 1 はフレームデータの出力先を差分器 3 に切り替える。フレームメモリ

50

5 2 0 0からはマクロブロック単位でセクタ5 2 0 1、動き推定器5 1 3に入力され、動き推定器5 1 3はフレームメモリ5 9 ~ 5 1 2の内容を読み出し、動きベクトルとフレーム番号を算出し、動き補償器5 1 4で予測画像を生成し、差分器5 3に入力する。差分器5 3は予測誤差を求め、その結果は変換/量子化器5 4に入力され、整数型直交変換と量子化を行なう。その結果はエントロピー符号化器5 5に入力される。エントロピー符号化器5 5では、入力された量子化結果をエントロピー符号化し、バッファ5 6にフレーム単位で格納する。

【0 2 8 1】

バッファ5 6に格納された符号化データは出力器5 1 6で代表フレームフラグ発生器5 2 2 4が出力する代表フレームを付加して出力される。

10

【0 2 8 2】

図1 7は、以上説明した符号化処理のフローチャートである。なお、同図において図1 4と同じ処理ステップには同じステップ番号をつけており、その説明を省略する。

【0 2 8 3】

先ず、上記ステップS 5 1の処理を行ない、次に、フレームカウンタ5 2 0 2が保持するカウンタ値を0にリセットする(ステップS 5 2 0 0)。次に、全フレームについて符号化を行っていない場合には処理をステップS 5 4に進め、上記ステップS 5 4と同様の処理でもってフレームメモリ5 2 0 0からフレームデータを後段の各部に出力する(ステップS 5 4)。

【0 2 8 4】

20

そしてフレームメモリ5 2 0 0から1フレームのフレームデータが出力される毎にフレームカウンタ5 2 0 2が保持するカウンタ値を1つカウントアップする(ステップS 5 2 0 1)。現在のカウンタ値をMで割った余りが1の場合(すなわちステップS 5 4でフレームメモリ5 2 0 0から出力されたフレームのフレーム番号が $(nM + 1)$ の場合)、ステップS 5 4でフレームメモリ5 2 0 0から出力されたフレームが代表フレームであると判定し、処理をステップS 5 2 0 4に進め、先ず、代表フレームフラグ発生器5 2 2 4により、代表フレーム判定器5 2 1 8から送出されたこの代表フレームのフレーム番号を出力器5 1 6に出力する(ステップS 5 2 0 4)。次に、上記各部によりこの代表フレームに対してフレーム内符号化(イントラ符号化)を行ない(ステップS 5 2 0 5)、更に符号化結果を復号して復元画像を生成し、これをフレームメモリ5 9 ~ 5 1 2のうち、上記

30

条件に従ったフレームメモリに格納する(ステップS 5 2 0 6)。上述の通り、このようにフレームメモリに格納した代表フレームの画像データは、以降符号化されるフレームを符号化する際に参照されうるものである。

【0 2 8 5】

一方、現在のカウンタ値をMで割った余りが1にはならない場合(すなわちステップS 5 4でフレームメモリ5 2 0 0から出力されたフレームのフレーム番号が $(nM + 1)$ ではない場合)、ステップS 5 4でフレームメモリ5 2 0 0から出力されたフレームは代表フレームではないと判定し、処理をステップS 5 2 0 3に進め、上記各部によりこの非代表フレームに対してフレーム間符号化を行なう(ステップS 5 2 0 3)。

【0 2 8 6】

40

符号化された対象が代表フレーム、非代表フレームの何れであっても、その符号化結果はバッファ5 6に出力されるので、出力器5 1 6は、この符号化結果に代表フレームフラグ発生器5 2 2 4から出力された「代表フレームのフレーム番号」を付加して外部に出力する(ステップS 5 2 0 6)。

【0 2 8 7】

このような構成と一連の選択動作により、一定間隔で代表フレームを設定することで符号化側のフレームメモリを削減することが可能になり、回路を小型化できコストを下げるといった効果がある。

【0 2 8 8】

[第8の実施形態]

50

本実施形態に係る動画像符号化装置は、第3の実施形態と同様の構成を有する装置である。以下では、本実施形態に係る動画像符号化装置によって、第3の実施形態に係る動画像符号化処理とは異なる動画像符号化処理を実行する場合について説明する。

【0289】

本実施形態に係る動画像符号化装置は第3の実施形態と同様に、カメラ306により撮像され、記憶装置305に記録された動画像を符号化し、符号化結果をメモリ301に格納する、もしくは通信I/F308を介してネットワーク309に送出する。なお、符号化方式としてH.264符号化方式を例にとって説明するがこれに限定されず、2フレーム以上の複数フレームを参照して符号化する符号化方式であれば構わない。また、説明を容易にするために前後3フレームずつを参照することとするが、これに限定されない。

10

【0290】

端末装置303に備わっている不図示のキーボードやマウスを用いて符号化指示を入力すると、この指示はCPU300に通知され、CPU300は各部の初期化を行なう。またCPU300は、CPU300に後述の動画像符号化処理を実行させるためのソフトウェア、CPU300に後述の通信処理を行わせるためのソフトウェア等のソフトウェアのプログラムやデータを記憶装置305から読み出してメモリ301にロードする。

【0291】

図18はメモリ301内のメモリマップの一例を示す図である。同図に示すように、メモリ301には、動画像符号化ソフトウェア、通信ソフトウェアを記憶するためのエリアが設けられているとともに、これらソフトウェアを制御するなどの処理をCPU300に実行させるためのOS（オペレーティングシステム）を記憶するためのエリアも備えている。また、後述する画像エリア（1～7）、符号エリア（1～3）、ワーキングエリアも備える。なお、符号エリアの数が3であるのは、参照するフレーム数を3としているためであって、参照するフレーム数が異なれば、符号エリアの数もそれに応じて異ならせる。

20

【0292】

図19は、CPU300が、この動画像符号化ソフトウェア、通信ソフトウェアを実行することでなされる処理のフローチャートである。

【0293】

先ず、以下行なう符号化方式が代表フレームを用いる符号化方式であることを示す情報や、参照するフレームの最大数情報を含むヘッダ情報を生成し、メモリ301に保持しておく（ステップS5301）。

30

【0294】

次に、全てのフレームの符号化が終了していない場合には処理をステップS5302からステップS5303に進め、3フレーム分の符号化が終了したか否かを判定する（ステップS5303）。3フレーム分の符号化処理を行っていない場合には処理をステップS5304に進め、記憶装置305から、カメラ306が撮像した動画像のうち、1フレーム分のデータを読み出して、メモリ301の画像エリアに格納する（ステップS5304）。ステップS5304における処理が最初の場合には、画像エリア1に格納する。

【0295】

次に、ステップS5304で入力したフレームデータをフレーム間符号化/フレーム内符号化の何れかでもって符号化する（ステップS5305）。フレーム間符号化/フレーム内符号化の何れを行なう場合であっても、上記実施形態のいずれかの方法を適用することができる。本実施形態では、フレーム間符号化でもって符号化したものとする。

40

【0296】

そしてステップS5305で符号化した結果を符号エリアに格納する（ステップS5306）。なお、ステップS5306における処理を最初に行った場合には、符号エリア1に格納する。

【0297】

次に、ステップS5305で符号化したフレームの画像を上記実施形態と同様の処理を行なうことで復号し、復号結果（復号画像）を画像エリア2～4のうち1つに格納する（

50

ステップS5307)。なお、格納する際、各画像エリアに格納されているフレームデータのフレーム番号を参照し、最も古いフレーム番号のフレームデータを格納している画像エリアに格納する。

【0298】

次に、ステップS5306で格納した画像と、ステップS5307でこの画像を復元した復元画像との歪みを計算する(ステップS5308)。この歪みを求めるための計算の一例として、例えば、2つの画像において位置的に対応する画素同士で画素値の差分の絶対値を計算し、画素毎に計算した差分の絶対値の総和値を計算することで、この1つの画像間の歪みを求める。なお、この歪みを求めるための方法については特に限定するものではないが、要は、一度符号化され、そして復号した画像が元の画像とどれだけかけ離れたものになっているのか(相違)を示す指標が値として得られればよい。

10

【0299】

そして処理をステップS5303に戻し、以降の処理を繰り返す。そして繰り返した結果、3フレーム分の符号化処理を行った場合、処理をステップS5303からステップS5309に進め、符号化した3フレーム分の各フレームについてステップS5308で求めた歪みを参照し、最も歪みの小さいフレームを、符号化した3フレームの各フレームを代表するフレームである代表フレームとして決定する(ステップS5309)。

【0300】

そして、決定した代表フレームのフレーム番号を、符号エリアに格納した各符号化データに付加する(ステップS5310)。

20

【0301】

そして、代表フレームの復号画像(画像エリア2~4のいずれかのメモリに格納されており、フレーム番号でもって一意に1つ選択できる)を画像エリア5~7内、最も古いフレーム番号のフレームを格納している画像エリアに格納する(ステップS5311)。

【0302】

そして、通信ソフトウェアを実行することにより、メモリ301上の各符号エリアに格納されている各フレームの符号化データを通信インターフェース308を介して通信回路309に出力する(ステップS5312)。なお、ステップS5312では、各フレームの符号化データの出力先はこれに限定するものではなく、メモリ301中の所定のエリアに格納するようにしても良いし、記憶装置304、305などに保存するようにしても良い。

30

【0303】

そして、次の3フレームを符号化するために処理をステップS5302に戻し、以降の処理を繰り返す。当然、次の3フレームが存在しない場合には同図に示したフローチャートに従った処理を終了する。

【0304】

このような一連の選択動作により、符号化による歪みが最も少ないフレームを代表フレームとして選択することで、符号化による画質の劣化を防ぎ、劣化を最小に抑える効果がある。

【0305】

なお、本実施形態では3フレームずつ符号化しているが、このフレーム数は特に限定するものではない。

40

【0306】

[第9の実施形態]

本実施形態は、第8の実施形態に係る動画像符号化装置による符号化データを復号する復号装置に関するものである。本実施形態に係る復号装置は、第3の実施形態と同様の構成を有する装置である。

【0307】

よって、本実施形態に係る復号装置と第8の実施形態に係る動画像符号化装置とを一体化し、同じ装置内で符号化、復号処理を行なうようにしても良い。その場合、第8の実施

50

形態では、符号化結果をメモリ 301 内の所定のエリアに記録するなど、装置内で符号化結果を保持しておく必要がある。

【0308】

また、本実施形態に係る復号装置と第 8 の実施形態に係る動画像符号化装置とを別個の装置にしてもよく、その場合、第 8 の実施形態では、符号化結果を通信インターフェース 308 を介して通信回路 309 に出力し、本実施形態に係る復号装置に入力する必要がある。

【0309】

しかし以下の説明は何れの場合であっても本質的な処理は同じである。

【0310】

図 20 はメモリ 301 内のメモリマップの一例を示す図である。同図に示すように、メモリ 301 には、H. 264 符号化データを復号する動画像復号ソフトウェア、通信ソフトウェアを記憶するためのエリアが設けられているとともに、これらソフトウェアを制御するなどの処理を CPU 300 に実行させるための OS (オペレーティングシステム) を記憶するためのエリアも備えている。また、後述する画像エリア (1 ~ 7)、符号エリア、ワーキングエリアも備える。なお、画像エリア 1 は復号画像を、画像エリア 2 ~ 7 は参照画像を格納する為のものである。なお、画像エリア 2 ~ 7 に画像を格納する際、画像エリア 2 ~ 7 の順に格納する。

【0311】

図 21 は、CPU 300 が、この動画像復号ソフトウェア、通信ソフトウェアを実行することでなされる処理のフローチャートである。

【0312】

先ず、符号化データに先立って外部から、もしくはメモリなどの内部から入力される上記ヘッダ情報を参照し、解析するとともに、各種の初期化処理を行なう (ステップ S 5401)。

【0313】

次に、全てのフレームの復号が終了したか否かを判断し (ステップ S 5402)、終了していなければ処理をステップ S 5403 に進め、各フレームの符号化データをメモリ 301 上の符号エリアに入力する (ステップ S 5403)。次に、この符号化データに付加されている、このフレームの代表フレームのフレーム番号を、この符号化データから分離

【0314】

そして入力された符号化データに対して第 6 の実施形態で説明したように復号処理を行ない、復号画像を画像エリア 1 に格納する (ステップ S 5405)。この復号処理の際、復号対象がフレーム間符号化によるものである場合には、画像エリア 2 ~ 7 に格納されている参照画像を参照して復号する。また、復号したフレームのフレーム番号をワーキングエリアに格納する。

【0315】

次に、ステップ S 5405 で復号したフレームが代表フレームであるか否かを、ワーキングエリアに書き込んだフレーム番号を参照し、ステップ S 5404 で分離したフレーム番号に一致するか否かで判断する (ステップ S 5406)。非代表フレームである場合には処理をステップ S 5407 に進め、画像エリア 2 ~ 4 のうち、最も古いフレーム番号のフレームデータを格納している画像エリアを特定し (ステップ S 5407)、特定した画像エリアに画像エリア 1 に格納されている画像を複写する (ステップ S 5408)。

【0316】

一方、代表フレームである場合には処理をステップ S 5409 に進め、画像エリア 5 ~ 7 のうち、最も古いフレーム番号のフレームデータを格納している画像エリアを特定し (ステップ S 5409)、特定した画像エリアに画像エリア 1 に格納されている画像を複写する (ステップ S 5410)。

【0317】

10

20

30

40

50

そして、画像エリア 1 に格納されている復号画像データを出力する（ステップ S 5 4 1 1）。出力先は特に限定しないが、例えば表示装置 3 0 7 に出力し、表示しても良い。

【 0 3 1 8 】

そして、処理をステップ S 5 4 0 2 に戻し、以降の処理を繰り返す。

【 0 3 1 9 】

このような一連の選択動作により、時間的に遠いフレームを代表フレームとして保持し、時間的に近いフレームの両方を参照することで、少ない情報量で再生画像を得ることが可能になるという効果がある。

【 0 3 2 0 】

なお、本実施形態においては被参照情報を参照された最後のフレームの位置としたがこれに限定されず、最終的に参照された時刻などの情報でももちろんかまわない。

【 0 3 2 1 】

[第 1 0 の実施形態]

図 2 3 は本実施形態に係る動画像符号化装置の構成を示す図である。本実施形態では、H . 2 6 4 符号化方式をベースにとって説明するが、H . 2 6 4 と異なり、参照フレームは時間的に不連続のものを参照することになる。但し、ベースは H . 2 6 4 に限定されない。また、説明を簡単にするため、過去のフレームを参照する前方向予測を例に取るが、これに限定されず、両方向予測に適用してももちろん構わない。また、以下の説明では、フレームの画像データをフレームデータと呼称する場合もある。また、このフレームデータ（フレームの画像）には、このフレームのフレーム番号を示すデータも含まれているものとする。

【 0 3 2 2 】

図 2 3 において、6 1 9 ~ 6 2 1 は外部から入力された動画像の各フレームの画像（入力画像）のうち、後述する処理により選択された代表フレームの画像を格納するためのフレームメモリである。6 1 はフレーム内符号化 / フレーム間符号化のモードに従って出力先を選択するセレクタである。

【 0 3 2 3 】

6 2 は H . 2 6 4 符号化方式によるイントラ予測を行なうイントラ予測器であり、6 3 は動き予測誤差を求める差分器である。6 4 は整数型直交変換を行ない、入力された係数を量子化する変換 / 量子化器であり、6 7 はその逆の作用を施す逆量子化 / 逆変換器である。6 5 は変換 / 量子化器 6 4 による量子化結果を符号化するエントロピー符号化器であり、6 6 は発生した複数フレーム分の符号を格納するバッファである。

【 0 3 2 4 】

6 9、6 1 0、6 1 1、6 1 2 はフレームメモリであり、局所復号された画像データとフレームの番号をフレーム単位で格納する。6 1 3 は入力画像と復号された画像データから最適な動きベクトルを該当するフレームから抽出する動き推定器であり、6 1 4 は動き推定器 6 1 3 で算出された動きベクトルと該当するフレームの情報から予測画像を生成する動き補償器である。6 1 5 は算出された動きベクトルと該当するフレームの情報から動きの情報を符号化する動き符号化器である。

【 0 3 2 5 】

6 1 7 はフレームメモリ 6 9 ~ 6 1 2 の入出力を制御するフレームメモリ制御器であり、6 1 6 はバッファ 6 6 に格納された符号化データを出力する出力器である。6 1 8 は入力画像と代表フレームの画像とを比較して類似度を算出する類似度判定器である。6 2 2 は、代表フレームのフレーム番号を出力器 6 1 6 に出力する代表フレームフラグ発生器である。

【 0 3 2 6 】

上記のように構成された動画像符号化装置における動画像の符号化動作を以下、説明する。

【 0 3 2 7 】

符号化に先立ち、本符号化データが代表フレームを参照する方式であることを示す符号

10

20

30

40

50

を代表フレームフラグ発生器 6 2 2 が生成して出力器 6 1 6 に出力する。この符号は H . 2 6 4 符号化方式と互換性を保つために、ユーザデータとして符号化されることとするがこれに限定されず、電子的な透かしとして埋め込みを行なってももちろん構わない。

【 0 3 2 8 】

さらに、参照に用いることができるフレーム数などを含むヘッダ情報を出力器 6 1 6 が生成して出力する。このとき、参照に用いることのできるフレーム数は 3 としておく。また、各フレームメモリは 0 にリセットされる。

【 0 3 2 9 】

動画像を構成する各フレームの画像（入力画像）は順次セレクタ 6 1、類似度判定器 6 1 8 に入力される。類似度算出器 6 1 8 は入力画像と、フレームメモリ 6 1 9、6 2 0、6 2 1 に格納されている代表フレームの画像データと比較を行ない、類似度を算出する。本実施形態ではこの類似度の計算方法として、入力画像と代表フレームの画像とで位置的に対応する画素同士の画素値の差分の絶対値を計算し、そして画素毎に計算した絶対値の総和を計算することで求めるが、計算方法はこれに限定するものではなく、他の計算方法を用いても良い。

【 0 3 3 0 】

このようにして類似度判定器 6 1 8 は入力画像と、3つの代表フレームの画像それぞれに対する類似度を求め、求めた3つの類似度のうち最小の類似度と所定の閾値との比較処理を行ない、最小の類似度が所定の閾値以上である場合には、この入力画像を代表フレームの画像として判定する。一方、最小の類似度が所定の閾値以下である場合には、この入力画像を非代表フレームの画像として判定する。

【 0 3 3 1 】

セレクタ 6 1 は、類似度判定器 6 1 8 が代表フレームと判定した入力画像に対しては、この入力画像の出力先をイントラ予測器 6 2 に切り替え、類似度判定器 6 1 8 が非代表フレームと判定した入力画像に対しては、この入力画像の出力先を差分器 6 3 に切り替える。

【 0 3 3 2 】

また、類似度判定器 6 1 8 は、入力画像が代表フレームの画像であると判定した場合には、この入力画像を、フレームメモリ 6 1 9 ~ 6 2 1 のうち、最も古いフレーム番号のフレームデータを格納しているフレームメモリに格納する。

【 0 3 3 3 】

また、類似度判定器 6 1 8 は、入力画像が代表フレームの画像であると判定した場合には、この入力画像のフレーム番号を代表フレームフラグ発生器 6 2 2 に通知し、代表フレームフラグ発生器 6 2 2 は、通知されたフレーム番号を出力器 6 1 6 に出力する。なお、代表フレームフラグ発生器 6 2 2 は、通知された最新のフレーム番号を保持しており、新たに通知されるまで、現在保持しているフレーム番号を出力する。

【 0 3 3 4 】

まず、代表フレームの画像データに対してフレーム内符号化を行なう場合について説明する。

【 0 3 3 5 】

入力画像はマクロブロック単位でイントラ予測器 6 2 に入力され、ブロック毎に予測が行なわれる。その結果は変換 / 量子化器 6 4 に入力され、整数型直交変換を行ない、その係数を量子化する。その結果はエントロピー符号化器 6 5 と逆量子化 / 逆変換器 7 に入力される。

【 0 3 3 6 】

エントロピー符号化器 6 5 では、入力された量子化結果をエントロピー符号化し、バッファ 6 6 にフレーム単位で格納する。逆量子化 / 逆変換器 6 7 では、入力された量子化結果から復号画像を得る。この復号画像は、フレームメモリ制御器 6 1 7 の指示に従って、フレームメモリ 6 9 ~ 6 1 2 のうち、最も古いフレーム番号の代表フレームを格納しているフレームメモリに格納される。

【0337】

一方、バッファ66に格納された符号化データは出力器616によって、代表フレームフラグ発生器622からの「代表フレームのフレーム番号」が付加されて外部に出力される。

【0338】

一方、非代表フレームに対してフレーム間符号化を行なう場合について説明する。入力画像データはマクロブロック単位でセクタ61と動き推定器613に入力される。動き推定器613はフレームメモリ69～612に保持されている画像を読み出し、第1の実施形態と同様にして動きベクトルを求める。そして動き補償器614は第1の実施形態と同様にして予測画像を生成し、差分器63に入力する。

10

【0339】

差分器63には上述の通りセクタ61から非代表フレームの入力画像が入力されるので、これと予測画像との差分、すなわち予測誤差を求める。求めた予測誤差は後段の変換/量子化器64に輸入され、整数型直交変換によって量子化され、更にその量子化結果はエントロピー符号化器65でエントロピー符号化が施される。そしてエントロピー符号化結果はバッファ66にフレーム単位で格納される。また、動きベクトルについても第1の実施形態と同様にして動き符号化器615で符号化され、バッファ66に格納される。

【0340】

そして、バッファ66に格納された符号化データは出力器616により、代表フレームフラグ発生器622から出力する代表フレームのフレーム番号とともに出力される。

20

【0341】

図24は、以上説明した、本実施形態に係る動画像符号化処理のフローチャートである。

【0342】

先ず、出力器616は、以下行なう符号化方式が代表フレームを用いる符号化方式であることを示す情報や、参照するフレームの最大数情報を含むヘッダ情報を生成して出力する(ステップS61)。次に、符号化すべき全てのフレームについて符号化処理を行ったか否かを判断する(ステップS62)。符号化すべきフレームがまだ残っている場合には処理をステップS63に進め、類似度判定器618、セクタ61に符号化対象のフレームデータを順に輸入する(ステップS63)。そして類似度判定器618は上述の通り、フレームメモリ619～621に格納されている各代表フレームの画像と入力画像との類似度を求め(ステップS64)、最小の類似度が所定の閾値以下であるか否かを判断する(ステップS65)。

30

【0343】

最小の類似度が所定の閾値以上である場合には処理をステップS66に進め、入力画像を代表フレームの画像であると判断し、この入力画像に対して上述したようにフレーム内符号化処理を行なう(ステップS66)。この符号化処理には第1の実施形態と同様に、符号化した入力画像を復号し、フレームメモリ69～612のうち、最も古いフレーム番号のフレームを格納しているフレームメモリに格納する処理を含む。

【0344】

また、代表フレームフラグ発生器622は、この入力画像のフレーム番号を出力器616に出力する(ステップS67)。また、ステップS63で入力した入力画像を、フレームメモリ619～621のいずれかに格納する(ステップS68)。

40

【0345】

一方、最小の類似度が所定の閾値以下である場合には処理をステップS69に進め、入力画像を非代表フレームの画像であると判断し、この入力画像に対して上述したようにフレーム間符号化処理を行なう(ステップS69)。ステップS66,ステップS69の何れの符号化結果も上述の通り、バッファ66に出力されるので、出力器616はこの符号化結果に代表フレームフラグ発生器622から出力されたフレーム番号を付加し(ステップS611)、外部に出力する(ステップS612)。

50

【 0 3 4 6 】

このような一連の選択動作により、少ないフレーム数で長い時間のフレームを参照することが可能になり、符号化時にメモリ容量を抑制するといった効果がある。また、代表フレームをフレーム内符号化することで、エラーの伝播による画質の劣化を防ぎ、劣化を最小に抑える効果がある。

【 0 3 4 7 】

また、符号化前の入力画像で類似度を比較することで、符号化による歪の影響を受けずに代表画像を選択することができる。

【 0 3 4 8 】

なお、本実施形態においては代表フレームのみをフレーム内符号化したが、これに限定されず、他のフレーム、例えば、代表フレーム以外を一定間隔でフレーム内符号化してももちろんかまわない。

【 0 3 4 9 】

また、符号化単位なども以上のようにフレーム単位に限定するものではない。

【 0 3 5 0 】

[第 1 1 の実施形態]

図 2 5 は本実施形態に係る動画像符号化装置の構成を示す図である。図 2 3 と同じ部分については同じ番号を付けており、その説明を省略する。

【 0 3 5 1 】

6 2 0 0 は符号化しようとする動画像を構成する各フレームの画像（入力画像）を保持するフレームメモリである。6 2 1 3 は動き推定器であり、入力画像データとフレームメモリ 6 9 ~ 6 1 2 の画像データとを比較して動きベクトルとその参照したフレーム番号を出力する。また、そのときの誤差も出力する。

【 0 3 5 2 】

6 2 0 2 は類似度判定器で、動き推定器 6 2 1 3 から動きベクトル探索時の誤差を入力し、後述の類似度を算出し、入力画像が代表フレームであるか否かを判定する。6 2 0 1 はセレクタで通常の出力先は差分器 6 3 とするが、類似度判定器 6 2 0 2 が代表フレームであると最後に判定したタイミングから予め決められた間隔毎には出力先をイントラ予測器 6 2 に切り替える。

【 0 3 5 3 】

以上の構成を備える、本実施形態に係る動画像符号化装置が行なう処理について以下、説明する。

【 0 3 5 4 】

第 1 0 の実施形態と同様に、符号化に先立ち、本符号化データが代表フレームを参照する方式であることを示す符号を代表フレームフラグ発生器 6 2 2 が生成して出力器 6 1 6 に出力する。さらに、参照に用いることができるフレーム数などを含むヘッダ情報を出力器 6 1 6 が生成して出力する。このとき、参照に用いることのできるフレーム数は 1 5 としておく。また、各フレームメモリは 0 にリセットされる。また、フレームカウンタ 6 2 0 2 が保持するカウント値を 0 にリセットする。

【 0 3 5 5 】

そして先ずフレームメモリ制御器 6 1 7 は、フレームメモリ 6 9 ~ 6 1 2 が保持する各フレームのうち、フレームメモリ 6 2 0 0 から入力される入力画像から 1 5 フレーム以上離れているフレームについては、削除する。これは、フレームメモリ 6 9 ~ 6 1 2 が保持する各フレームのフレーム番号、フレームメモリ 6 2 0 0 から入力される入力画像のフレーム番号を参照し、入力画像のフレーム番号から 1 5 以上異なるフレーム番号を有するフレームデータを削除することによりなされる処理である。これにより、フレームメモリ 6 2 0 0 から入力される入力画像から 1 5 フレーム以上離れているフレームについては動き推定器 6 2 1 3 は参照することができなくなる。

【 0 3 5 6 】

そしてフレームメモリ 6 2 0 0 には、各フレームの画像データが入力され、各フレーム

10

20

30

40

50

の画像は後述の符号化に先立ち、動き推定器 6 1 3 に入力される。動き推定器 6 1 3 は、フレームメモリ 6 9 ~ 6 1 2 に格納されている全ての画像に対してそれぞれのマクロブロック単位で動きベクトルの算出を行なう。

【 0 3 5 7 】

動きベクトルの算出はブロックマッチング法で行なわれる。これは参照画像の該当する部分と入力部分の画素単位での差異の絶対値和が最小となる画素位置の差異を動きベクトルとする方式である。その際の最小の際の絶対値和を誤差として類似度判定器 6 2 0 2 に入力する。

【 0 3 5 8 】

類似度判定器 6 2 0 2 は、フレームごとに発生した誤差の絶対値の総和を求める。この総和を類似度とする。誤差の絶対値の総和が予め決められた閾値よりも小さければ、入力画像は非代表フレームと判断し、誤差の絶対値の総和が予め決められた閾値よりも大きければ、この入力画像を代表フレームと判断する。

10

【 0 3 5 9 】

また、類似度判定器 6 2 0 2 は、入力画像が代表フレームであると判断した際には、この入力画像のフレーム番号を代表フレームフラグ発生器 6 2 2 に通知するので、代表フレームフラグ発生器 6 2 2 は通知されたフレーム番号を後段の出力器 6 1 6 に出力する。

【 0 3 6 0 】

また、代表フレームの画像、又は一定間隔でフレーム内符号化を行なう場合、第 1 0 の実施形態と同様に、入力画像はマクロブロック単位でイントラ予測器 6 2 に入力され、ブロック毎に予測が行なわれる。その結果は変換 / 量子化器 6 4 で整数型直交変換と量子化が行なわれ、その結果はエントロピー符号化器 6 5 と逆量子化 / 逆変換器 6 7 に入力される。

20

【 0 3 6 1 】

エントロピー符号化器 6 5 は、入力された量子化結果をエントロピー符号化し、その符号化結果をバッファ 6 6 にフレーム単位で格納する。逆量子化 / 逆変換器 6 7 は、入力された量子化結果から第 1 0 の実施形態と同様にして復号画像を得る。

【 0 3 6 2 】

フレームメモリ制御器 6 1 7 には、類似度判定器 6 2 0 2 による判断結果が入力されるので、逆量子化 / 逆変換器 6 7 からの復号画像が代表フレームのものであるか否かを判断することができる。よって復号画像が代表フレームのものである場合には、この入力画像を、フレームメモリ 6 9 ~ 6 1 2 のうち、最も古いフレーム番号のフレームを格納しているフレームメモリに格納する。

30

【 0 3 6 3 】

そして以上の処理によりバッファ 6 6 に格納された符号化データは、出力器 6 1 6 により、代表フレームフラグ発生器 6 2 2 からの「フレーム番号」が付加されて外部に出力される。

【 0 3 6 4 】

一方、非代表フレームに対してフレーム間符号化を行なう場合にも、第 1 0 の実施形態と同様である。すなわち、非代表フレームの入力画像データはマクロブロック単位でセクタ 6 2 0 1 と動き推定器 6 2 1 3 に入力される。動き推定器 6 2 1 3 はフレームメモリ 6 9 ~ 6 1 2 に保持されている画像を読み出し、第 1 0 の実施形態と同様にして動きベクトルを求める。そして動き補償器 6 1 4 は第 1 0 の実施形態と同様にして予測画像を生成し、差分器 6 3 に入力する。

40

【 0 3 6 5 】

差分器 6 3 には上述の通りセクタ 6 2 0 1 から非代表フレームの入力画像が入力されるので、これと予測画像との差分、すなわち予測誤差を求める。求めた予測誤差は後段の変換 / 量子化器 6 4 に入力され、整数型直交変換によって量子化され、更にその量子化結果はエントロピー符号化器 6 5 でエントロピー符号化が施される。そしてエントロピー符号化結果はバッファ 6 6 にフレーム単位で格納される。また、動きベクトルについても第

50

10の実施形態と同様にして動き符号化器615で符号化され、バッファ66に格納される。

【0366】

そして、バッファ66に格納された符号化データは出力器616により、代表フレームフラグ発生器622から出力する代表フレームのフレーム番号とともに出力される。

【0367】

図26は、以上説明した、本実施形態に係る動画像符号化処理のフローチャートである。なお、同図において、図24と同じ処理ステップには同じステップ番号を付けており、その説明を省略する。

【0368】

ステップS6200では上述の通り、フレームメモリ制御器617は、フレームメモリ69~612が保持する各フレームのうち、フレームメモリ6200から入力される入力画像から15フレーム以上離れているフレームについては、削除する。

【0369】

またステップS6201では、図24におけるステップS68とは格納先が異なるのみでそれ以外は同じ処理である。すなわちステップS6201では、代表フレームの画像をフレームメモリ69~612に格納する。

【0370】

このような構成と一連の選択動作により、局所復号画像を参照することで符号化側のフレームメモリを削減することが可能になり、回路を小型化できコストを下げるといった効果がある。

【0371】

また、類似度の算出で動き補償後の誤差を使用することで、画像の動きを考慮した類似度を判定でき、精度を向上することができる。

【0372】

また、代表フレームをフレーム内符号化することで、エラーの伝播による画質の劣化を防ぎ、劣化を最小に抑える効果がある。

【0373】

また、参照フレーム数を限定することで既存のH.264と互換性を保つことが可能となり、第6の実施形態に係る復号装置のみならず、既存の復号装置でも復号が可能になる。

【0374】

[第12の実施形態]

本実施形態に係る動画像符号化装置は、第3の実施形態と同様の構成を有する装置である。以下では、本実施形態に係る動画像符号化装置によって、第3の実施形態に係る動画像符号化処理とは異なる動画像符号化処理を実行する場合について説明する。

【0375】

本実施形態に係る動画像符号化装置は第3の実施形態と同様に、カメラ306により撮像され、記憶装置305に記録された動画像を符号化し、符号化結果をメモリ301に格納する、もしくは通信I/F308を介してネットワーク309に送出する。なお、符号化方式としてH.264符号化方式を例にとって説明するがこれに限定されず、2フレーム以上の複数フレームを参照して符号化する符号化方式であれば構わない。また、説明を容易にするために前後3フレームずつを参照することとするが、これに限定されない。

【0376】

端末装置303に備わっている不図示のキーボードやマウスを用いて符号化指示を入力すると、この指示はCPU300に通知され、CPU300は各部の初期化を行なう。またCPU300は、CPU300に後述の動画像符号化処理を実行させるためのソフトウェア、CPU300に後述の通信処理を行わせるためのソフトウェア等のソフトウェアのプログラムやデータを記憶装置305から読み出してメモリ301にロードする。

【0377】

10

20

30

40

50

本実施形態に係るメモリ 301 内のメモリマップの一例としては図 20 に示したものが適用可能である。

【0378】

図 27 は、CPU 300 が、この動画像符号化ソフトウェア、通信ソフトウェアを実行することでなされる処理のフローチャートである。

【0379】

先ず、以下行なう符号化方式が代表フレームを用いる符号化方式であることを示す情報や、参照するフレームの最大数情報を含むヘッダ情報を生成し、メモリ 301 に保持しておく（ステップ S6301）。

【0380】

次に、全てのフレームの符号化が終了していない場合には処理をステップ S6302 からステップ S6303 に進め、記憶装置 305 から、カメラ 306 が撮像した動画像のうち、1 フレーム分のデータを読み出して、メモリ 301 の画像エリア 1 に格納する（ステップ S6303）。

【0381】

そして画像エリア 4 ~ 7 に格納されているそれぞれの画像について、画像エリア 1 に格納されている画像との類似度を計算する（ステップ S6304）。この類似度の計算方法については特に限定しないが、本実施形態では 2 つの画像間で位置的に対応する画素同士の画素値の差分の絶対値を計算し、それぞれの画素について計算した絶対値の総和値を求め、これをこの 2 つの画像間の類似度とする。

【0382】

そして画像エリア 4 ~ 7 に格納されているそれぞれの画像について求めた類似度のうち、最小の類似度の値が所定の閾値以下であるか否かを判断する（ステップ S6305）。最小の類似度の値が所定の閾値以下であれば処理をステップ S6306 に進め、画像エリア 1 に格納されている画像は代表フレームにおける画像として判断し、この代表フレームのフレーム番号をメモリ 301 中の所定のエリア（例えばワーキングエリア）に出力（格納）する（ステップ S6306）。そして上記実施形態と同様にこの代表フレームの画像に対してフレーム内符号化を施し、その符号化結果を符号エリアに格納する（ステップ S6307）。

【0383】

そして符号化したフレームの画像を上記実施形態と同様の処理を行なうことで復号し、復号結果（復号画像）を画像エリア 5 以降のうち 1 つに格納する（ステップ S6308）。なお、格納する際、各画像エリアに格納されているフレームデータのフレーム番号を参照し、最も古いフレーム番号のフレームデータを格納している画像エリアに格納する。

【0384】

一方、最小の類似度の値が所定の閾値以上であれば処理をステップ S6309 に進め、画像エリア 1 に格納されている画像は非代表フレームにおける画像として判断し、そして上記実施形態と同様にこの代表フレームの画像に対してフレーム内符号化もしくはフレーム間符号化を施し、その符号化結果を符号エリアに格納する（ステップ S6309）。

【0385】

そして符号化したフレームの画像を上記実施形態と同様の処理を行なうことで復号し、復号結果（復号画像）を画像エリア 2 ~ 4 のうち 1 つに格納する（ステップ S6310）。なお、格納する際、各画像エリアに格納されているフレームデータのフレーム番号を参照し、最も古いフレーム番号のフレームデータを格納している画像エリアに格納する。

【0386】

そして、通信ソフトウェアを実行することにより、メモリ 301 上の各符号エリアに格納されている各フレームの符号化データを、ステップ S6306 で格納したフレーム番号を付加して、通信インターフェース 308 を介して通信回路 309 に出力する（ステップ S6312）。なお、ステップ S6312 では、各フレームの符号化データの出力先はこ

10

20

30

40

50

れに限定するものではなく、メモリ 301 中の所定のエリアに格納する要にしても良いし、記憶装置 304、305 などに保存するようにしても良い。

【0387】

そして、次のフレームを符号化するために処理をステップ S 6302 に戻し、以降の処理を繰り返す。当然、次のフレームが存在しない場合には同図に示したフローチャートに従った処理を終了する。

【0388】

このような一連の選択動作により、時間的に近いフレームと時間的に遠い代表フレームを参照することで、時間的に近くて、類似度の高いフレームも参照が可能になり、符号化効率を改善できる効果がある。

10

【0389】

また、代表フレームをフレーム内符号化することで、エラーの伝播による画質の劣化を防ぎ、劣化を最小に抑える効果がある。

【0390】

[第13の実施形態]

本実施形態は、第12の実施形態に係る動画像符号化装置による符号化データを復号する復号装置に関するものである。本実施形態に係る復号装置は、第3の実施形態と同様の構成を有する装置である。

【0391】

よって、本実施形態に係る復号装置と第12の実施形態に係る動画像符号化装置とを一体化し、同じ装置内で符号化、復号処理を行なうようにしても良い。その場合、第13の実施形態では、符号化結果をメモリ 301 内の所定のエリアに記録するなど、装置内で符号化結果を保持しておく必要がある。

20

【0392】

また、本実施形態に係る復号装置と第12の実施形態に係る動画像符号化装置とを別個の装置にしてもよく、その場合、第13の実施形態では、符号化結果を通信インターフェース 308 を介して通信回路 309 に出力し、本実施形態に係る復号装置に入力する必要がある。

【0393】

しかし以下の説明は何れの場合であっても本質的な処理は同じである。

30

【0394】

本実施形態に係るメモリ 301 内のメモリマップの一例としては図 20 に示したものが適用可能である。

【0395】

図 28 は、CPU 300 が、動画像復号ソフトウェア、通信ソフトウェアを実行することでなされる処理のフローチャートである。

【0396】

まず、符号化データに先立って外部から、もしくはメモリなどの内部から入力される上記ヘッダ情報を参照し、解析するとともに、各種の初期化処理を行なう（ステップ S 6401）。

40

【0397】

次に、全てのフレームの復号が終了したか否かを判断し（ステップ S 6402）、終了していなければ処理をステップ S 6403 に進め、各フレームの符号化データをメモリ 301 上の符号エリアに入力する（ステップ S 6403）。次に、この符号化データに付加されている、このフレームの代表フレームのフレーム番号を、この符号化データから分離する（ステップ S 6404）。

【0398】

そして入力された符号化データに対して第6の実施形態で説明したように復号処理を行ない、復号画像を画像エリア1に格納する（ステップ S 6405）。この復号処理の際、復号対象がフレーム間符号化によるものである場合には、画像エリア2～7に格納されて

50

いる参照画像を参照して復号する。また、復号したフレームのフレーム番号をワーキングエリアに格納する。

【0399】

次に、ステップS6405で復号したフレームが代表フレームであるか否かを、ワーキングエリアに書き込んだフレーム番号を参照し、ステップS6404で分離したフレーム番号に一致するか否かで判断する(ステップS6406)。非代表フレームである場合には処理をステップS6407に進め、画像エリア2~4のうち、最も古いフレーム番号のフレームデータを格納している画像エリアを特定し(ステップS6407)、特定した画像エリアに画像エリア1に格納されている画像を複写する(ステップS6408)。

【0400】

一方、代表フレームである場合には処理をステップS6409に進め、画像エリア5~7のうち、最も古いフレーム番号のフレームデータを格納している画像エリアを特定し(ステップS6409)、特定した画像エリアに画像エリア1に格納されている画像を複写する(ステップS6410)。

【0401】

そして、画像エリア1に格納されている復号画像データを出力する(ステップS6411)。出力先は特に限定しないが、例えば表示装置307に出力し、表示しても良い。

【0402】

そして、処理をステップS6402に戻し、以降の処理を繰り返す。

【0403】

このような一連の選択動作により、時間的に遠いフレームを代表フレームとして保持し、時間的に近いフレームの両方を参照することで、少ない情報量で再生画像を得ることが可能になるという効果がある。

また、代表フレームをフレーム内符号化することで、高い画質のフレームを参照画像データとすることで画質が向上するという効果がある。

【0404】

[その他の実施形態]

本発明の目的は、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記録媒体(または記憶媒体)を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ(またはCPUやMPU)が記録媒体に格納されたプログラムコードを読み出し実行することによっても、達成されることは言うまでもない。この場合、記録媒体から読み出されたプログラムコード自体が前述した実施形態の機能を実現することになり、そのプログラムコードを記録した記録媒体は本発明を構成することになる。

【0405】

また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているオペレーティングシステム(OS)などが実際の処理の一部または全部を行ない、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0406】

さらに、記録媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張カードやコンピュータに接続された機能拡張ユニットに備わるメモリに書込まれた後、そのプログラムコードの指示に基づき、その機能拡張カードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行ない、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0407】

本発明を上記記録媒体に適用する場合、その記録媒体には、先に説明したフローチャートに対応するプログラムコードが格納されることになる。

【図面の簡単な説明】

【 0 4 0 8 】

【図 1】本発明の第 1 の実施形態に係る動画像符号化装置の構成を示す図である。

【図 2】本発明の第 1 の実施形態に係る符号化処理のフローチャートである。

【図 3】本発明の第 2 の実施形態に係る動画像符号化装置の構成を示す図である。

【図 4】本発明の第 2 の実施形態に係る符号化処理のフローチャートである。

【図 5】本発明の第 3 の実施形態に係る動画像符号化装置の基本構成を示すブロック図である。

【図 6】メモリ 3 0 1 内のメモリマップの一例を示す図である。

【図 7】CPU 3 0 0 が、本発明の第 3 の実施形態に係る動画像符号化ソフトウェア、通信ソフトウェアを実行することでなされる処理のフローチャートである。

10

【図 8】ステップ S 2 0 5 における処理の詳細を示すフローチャートである。

【図 9】区間を表すテーブルの内容を示す図である。

【図 1 0】本発明の第 4 の実施形態に係る動画像符号化処理のフローチャートである。

【図 1 1】H . 2 6 4 の符号化器の構成を示すブロック図である。

【図 1 2】本発明の第 5 の実施形態に係る動画像符号化装置の構成を示すブロック図である。

【図 1 3】本発明の第 5 実施形態に係る符号化処理のフローチャートである。

【図 1 4】本発明の第 6 の実施形態に係る復号装置の構成を示す図である。

【図 1 5】本発明の第 6 の実施形態に係る復号処理のフローチャートである。

【図 1 6】本発明の第 7 の実施形態に係る動画像符号化装置の構成を示す図である。

20

【図 1 7】本発明の第 7 の実施形態に係る符号化処理のフローチャートである。

【図 1 8】本発明の第 8 の実施形態におけるメモリ 3 0 1 内のメモリマップの一例を示す図である。

【図 1 9】CPU 3 0 0 が、本発明の第 8 の実施形態に係る動画像符号化ソフトウェア、通信ソフトウェアを実行することでなされる処理のフローチャートである。

【図 2 0】本発明の第 9 の実施形態に係るメモリ 3 0 1 内のメモリマップの一例を示す図である。

【図 2 1】CPU 3 0 0 が、本発明の第 9 の実施形態に係る動画像復号ソフトウェア、通信ソフトウェアを実行することでなされる処理のフローチャートである。

【図 2 2】H . 2 6 4 の従来の復号器の構成を示す図である。

30

【図 2 3】本発明の第 1 0 の実施形態に係る動画像符号化装置の構成を示す図である。

【図 2 4】本発明の第 1 0 の実施形態に係る動画像符号化処理のフローチャートである。

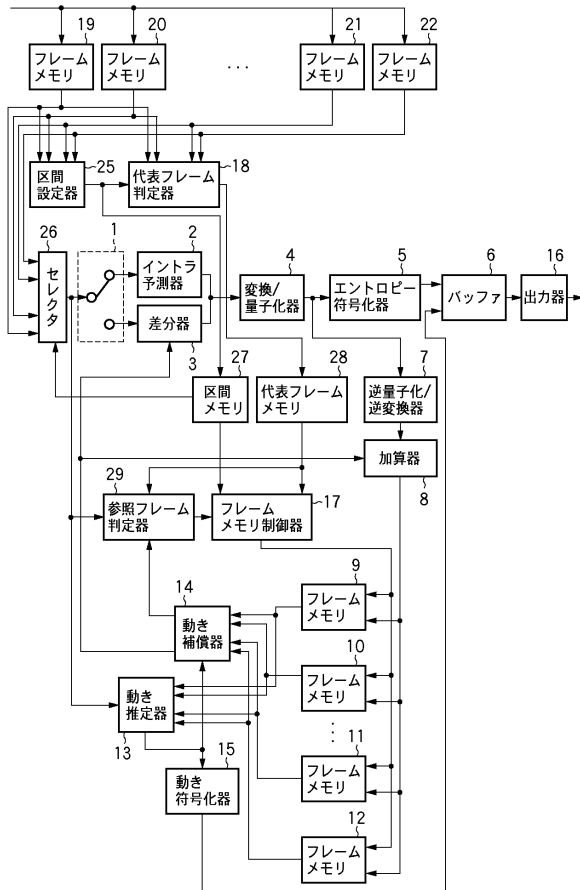
【図 2 5】本発明の第 1 1 の実施形態に係る動画像符号化装置の構成を示す図である。

【図 2 6】本発明の第 1 1 の実施形態に係る動画像符号化処理のフローチャートである。

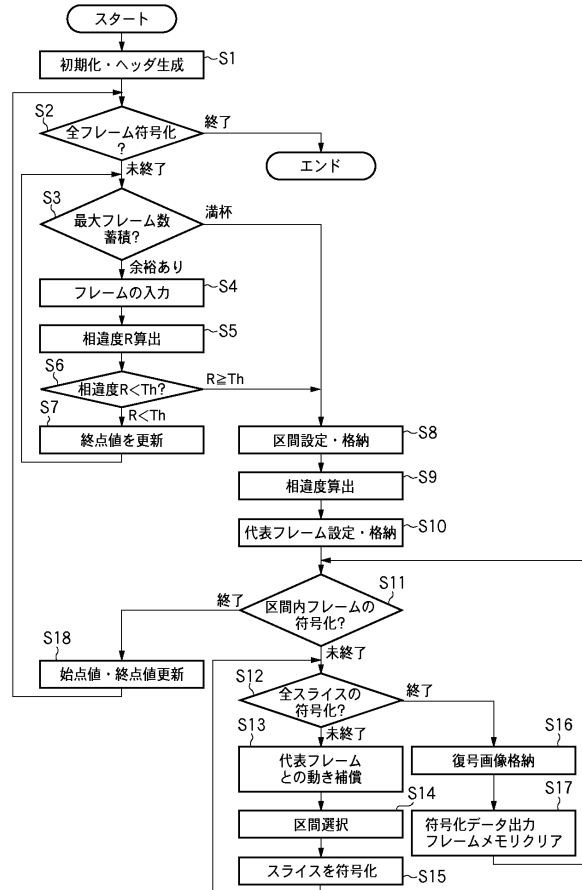
【図 2 7】CPU 3 0 0 が、本発明の第 1 2 の実施形態に係る動画像符号化ソフトウェア、通信ソフトウェアを実行することでなされる処理のフローチャートである。

【図 2 8】CPU 3 0 0 が、本発明の第 1 3 の実施形態に係る動画像復号ソフトウェア、通信ソフトウェアを実行することでなされる処理のフローチャートである。態にかかる符号化処理のフローチャートである。

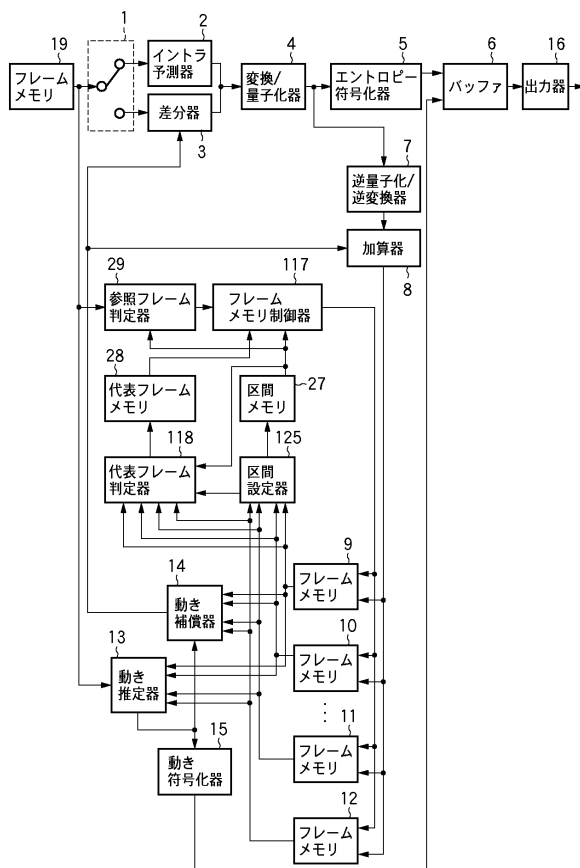
【 図 1 】



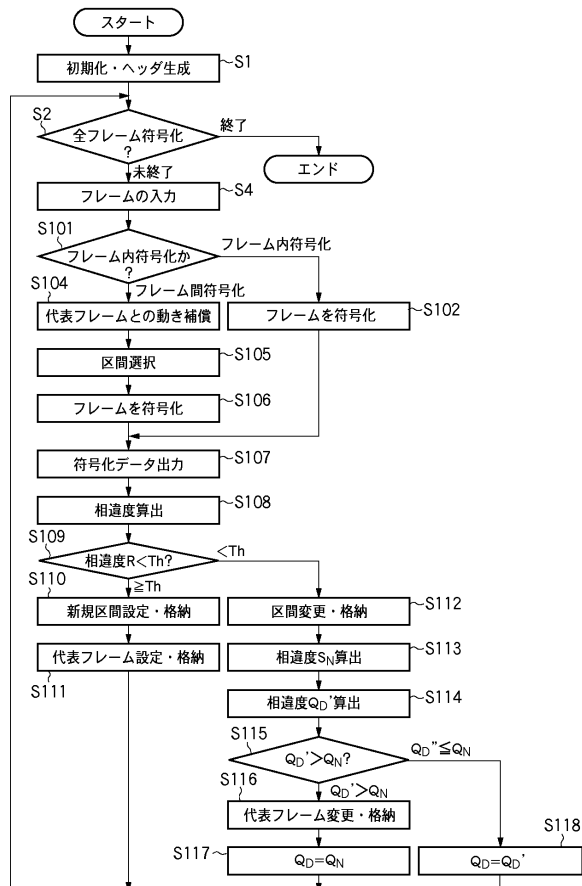
【 図 2 】



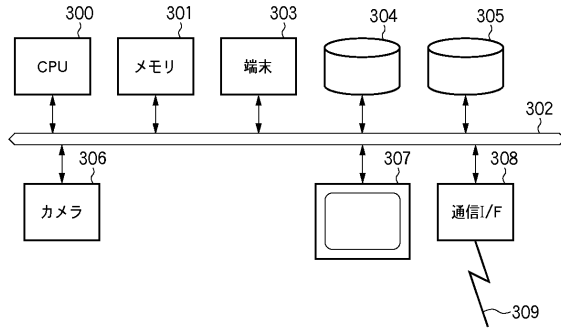
【 図 3 】



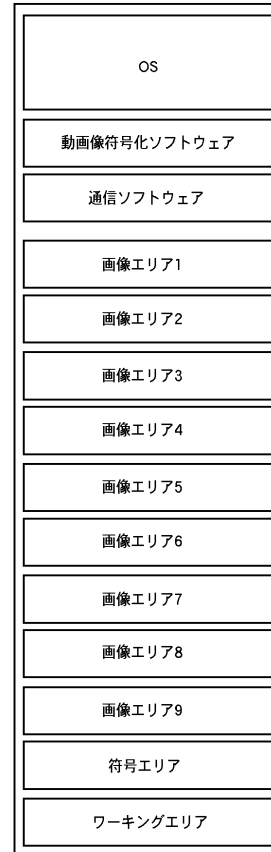
【 図 4 】



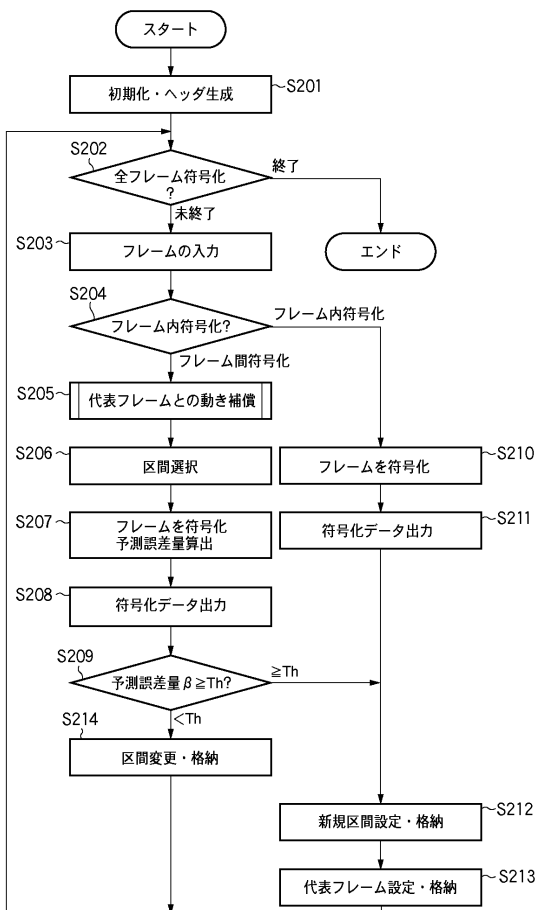
【図 5】



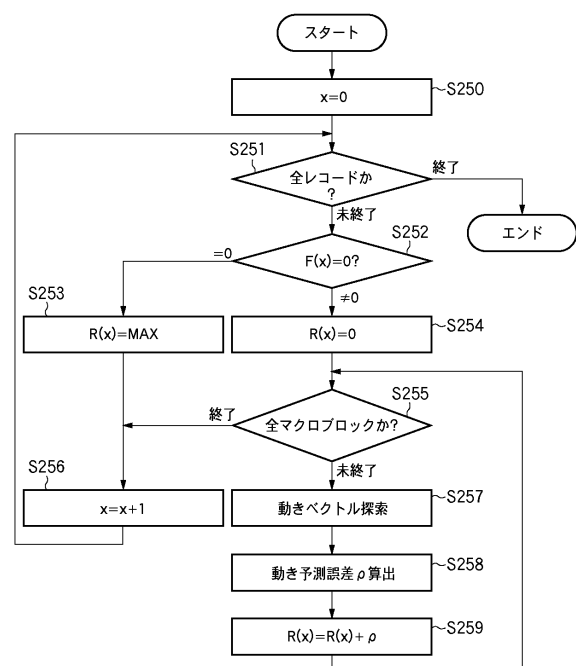
【図 6】



【図 7】



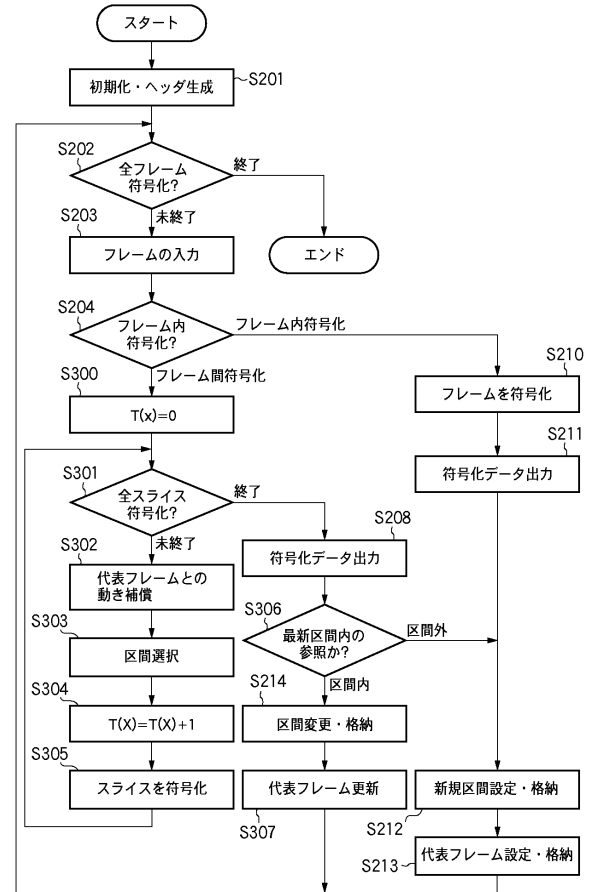
【図 8】



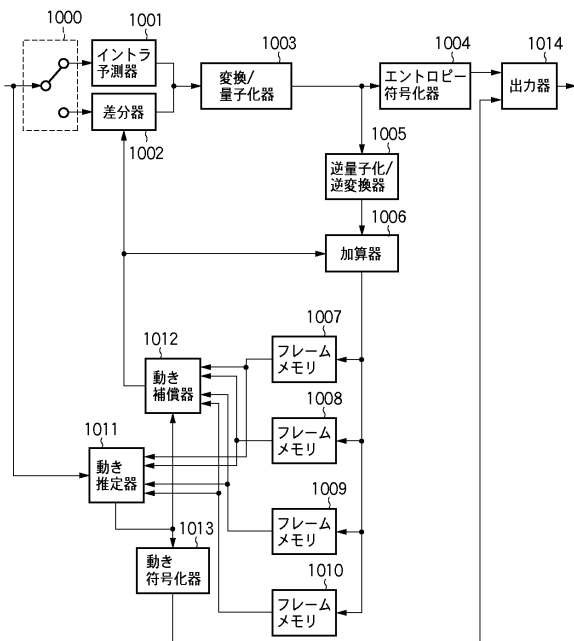
【図 9】

レコードX	有効フラグF	区間始点値S	区間終点値E	代表フレーム番号D
0	1	3	7	4
1	1	8	9	8
2	0	0	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0

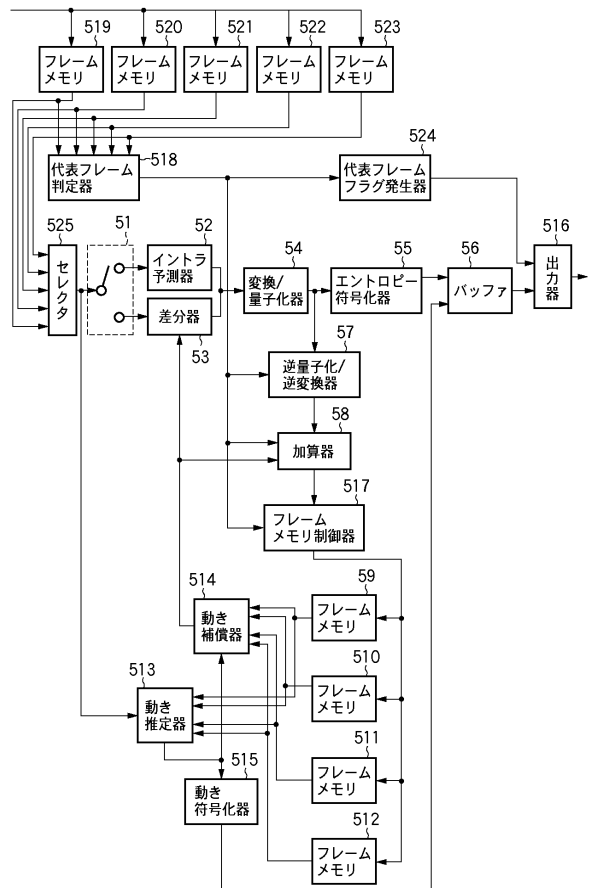
【図 10】



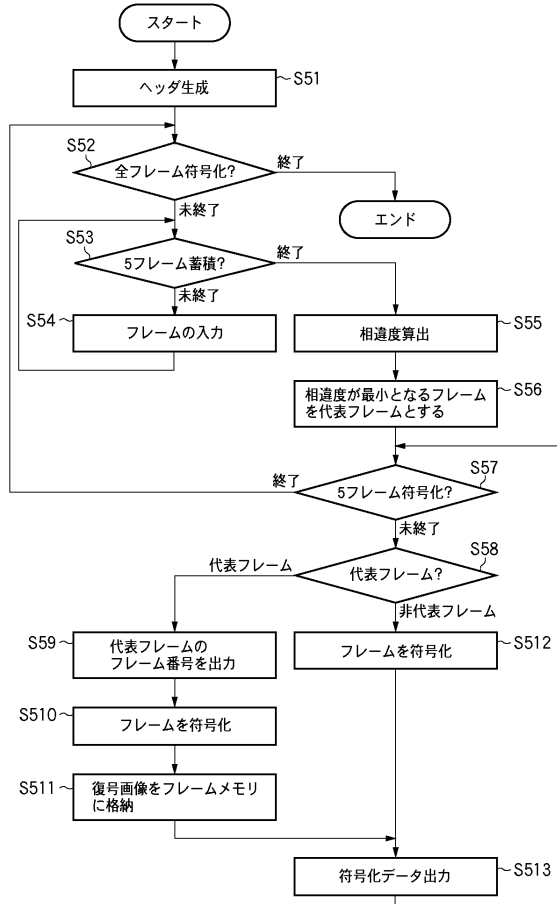
【図 11】



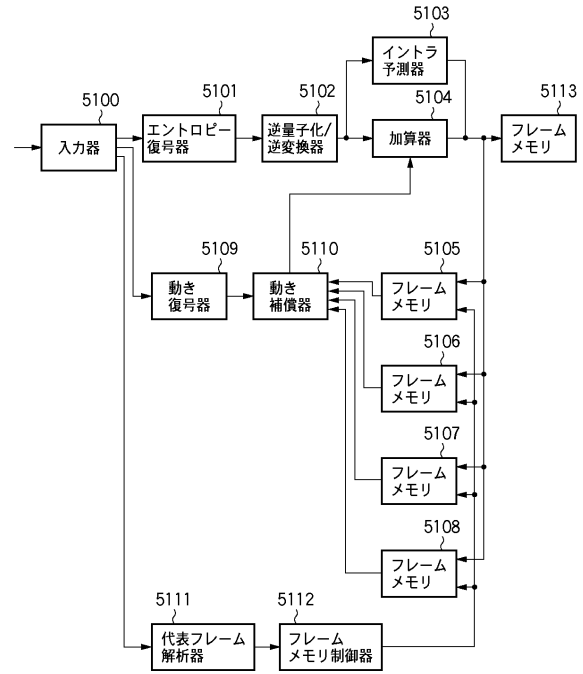
【図 12】



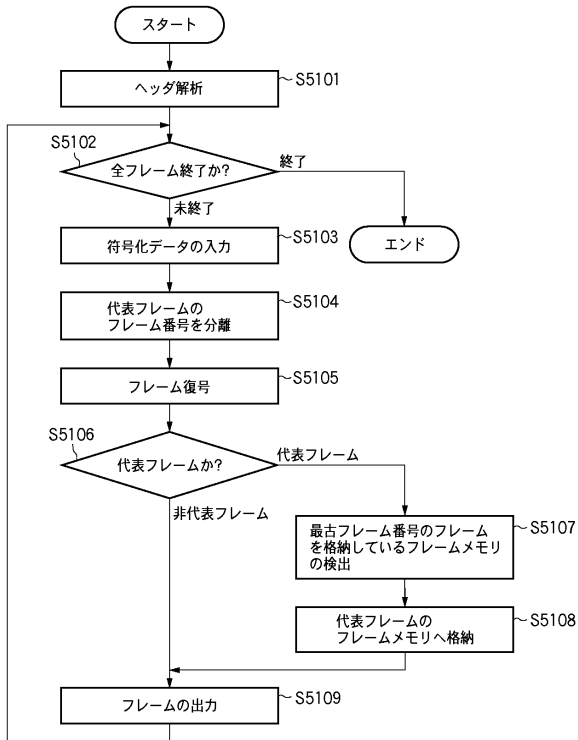
【図 13】



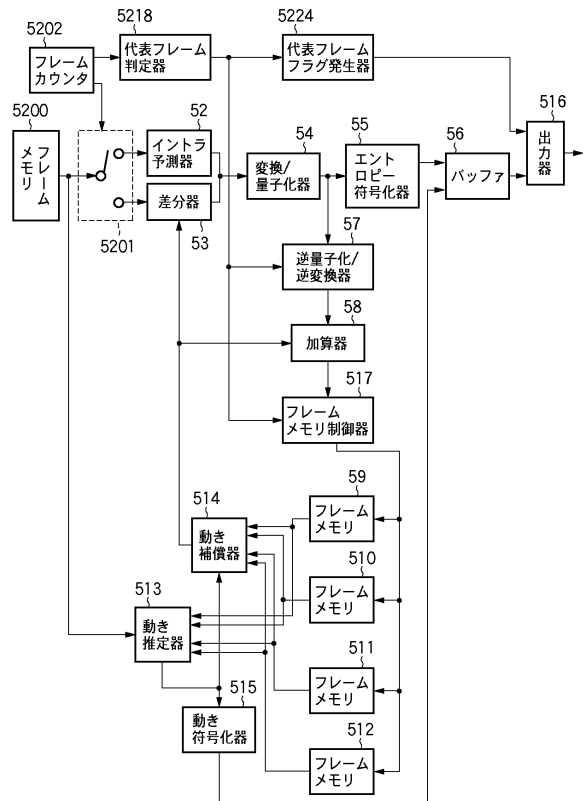
【図 14】



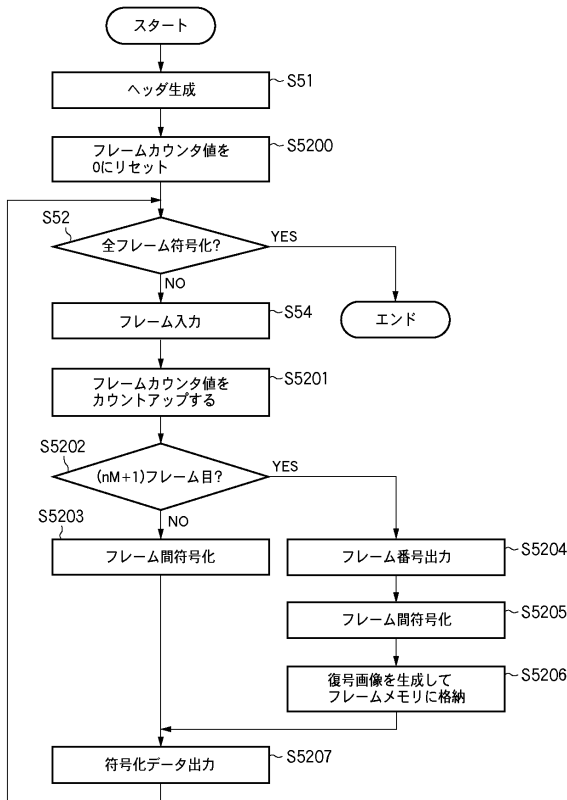
【図 15】



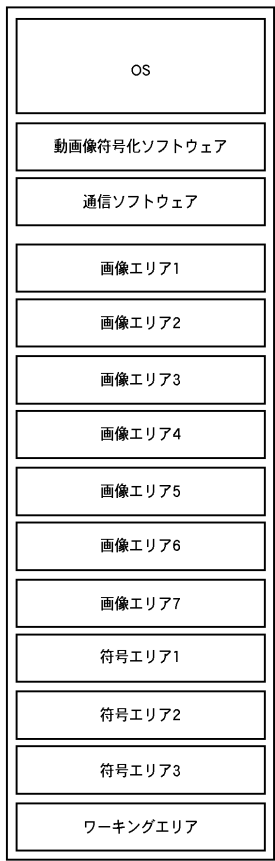
【図 16】



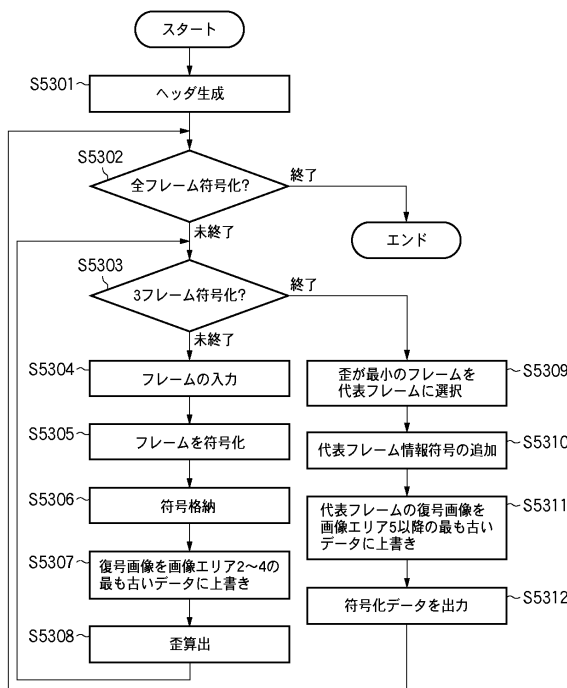
【図 17】



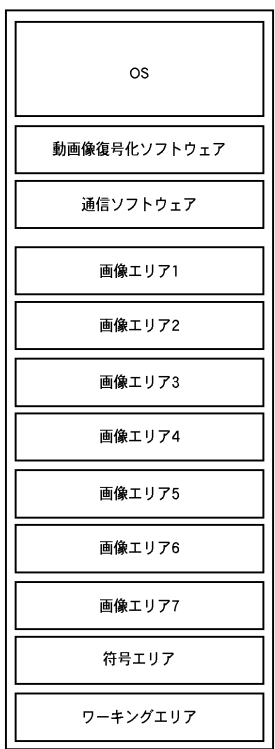
【図 18】



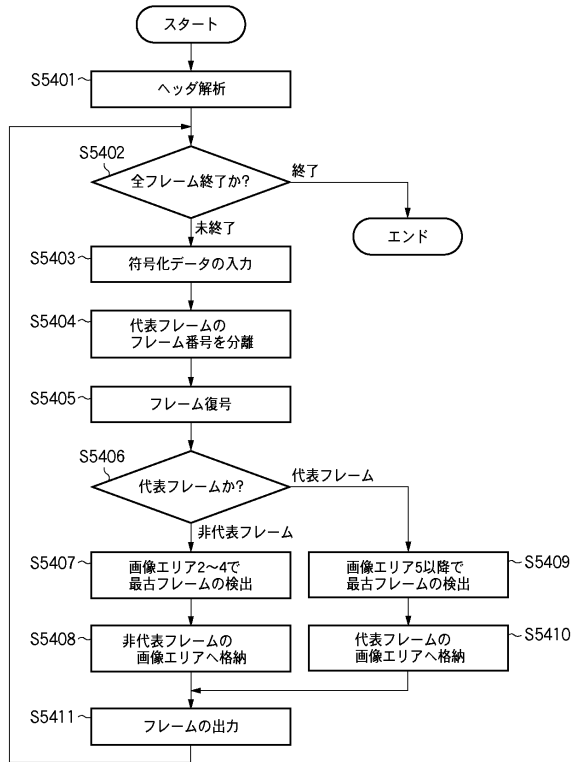
【図 19】



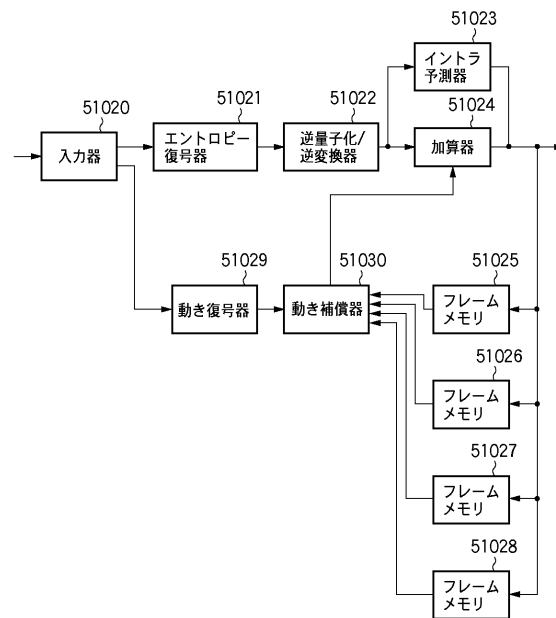
【図 20】



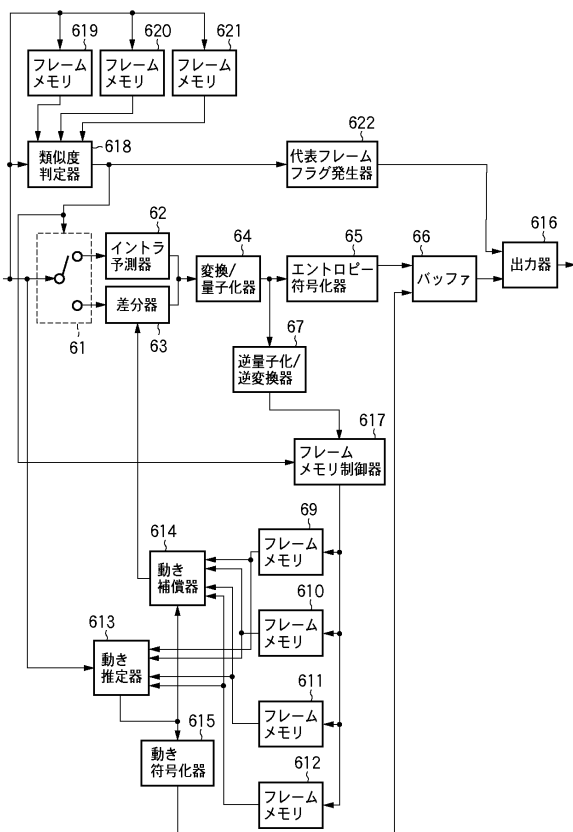
【図 2 1】



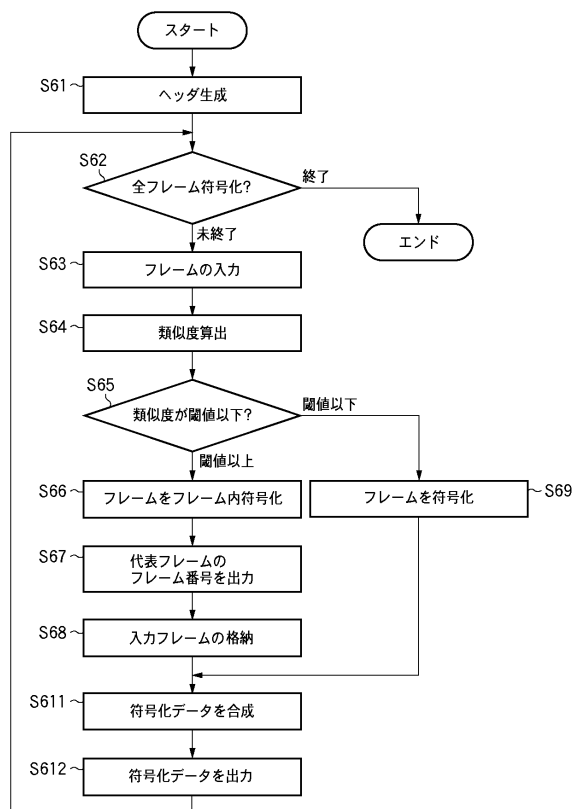
【図 2 2】



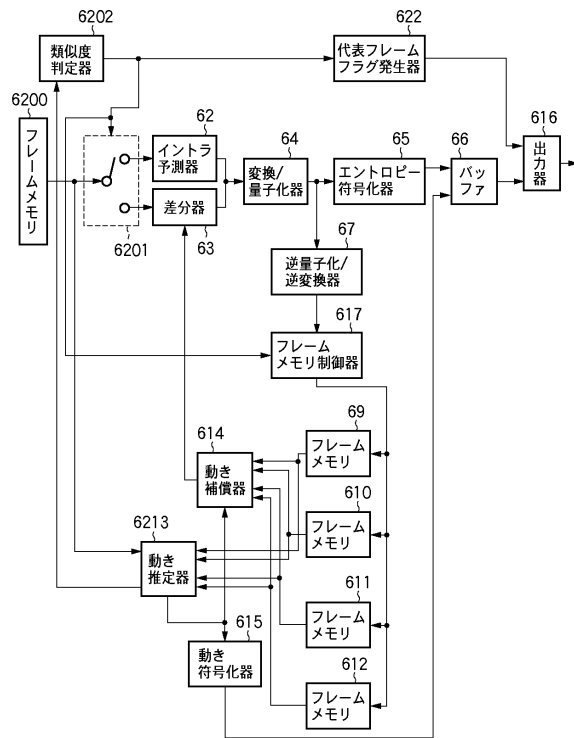
【図 2 3】



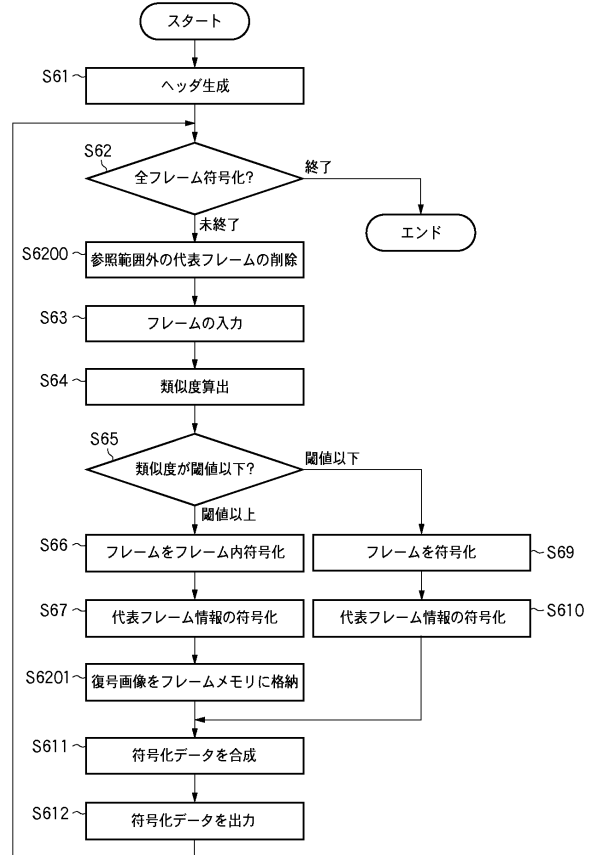
【図 2 4】



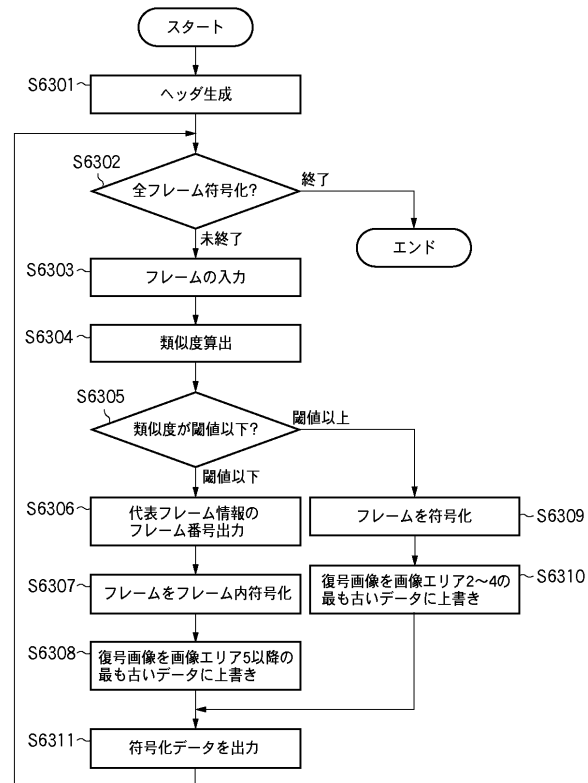
【図 25】



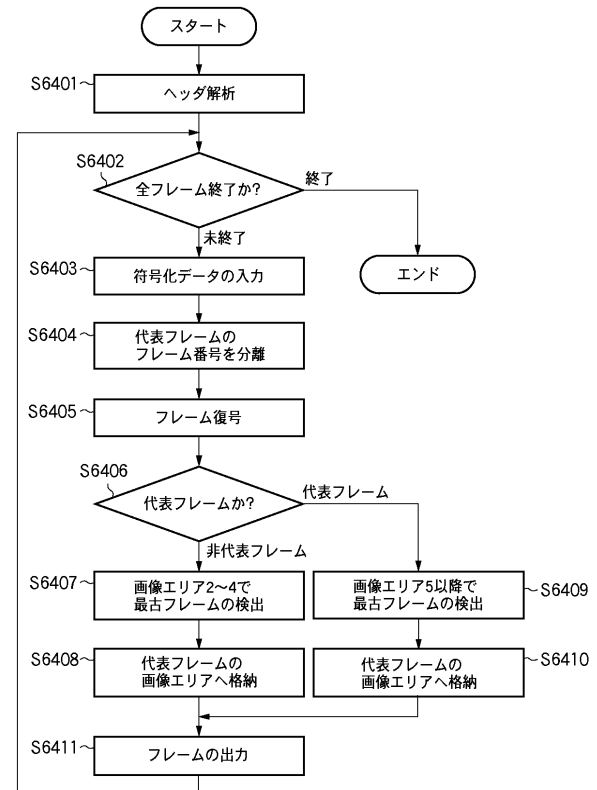
【図 26】



【図 27】



【図 28】



フロントページの続き

審査官 古市 徹

(56)参考文献 特開 2 0 0 3 - 1 2 5 4 1 0 (J P , A)

Xu-Dong Zhang, ET AL. , Dynamic selection and effective compression of key frames for v
ideo abstraction , Pattern Recognition Letters , NL , 2 0 0 3 年 6 月 , vol.24, no.9-10 ,
p.1533-1542

(58)調査した分野(Int.Cl. , D B 名)

H 0 4 N 7 / 2 4 - 7 / 6 8