世界知的所有権機関
国際事務局

PCT

特許協力条約に基づいて公開された国際出願

(51) 国際特許分類6 **H04N 7/32**	**A1**	(11) 国際公開番号	**WO98/56185**
		(43) 国際公開日	1998年12月10日(10.12.98)

(21) 国際出願番号 PCT/JP98/02435

(22) 国際出願日 1998年6月2日(02.06.98)

(30) 優先権データ
特願平9/144916 1997年6月3日(03.06.97) JP

(71) 出願人（米国を除くすべての指定国について）
株式会社 日立製作所(HITACHI, LTD.)[JP/JP]
〒101-8010 東京都千代田区神田駿河台四丁目6番地
Tokyo, (JP)
(72) 発明者；および
(75) 発明者／出願人（米国についてのみ）
中屋雄一郎(NAKAYA, Yuichiro)[JP/JP]
〒185-8601 東京都国分寺市東恋ケ窪一丁目280番地
株式会社 日立製作所 中央研究所内 Tokyo, (JP)
(74) 代理人
弁理士 高橋明夫(TAKAHASHI, Akio)
〒103-0025 東京都中央区日本橋茅場町二丁目9番8号
友泉茅場町ビル 日東国際特許事務所 Tokyo, (JP)

(81) 指定国 CN, JP, KR, SG, US, 欧州特許 (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

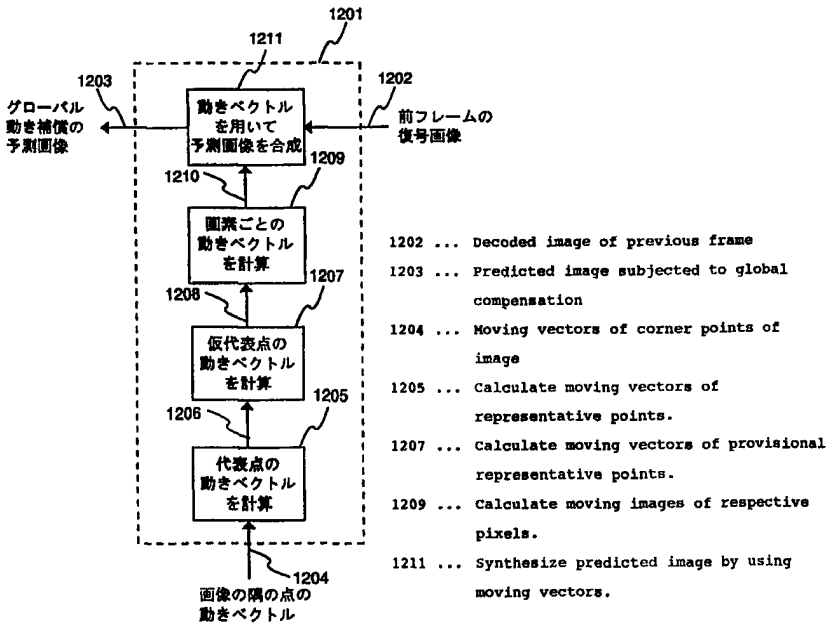
添付公開書類
国際調査報告書

(54)Title: IMAGE ENCODING AND DECODING METHOD AND DEVICE

(54)発明の名称 画像符号化及び復号化方法及び装置

(57) Abstract

A method for simplifying the arithmetic operation of the global motion compensation and warping prediction in encoding and decoding the image signal of motion compensation. In a process for synthesizing a predicted image (1203) subjected to global motion compensation and formed to predict the present frame image from a previous frame image (1202) by using moving vectors (1205) of representative points which are spaced at spacial intervals having specific features in an image frame, the moving vectors of provisional representative points are determined from the moving vectors (1206) of the representative points by first interpolation/exterpolation processes (1207), and then the moving vectors (1210) of the pixels are determined from the moving vectors (1208) of the provisional representative points by second interpolation/exterpolation processes (1209). Since the division for synthesizing the predicted image is substituted by a shift calculation with a small number of bits shifted, the processing with a computer or an exclusive hardware can be simplified.



(57)要約

動き補償の画像信号の符号化及び複合におけるグローバル動き補償及びワーピング予測の処理の演算処理を簡略化するため方法を提供する。画像フレームの空間的な間隔が特別な特徴をもつ複数の代表点の動きベクトル１２０５を用いて、前フレーム画像１２０２から現フレーム画像を予測したグローバル動き補償した予測画像１２０３を合成する処理において、代表点の動きベクトル１２０６から仮代表点の動きベクトルを求める第１の内・外挿処理１２０７を行い、その後、仮代表点の動きベクトル１２０８から画素ごとの動きベクトル１２１０を求める第２の内・外挿処理１２０９を行う。

予測画像を合成する際の除算がシフトされるビット数が少ないシフト演算によって代用できるため、計算機又は専用ハードウェアによる処理を簡略化することができる。

PCTに基づいて公開される国際出願のパンフレット第一頁に掲載されたPCT加盟国を同定するために使用されるコード(参考情報)

AL アルバニア
AM アルメニア
AT オーストリア
AU オーストラリア
AZ アゼルバイジャン
BA ボズニア・ヘルツェゴビナ
BB バルバドス
BE ベルギー
BF ブルギナ・ファソ
BG ブルガリア
BJ ベナン
BR ブラジル
BY ベラルーシ
CA カナダ
CF 中央アフリカ
CG コンゴー
CH スイス
CI コートジボアール
CM カメルーン
CN 中国
CU キューバ
CY キプロス
CZ チェッコ
DE ドイツ
DK デンマーク
EE エストニア
ES スペイン
FI フィンランド
FR フランス
GA ガボン
GB 英国
GD グレナダ
GE グルジア
GH ガーナ
GM ガンビア
GN ギニア
GW ギニア・ビサオ
GR ギリシャ
HR クロアチア
HU ハンガリー
ID インドネシア
IE アイルランド
IL イスラエル
IN インド
IS アイスランド
IT イタリア
JP 日本
KE ケニア
KG キルギスタン
KP 北朝鮮
KR 韓国
KZ カザフスタン
LC セントルシア
LI リヒテンシュタイン
LK スリ・ランカ
LR リベリア
LS レソト
LT リトアニア
LU ルクセンブルグ
LV ラトヴィア
MC モナコ
MD モルドヴァ
MG マダガスカル
MK マケドニア旧ユーゴスラヴィア共和国
ML マリ
MN モンゴル
MR モーリタニア
MW マラウイ
MX メキシコ
NE ニジェール
NL オランダ
NO ノールウェー
NZ ニュー・ジーランド
PL ポーランド
PT ポルトガル
RO ルーマニア
RU ロシア
SD スーダン
SE スウェーデン
SG シンガポール
SI スロヴェニア
SK スロヴァキア
SL シエラ・レオネ
SN セネガル
SZ スワジランド
TD チャード
TG トーゴー
TJ タジキスタン
TM トルクメニスタン
TR トルコ
TT トリニダッド・トバゴ
UA ウクライナ
UG ウガンダ
US 米国
UZ ウズベキスタン
VN ヴィェトナム
YU ユーゴースラビア
ZW ジンバブエ

明細書

画像符号化及び復号化方法及び装置

技術分野

本発明は、画像符号化及び復号化方法及び装置、さらに詳しく言えば、動画像を含む画像信号の符号化、復号化の処理において、代表点の動きベクトルに対して内・外挿処理を行うことにより画像内の画素の動きベクトルを計算するフレーム間予測画像の合成処理を含む画像符号化及び復号化方法及び装置に関するものである。

発明の背景

動画像の高能率符号化において、異なる時間に発生したフレーム間の類似性を活用するフレーム間予測（動き補償）は情報圧縮に大きな効果を示すことが知られている。現在の画像符号化技術の主流となっている動き補償方式は、動画像符号化方式の国際標準であるＨ．２６１、ＭＰＥＧ１、ＭＰＥＧ２に採用されているブロックマッチング方式である。この方式では、符号化しようとする画像を多数のブロックに分割し、ブロックごとにその動きベクトルを求める。

ブロックマッチングは現在最も広く利用されている動き補償方式であるが、画像全体が拡大・縮小・回転している場合には、全てのブロックに対して動きベクトルを伝送しなければならず、符号化効率が悪くなる問題がある。この問題に対し、画像全体の動きベクトル場を少ないパラメータを用いて表現するグローバル動き補償（例えば、M.Hotter, "Differential estimation of the global motion parameters zoom and pan", Signal Processing, vol. 16, no. 3, pp. 249-265, Mar. 1989）が提案されている。これは、画像内の画素（ｘ，ｙ）の動きベクトル（ｕｇ（ｘ，ｙ），ｖｇ（ｘ，ｙ））を、

$$u_g(x, y) = a_0x + a_1y + a_2$$
$$v_g(x, y) = a_3x + a_4y + a_5 \quad \cdots (1)$$

や、

$$u_g(x, y) = b_0xy + b_1x + b_2y + b_3$$
$$v_g(x, y) = b_4xy + b_5x + b_6y + b_7 \quad \cdots (2)$$

の形式で表し、この動きベクトルを利用して動き補償を行う方式である。ここでａ０～ａ５、ｂ０～ｂ７は動きパラメータである。動き補償を行う際には、送信側と受信側で同じ予測画像が得られなければならない。このために、送信側は受信側へａ０～ａ５又はｂ０～ｂ７の値を直接伝送しても良いが、代わりにいくつかの代表点の動きベクトルを伝送する方法もある。いま、画像の左上端、右上端、左下端、右下端の画素の座標がそれぞれ（０，０）、（ｒ，０）、（０，ｓ）、（ｒ，ｓ）で表されるとする（ただし、ｒとｓは正の整数）。このとき、代表点（０，０）、（ｒ，０）、（０，ｓ）の動きベクトルの水平・垂直成分をそれぞれ（ｕａ，ｖａ）、（ｕｂ，ｖｂ）、（ｕｃ，ｖｃ）とすると、式(１）は

$$u_g(x, y) = \frac{u_b - u_a}{r}x + \frac{u_c - u_a}{s}y + u_a$$
$$v_g(x, y) = \frac{v_b - v_a}{r}x + \frac{v_c - v_a}{s}y + v_a \quad \cdots (3)$$

となる。このことはａ０～ａ５を伝送する代わりにｕａ、ｖａ、ｕｂ、ｖｂ、ｕｃ、ｖｃを伝送しても同様の機能が実現できることを意味する。これと同じように、４個の代表点（０，０）、（ｒ，０）、（０，ｓ）、（ｒ，ｓ）の動きベクトルの水平・垂直成分（ｕａ，ｖａ）、（ｕｂ，ｖｂ）、（ｕｃ，ｖｃ）、（ｕｄ，ｖｄ）を用いて式(２）は、

$$u_g(x, y) = \frac{s-y}{s}\left(\frac{r-x}{r}u_a + \frac{x}{r}u_b\right) + \frac{y}{s}\left(\frac{r-x}{r}u_c + \frac{x}{r}u_d\right)$$
$$= \frac{u_a-u_b-u_c+u_d}{rs}xy + \frac{-u_a+u_b}{r}x + \frac{-u_a+u_c}{s}y + u_a \quad \cdots (4)$$
$$v_g(x, y) = \frac{v_a-v_b-v_c+v_d}{rs}xy + \frac{-v_a+v_b}{r}x + \frac{-v_a+v_c}{s}y + v_a$$

と書き換えることができる。　従って、ｂ０～ｂ７を伝送する代わりにｕａ、ｖａ、ｕｂ、ｖｂ、ｕｃ、ｖｃ、ｕｄ、ｖｄを伝送しても同様の機能が実現できる。この様子を第１図に示す。現フレームの原画像１０２と参照画像１０１の間でグローバル動き補償が行われたとして、動きパラメータの代わりに代表点１０３、１０４、１０５、１０６の動きベクトル１０７、１０８、１０９、１１０（このとき、動きベクトルは現フレームの原画像の点を出発点として、参照画像内の対応する点を終点とするものとして定義する）を伝送しても良い。本明細書では式(１）を用いる方式を線形内・外挿に基づくグローバル動き補償、式(２）を用いる方式を共１次内・外挿に基づくグローバル動き補償とよぶ。

このグローバル動き補償の処理を、画像内のより小さい領域に適用するのがワーピング予測である。第２図に共一次内・外挿を用いるワーピング予測の例を示す。この図は、参照画像２０１を用いて現フレームの原画像２０２の予測画像を合成する処理を示したものである。まず現フレームは複数の多角形のパッチに分割され、パッチ分割された画像２０９となる。パッチの頂点は格子点とよばれ、各格子点は複数のパッチに共有される。例えば、パッチ２１０は、格子点２１１、２１２、２１３、２１４から構成され、これらの格子点は他のパッチの頂点を兼ねている。このように画像を複数のパッチに分割した後に、動き推定が行なわれる。ここに示す例では、動き推定は各格子点を対象として参照画像との間で行なわれる。この結果、動き推定後の参照画像２０３で各パッチは変形されたものとなる。例えば、パッチ２１０は、変形されたパッチ２０４に対応している。これは、動き推定の結果、格子点２０５、２０６、２０７、２０８がそれぞれ２１１、２１２、２１３、２１４に対応していると推定されたためである。このようにして格子点の動きベクトルを求め、共１次内挿によってパッチ内の各画素の動きベクトルを計算するこ

とにより、フレーム間予測画像が合成される。このワーピング予測の処理は基本的に第１図に示したグローバル動き補償と同じ処理であり、「画像の隅の点の動きベクトル」が「格子点の動きベクトル」に変えられているだけである。また、長方形の代わりに３角形のパッチを使用すれば、線形内・外挿によるワーピング予測も実現することができる。

なお、画像全体の動きベクトル場を少ないパラメータを用いて表現するグローバル動き補償の処理を簡易にした符号化及び復号方法に関して本願出願人による発明「画像符号化方法および復号化方法」（特願平８-６０５７２号）及び「フレーム間予測画像の合成方法」（特願平８-２４９６０１号）がある。

上述のグローバル動き補償やワーピング予測を導入することにより、画像の動きを少ないパラメータを用いて正確に表現することが可能となり、より高い情報圧縮率が実現できる。しかし、その一方で符号化及び復号化における処理量は従来の方式と比較して増加する。特に式(３）及び式(４)の除算は、処理を複雑にする大きな要因となってしまう。すなわち、グローバル動き補償やワーピング予測では、予測画像の合成のための処理量が多くなる問題が発生する。

発明の開示

発明の目的は、これらの動き補償方式における除算の処理をビット数の少ないレジスタを用いた２進数のシフト演算に置き換えることにより、演算量を減少させることにある。

上記目的を達成するため、本発明は、グローバル動き補償やワーピング予測によってフレーム間予測画像の合成処理を行う画像符号化及び復号化において、空間的な間隔が特徴を持つ複数の代表点の動きベクトルに対し、２段階の内・外挿入処理を行うことにより動きベクトルを求め

る。さらに詳しく言えば、フレーム間予測画像の合成処理において、

画素のサンプリング間隔を水平、垂直方向共に１として、サンプリング点が座標の水平、垂直成分が、共に整数にｗを加えた数である点の上に存在している画像を対象として（ただし、ｗ＝ｗｎ／ｗｄ、かつｗｎは負ではない整数、かつｗｄは２のｈｗ乗、かつｈｗは負ではない整数、かつｗｎ＜ｗｄ）、４個の代表点における動きベクトルに対し、共１次内・外挿を行うことによって画素の動きベクトルを計算する場合に、座標（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）に（ｉ、ｊ、ｐ、ｑは整数）に代表点が存在し、かつ代表点の動きベクトルの水平・垂直成分が１／ｋの整数倍の値をとり（ただし、ｋは２のｈｋ乗、かつｈｋは負ではない整数）、かつ座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルを求めるときに、座標（ｉ，ｊ）と（ｉ，ｊ＋ｑ）［又は（ｉ＋ｐ，ｊ）］に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標（ｉ，ｙ＋ｗ）［又は（ｘ＋ｗ，ｊ）］に位置する点の動きベクトルの水平・垂直成分をそれぞれ１／ｚの整数倍をとる数値として（ただし、ｚは２のｈｚ乗、かつｈｚは負ではない整数）求め、さらに座標（ｉ＋ｐ，ｊ）［又は（ｉ，ｊ＋ｑ）］と（ｉ＋ｐ，ｊ＋ｑ）に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標（ｉ＋ｐ，ｙ＋ｗ）［又は（ｘ＋ｗ，ｊ＋ｑ）］に位置する点の動きベクトルの水平・垂直成分をそれぞれ１／ｚの整数倍をとる数値として求めた後に、（ｉ，ｙ＋ｗ）［又は（ｘ＋ｗ，ｊ）］と（ｉ＋ｐ，ｙ＋ｗ）［又は（ｘ＋ｗ，ｊ＋ｐ）］に位置する上記２個の動きベクトルに対して線形内・外挿を行うことにより、座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルの水平・垂直成分をそれぞれ１／ｍの整数倍をとる数値として（ただし、ｍは２のｈｍ乗、かつｈｍは負ではない整数）求める。

本発明は、代表点の座標を巧みに選択することによって除算処理をシフト演算で実現できるようにし、さらにシフト演算においてシフトされるビット数を少なくすることにより、ビット数の少ないレジスタによって上記動き補償方式の演算が実現できる。

図面の簡単な説明

第１図は、代表点の動きベクトルを伝送するグローバル動き補償の例を示した図である。

第２図は、ワーピング予測の処理例を示した図である。

第３図は、高速な処理を行うための代表点の配置の例を示した図である。

第４図は、本発明のソフトウェア画像符号化装置の構成例を示した図である。

第５図は、本発明のソフトウェア画像復号化装置の構成例を示した図である。

第６図は、本発明による画像符号化装置の構成例を示した図である。

第７図は、本発明による画像復号化装置の構成例を示した図である。

第８図は、第６図の動き補償処理部616の構成例を示した図である。

第９図は、第６図の動き補償処理部616の他の構成例を示した図である。

第１０図は、第７図の予測画像合成部７１１の構成例を示した図である。

第１１図は、第９図の予測画像合成部１１０３の構成例を示した図である。

第１２図は、グローバル動き補償予測画像合成部の構成例を示した図である。

第１３図は、ソフトウェア画像符号化装置における処理のフローチャートの例を示した図である。

第１４図は、ソフトウェア画像符号化装置における動き補償処理のフローチャートの例を示した図である。

第１５図は、ソフトウェア画像復号化装置における処理のフローチャートの例を示した図である。

第１６図は、ソフトウェア画像復号化装置における予測画像合成処理のフローチャートの例を示した図である

第１７図は、２段階の処理によりグローバル動き補償予測画像を合成する画像符号化・復号化を使用する装置の具体例を示した図である。

発明を実施するための最良の形態

本発明の理解を容易にするため、グローバル動き補償及びワーピング予測における演算の高速化方法に関する本出願人が先に出願した発明（特願平 08-060572 及び特願平 08-249601）を説明する。また、以下では本発明をグローバル動き補償に適用した場合に関して説明するが、本発明はグローバル動き補償と同様の処理を行うワーピング予測にも応用することが可能である。

以下の説明では、画素のサンプリング間隔が水平・垂直方向共に１であるとして、座標の水平・垂直成分が共に整数にｗを加えた値である点（ただし、ｗ＝ｗｎ／ｗｄ、かつｗｎは負ではない整数、かつｗｄは正の整数、かつｗ ｎ＜ｗ ｄ）に画素が存在しているとする。ｗはグローバル動き補償における代表点の座標と画素の座標の位相のずれを表しており、典型的な値としては 0、1/2、1/4 などが挙げられる。また、画像の

水平方向と垂直方向の画素数はそれぞれｒとｓであり（ただし、ｒとｓは正の整数）、かつ画像の画素は水平座標が０以上ｒ未満、垂直座標が０以上ｓ未満の範囲に存在しているとする。

線形内・外挿（アフィン変換）又は共１次内・外挿（共１次変換）を用いた動き補償を行う際には、画素ごとの動きベクトルに対して量子化を行うと、ミスマッチの防止や演算の簡略化などの効果を得ることができる（特願平 06-193970）。以下では、画素の動きベクトルの水平成分と垂直成分が１／ｍ（ｍは正の整数）の整数倍であるとする。また、「従来の技術」で説明した代表点の動きベクトルを用いるグローバル動き補償を行うと仮定し、各代表点の動きベクトルは１／ｋ（ｋは正の整数）の整数倍であるとする。なお、本明細書では、「画素の動きベクトル」はグローバル動き補償を行う際に、実際に予測画像を合成するために用いる動きベクトルのことを指す。一方、「代表点の動きベクトル」は画素の動きベクトルを計算するために用いるパラメータを意味している。従って、量子化ステップサイズの違いなどが原因で、同じ座標上に存在していても画素の動きベクトルと代表点の動きベクトルが一致しない場合も起こり得る。

まず、共１次内・外挿を用いるグローバル動き補償について第３図を用いて説明する。この例では、第１図に示したように、代表点を画像３０１の隅に位置する点とはせず、（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）に位置する点３０２、３０３、３０４、３０５として一般化している（ｉ、ｊ、ｐ、ｑは整数）。このとき、点３０２、３０３、３０４、３０５は画像の内部に存在していても外部に存在していても良い。代表点の動きベクトルの水平・垂直成分をｋ倍したものをそれぞれ（ｕ０，ｖ０）、（ｕ１，ｖ１）、（ｕ２，ｖ２）、（ｕ３，ｖ３）とすると（ｕ０、ｖ０、ｕ１、ｖ１、ｕ２、ｖ２、ｕ３、ｖ３は

整数）、（ｘ＋ｗ、ｙ＋ｗ）に位置する画素の動きベクトルの水平・垂直成分をｍ倍したもの（ｕ（ｘ＋ｗ，ｙ＋ｗ），ｖ（ｘ＋ｗ，ｙ＋ｗ））は、ｗ＝０のときは以下の式で表すことができる（ただし、ｘ、ｙ、ｕ（ｘ，ｙ）、ｖ（ｘ，ｙ）は整数）。

$$
\begin{aligned}
u(x+w, y+w) &= u(x, y) \\
&= (((j+q-y)((i+p-x)u_0 + (x-i)u_1) \\
&\quad + (y-j)((i+p-x)u_2 + (x-i)u_3))m) // (pqk) \\
v(x+w, y+w) &= v(x, y) \\
&= (((j+q-y)((i+p-x)v_0 + (x-i)v_1) \\
&\quad + (y-j)((i+p-x)v_2 + (x-i)v_3))m) // (pqk)
\end{aligned}
\qquad \cdots (5)
$$

ただし、「//」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等である。演算誤差を小さくするためには、非整数値は最も近い整数に丸め込まれることが望ましい。このとき整数に１／２を加えた値の丸め込み方法は、

(１) ０に近づける方向に丸め込む、

(2) ０から遠ざける方向に丸め込む、

(3) 被除数が負の場合は０に近づける方向、正の場合は０から遠ざける方向に丸め込む（除数は常に正であるとする）、

(4) 被除数が負の場合は０から遠ざける方向、正の場合は０に近づける方向に丸め込む（除数は常に正であるとする）、

などが考えられる。これらの中で(3)と(4)は、被除数の正負に関わらず丸め込みの方向が変化しないため、正負判定が必要ない分だけ処理量の点で有利である。(3)を用いた高速処理は、例えば以下の式によって実現することができる。

$$
\begin{aligned}
u(x+w, y+w) &= u(x, y) \\
&= (Lpqk + ((j+q-y)((i+p-x)u_0 + (x-i)u_1) \\
&\quad + (y-j)((i+p-x)u_2 + (x-i)u_3))m + ((pqk)\#2)) \\
&\quad \#(pqk) - L \\
v(x+w, y+w) &= v(x, y) \\
&= (Mpqk + ((j+q-y)((i+p-x)v_0 + (x-i)v_1) \\
&\quad + (y-j)((i+p-x)v_2 + (x-i)v_3))m + ((pqk)\#2)) \\
&\quad \#(pqk) - M
\end{aligned}
\qquad \cdots (6)
$$

ただし、「#」は小数点以下を０の方向に切り捨てる整数の除算であり、演算の優先順位は乗除算と同じであるとする。これは、一般に計算機では最も実現しやすい形式の除算である。ここで、ＬとＭは除算の被除数を常に正に保つための数で、十分に大きな正の整数である。また、（ｐｑｋ＃２）の項は、除算結果を最も近い整数に丸め込むために用いられる。

処理を整数化することはそれ自体処理量の低減に貢献するが、ここでｐ、ｑ、ｋをそれぞれ２のα、β、ｈｋ乗（α、β、ｈｋは負ではない整数）とすると、数５の除算は$\alpha+\beta+$ｈｋビットのシフト演算で実現できるため、計算機や専用ハードウェアにおける処理量を大きく減らすことができる。さらにｍを２のｈｍ乗とすれば（ｈｍは負ではない整数、ｈｍ＜$\alpha+\beta+$ｈｋ）、式（６）は、

$$
\begin{aligned}
u(x+w, y+w) &= u(x, y) \\
&= \left((2L+1)<<\left(\alpha+\beta+h_k-h_m-1\right)\right. \\
&\quad + (j+q-y)((i+p-x)u_0 + (x-i)u_1) \\
&\quad + (y-j)((i+p-x)u_2 + (x-i)u_3)\Big) \\
&\quad >> \left(\alpha+\beta+h_k-h_m\right) - L \\
v(x+w, y+w) &= v(x, y)
\end{aligned}
\qquad \cdots (7)
$$

$$= \left((2M+1)<<\left(\alpha+\beta+h_k-h_m-1\right)\right.$$
$$+ (j+q-y)\left((i+p-x)\,v_0 + (x-i)\,v_1\right)$$
$$+ (y-j)\left((i+p-x)\,v_2 + (x-i)\,v_3\right)\Big)$$
$$>>\left(\alpha+\beta+h_k-h_m\right) - M$$

と書き換えることができ（「x<<α」はxをαビット左にシフトして下位αビットに0を入れる、「x>>α」はxをαビット右にシフトして上位αビットに0を入れることを意味し、これらの演算子の優先順位は加減算と乗除算の中間であるとする）、シフトされるビット数をα＋β＋hk－hmとすることができる。

wが0ではないときには、w＝wn／wdの定義にしたがい、式（5）は以下のように書き換えることができる。

$$u(x+w, y+w) = u\left(x+\frac{w_d}{w_n}, y+\frac{w_d}{w_n}\right)$$
$$= \left(\left(\left(w_d j+w_d q-w_d y-w_n\right)\left(\left(w_d i+w_d p-w_d x-w_n\right) u_0\right.\right.\right.$$
$$\left.\left.\left. + \left(w_d x+w_n-w_d i\right) u_1\right)\right.\right.$$
$$\left. + \left(w_d y+w_n-w_d j\right)\left(\left(w_d i+w_d p-w_d x-w_n\right) u_2\right.\right.$$
$$\left.\left. + \left(w_d x+w_n-w_d i\right) u_3\right)\right) m\Big)$$
$$/\!/\left(w_d{}^2 pqk\right)$$

$$v(x+w, y+w) = v\left(x+\frac{w_d}{w_n}, y+\frac{w_d}{w_n}\right) \qquad \cdots (8)$$
$$= \left(\left(\left(w_d j+w_d q-w_d y-w_n\right)\left(\left(w_d i+w_d p-w_d x-w_n\right) v_0\right.\right.\right.$$
$$\left.\left.\left. + \left(w_d x+w_n-w_d i\right) v_1\right)\right.\right.$$
$$\left. + \left(w_d y+w_n-w_d j\right)\left(\left(w_d i+w_d p-w_d x-w_n\right) v_2\right.\right.$$
$$\left.\left. + \left(w_d x+w_n-w_d i\right) v_3\right)\right) m\Big)$$
$$/\!/\left(w_d{}^2 pqk\right)$$

このとき、wdが2のhw乗であり、かつhwは負ではない整数であるとすれば、（p・q・k・wd・wd）による除算はα＋β＋hk＋2hwビットのシフト演算となり、w＝0の場合と同様に除算をシフト演算に置換することが可能となる。また、式(7）の場合と同様に、hm＜α＋β＋hk＋2hwであれば、分母、分子の両方をmで割ることによってシフトされるビット数をα＋β＋hk＋2hw－hmビットに減

らすことが可能となる。このように、ｗｄが２のｈｗ乗であれば、ｗ＝０の場合の処理とｗ≠０の場合の処理は本質的に同じである。以下本明細書では、数式が多少複雑となるが、ｗ≠０の場合について検討を行う。ｗ＝０の場合の計算結果を求めるためには、ｗｎ＝０、ｗｄ＝１、ｈｗ＝０を代入すれば良い。

送信側と受信側で同じグローバル動き補償予測画像を得るためには、代表点の動きベクトルに関する情報を何らかの形で受信側に伝える必要がある。代表点の動きベクトルそのまま伝送する方法もあるが、画像の隅の点の動きベクトルを伝送し、この値から代表点の動きベクトルを計算する方法もある。この方法に関し、以下に説明する。

画像の隅の４個の点（－ｃ，－ｃ）、（ｒ－ｃ，－ｃ）、（－ｃ，ｓ－ｃ）、（ｒ－ｃ，ｓ－ｃ）の動きベクトルが１／ｎ整数倍の値のみとれるとして（ｎは正の整数、ｃ＝ｃｎ／ｃｄ、かつｃｎは負ではない整数、かつｃｄは正の整数、かつｃｎ＜ｃｄ）、これらの水平・垂直成分をｎ倍した（ｕ００，ｖ００）、（ｕ０１，ｖ０１）、（ｕ０２，ｖ０２）、（ｕ０３，ｖ０３）がグローバル動きパラメータとして伝送されるとする。ｃは画像の隅の点と代表点の間の位相のずれを表している。このｃの典型的な値としては0、1/2、1/4などが挙げられる。このとき、点（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）それぞれの動きベクトルの水平・垂直成分をｋ倍したものである（ｕ０，ｖ０）、（ｕ１，ｖ１）、（ｕ２，ｖ２）、（ｕ３，ｖ３）を、

$$
\begin{aligned}
u_0 &= u'(i, j) \\
v_0 &= v'(i, j) \\
u_1 &= u'(i+p, j) \\
v_1 &= v'(i+p, j) \\
u_2 &= u'(i, j+q) \\
v_2 &= v'(i, j+q) \\
u_3 &= u'(i+p, j+q) \\
v_3 &= v'(i+p, j+q)
\end{aligned}
\qquad \cdots (9)
$$

と定義する。ただし、u'（x，y）、v'（x，y）は、式(5)を変形して、

$$u'(x, y) = (((c_ds-c_n-c_dy)((c_dr-c_n-c_dx)u_{00} + (c_dx+c_n)u_{01}) + (c_dy+c_n)((c_dr-c_n-c_dx)u_{02} + (c_dx+c_n)u_{03}))k) /// (c_d^2rsn)$$

$$v'(x, y) = (((c_ds-c_n-c_dy)((c_dr-c_n-c_dx)v_{00} + (c_dx+c_n)v_{01}) + (c_dy+c_n)((c_dr-c_n-c_dx)v_{02} + (c_dx+c_n)v_{03}))k) /// (c_d^2rsn) \quad \cdots (10)$$

と定義する。このとき、「///」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等である。こうして（u 0，v 0）、（u 1，v 1）、（u 2，v 2）、（u 3，v 3）を計算し、（i，j）、（i＋p，j）、（i，j＋q）、（i＋p，j＋q）を代表点とするグローバル動き補償を行えば、（－c，－c）、（r－c，－c）、（－c，s－c）、（r－c，s－c）を代表点とするグローバル動き補償を近似することができる。このときに、上で述べたようにpとqを2の負ではない整数乗とすれば、処理を簡略化することが可能となる。一般的に、式(5)に示したような計算によって画像内の画素の動きベクトルを求めるときには、外挿の処理を行わないようにすることが望ましい。これは、外挿処理によって代表点の動きベクトルの量子化誤差を増幅しないようにするためである。以上の理由から、代表点は画像内の画素をすべて囲むような形に配置することが望ましい。従って、i＝j＝c＝0の場合などはpとqはrとsとほぼ同じか、やや大きめの値をとするのが適当である。しか

し、ｐとｑの値をあまり大きくし過ぎると演算に必要なビット数が増加してしまうので注意が必要である。

式(９）、式(１０）の処理において演算誤差を小さくするためには、「///」は非整数値を最も近い整数に丸め込むことが望ましい。このとき整数に１／２を加えた値の丸め込み方法としては、上で述べた(1)～(4)の方法が考えられる。ただし、式(５）（画素ごとに計算）の場合と比較して、式(１４）（１枚の画像で４回のみ計算）は演算が実行される回数が少ないため、式(1)又は(2)の方法を選んだとしても全体の演算量に大きな影響は与えない。

上述の例のように、ｐとｑの値が２の負ではない整数乗となるようにすれば、グローバル動き補償におけるフレーム間予測画像の合成処理は大幅に簡略化することができる。しかし、ここでもう１つの問題が発生する。例えば、画像符号化における典型的なパラメータとしてｐ＝５１２、ｑ＝５１２、ｋ＝３２、ｍ＝１６、ｗｄ＝２、ｗｎ＝１（ｗ＝0.5）である場合を考えると、$\alpha+\beta+h\ k+2\ h\ w-h\ m=21$となる。このことは、ｕ（ｘ＋ｗ，ｙ＋ｗ）が２進数で１２ビット以上を必要とする値である場合には、式(８の演算を高速に実行するために３３ビット以上のレジスタが必要になることを意味している。ｍ＝１６である場合などには、ｕ（ｘ＋ｗ，ｙ＋ｗ）は実際の動きベクトルの水平成分に１６を掛けた値となるため、これが２進数で１２ビット以上必要な値となるケースは十分にあり得る。しかし、その一方で３３ビット以上の整数を格納できるレジスタを持つプロセッサは現時点では少なく、かつ将来的にも高価となることが予想される。また、一般的にプロセッサの回路規模が大きくなれば、その分だけ消費電力も多くなるため、大きなレジスタを要求するアルゴリズムは消費電力の観点からも不利となる。従って、除算をシフト演算に置換できた場合でも、シフトされるビット数はでき

るだけ少ないことが望ましい。

この問題を解決するため本発明では、以下に説明する２段階の処理によるアルゴリズムをとる。（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）に位置する代表点の動きベクトルを用いて（ｘ＋ｗ、ｙ＋ｗ）に位置する画素の動きベクトルを計算する前に、まず（ｉ，ｙ＋ｗ）と（ｉ＋ｐ，ｙ＋ｗ）に存在する仮代表点の動きベクトルを、水平・垂直成分が１／ｚの整数倍（ｚは正の整数）となるように求める。上の例と同様に代表点（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）の動きベクトルの水平・垂直成分をｋ倍したものをそれぞれ（ｕ０，ｖ０）、（ｕ１，ｖ１）、（ｕ２，ｖ２）、（ｕ３，ｖ３）とする（ｕ０、ｖ０、ｕ１、ｖ１、ｕ２、ｖ２、ｕ３、ｖ３は整数）。このとき、（ｉ，ｙ＋ｗ）と（ｉ＋ｐ，ｙ＋ｗ）に仮代表点を配置し、これらの仮代表点の動きベクトルの水平・垂直成分をｚ倍したものである（ｕ Ｌ(ｙ＋ｗ)，ｖ Ｌ(ｙ＋ｗ)）と（ｕ Ｒ(ｙ＋ｗ)，ｖ Ｒ(ｙ＋ｗ)）を、以下のように定義する。

$$
\begin{aligned}
&u_L(y+w) \\
&\quad = (((w_d j + w_d q - w_d y - w_n)\, u_0 + (w_d y + w_n - w_d j)\, u_2)\, z) \,////\, (w_d q k) \\
&v_L(y+w) \\
&\quad = (((w_d j + w_d q - w_d y - w_n)\, v_0 + (w_d y + w_n - w_d j)\, v_2)\, z) \,////\, (w_d q k) \\
&u_R(y+w) \\
&\quad = (((w_d j + w_d q - w_d y - w_n)\, u_1 + (w_d y + w_n - w_d j)\, u_3)\, z) \,////\, (w_d q k) \\
&v_R(y+w) \\
&\quad = (((w_d j + w_d q - w_d y - w_n)\, v_1 + (w_d y + w_n - w_d j)\, v_3)\, z) \,////\, (w_d q k)
\end{aligned}
\qquad \cdots (11)
$$

このとき、「////」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等である（この「////」には、上で説明した「///」と同様の機能が要求

される）。（ｉ，ｙ＋ｗ）は（ｉ，ｊ）と（ｉ，ｊ＋ｑ）を結んだ線上に存在しているため、（ｕ Ｌ(ｙ＋ｗ)，ｖ Ｌ(ｙ＋ｗ)）は、（ｕ 0，ｖ 0）と（ｕ 2，ｖ 2）を用いた１次元の線形内・外挿で容易に求めることができる。また、同様に（ｉ＋ｐ，ｙ＋ｗ）は（ｉ＋ｐ，ｊ）と（ｉ＋ｐ，ｊ＋ｑ）を結んだ線上に存在しているため、同じように１次元の線形内・外挿で求めることができる。

このようにして求めた仮代表点の動きベクトル（ｕ Ｌ(ｙ＋ｗ)，ｖ Ｌ(ｙ＋ｗ)）と（ｕ Ｒ(ｙ＋ｗ)，ｖ Ｒ(ｙ＋ｗ)）に対して１次元の線形内・外挿を行うことにより、（ｘ＋ｗ、ｙ＋ｗ）に存在する画素の動きベクトルの水平・垂直成分をｍ倍したものである（ｕ（ｘ＋ｗ、ｙ＋ｗ），ｖ（ｘ＋ｗ、ｙ＋ｗ））を求める。この処理は、以下の式（１２）に従って行われる。

$$
\begin{aligned}
u(x+w,y+w) &= (((w_d i+w_d p-w_d x-w_n)\, u_L(y+w) \\
&\quad +(w_d x+w_n-w_d i)\, u_R(y+w))\, m) \,//\, (w_d p z) \\
v(x+w,y+w) &= (((w_d i+w_d p-w_d x-w_n)\, v_L(y+w) \\
&\quad +(w_d x+w_n-w_d i)\, v_R(y+w))\, m) \,//\, (w_d p z)
\end{aligned}
\qquad \cdots (12)
$$

ここでも上と同様にｐを２のα乗、ｍを２のｈｍ乗、ｚを２のｈｚ乗、ｗｄを２のｈｗ乗（α、ｈｍ、ｈｚ、ｗｄは負ではない整数）とすれば、式(１２におけるｐ・ｚ・ｗｄによる除算は、α＋ｈｚ＋ｈｗ－ｈｍビットの右シフト（ただし、ｈｍ＜α＋ｈｚ＋ｈｗの場合）に置換することができる。しかも、ｚ＝１６（ｈｚ＝４）とした上で、上で述べた典型的なパラメータｐ＝５１２、ｑ＝５１２、ｋ＝３２、ｍ＝１６、ｗｄ＝２、ｗｎ＝１（ｗ＝0.5）を使用した場合、シフトされるビット数は１０ビットとなり、演算に用いるレジスタに必要なビット数を大幅に抑えることが可能となる。なお、上の例では、まず代表点の動きベクトルに対して垂直方向の１次元線形内・外挿を行って仮代表点の動きベクトルを

求め、この仮代表点の動きベクトルに対して水平方向の１次元線形内・外挿を行って画素の動きベクトルを求めている。これとは逆に、仮代表点の動きベクトルを求める際には水平方向、画素の動きベクトルを求める際には垂直方向の１次元線形内・外挿を行っても同様の機能を実現することができる。

この方式では、画素の動きベクトルを求める際に式（１１）と式(１２)の２段階の処理が必要となるため、一見演算量が多くなるように思われる。しかし、一旦仮代表点の動きベクトルを求めてしまえば、これが垂直座標ｙ＋ｗに存在しているライン上のｒ個の画素すべてに対して使用できるため、全体の処理量の中に占める式(１１)の処理量はきわめて少なくなる。従って、シフトされるビット数の削減によって得られる利益（＝より小さいレジスタの活用）の影響の方が、式(１１)の計算を実行する分の演算量の増加による悪影響より大きくなる。

上記処理により（ｕ（ｘ＋ｗ、ｙ＋ｗ），ｖ（ｘ＋ｗ、ｙ＋ｗ））の値が得られた後には、以下の処理によって（ｕ（ｘ＋ｗ、ｙ＋ｗ），ｖ（ｘ＋ｗ、ｙ＋ｗ））を整数部（ｕＩ（ｘ＋ｗ、ｙ＋ｗ），ｖＩ（ｘ＋ｗ、ｙ＋ｗ））と小数部（ｕＦ（ｘ＋ｗ、ｙ＋ｗ），ｖＦ（ｘ＋ｗ、ｙ＋ｗ））に分けることができる。

$$u_I(x+w,y+w) = ((Lm + u(x+w,y+w)) >> h_m) - L$$
$$v_I(x+w,y+w) = ((Mm + v(x+w,y+w)) >> h_m) - M \quad \cdots (13)$$

$$u_F(x+w,y+w) = u(x+w,y+w) - u_I(x+w,y+w)\,m$$
$$v_F(x+w,y+w) = v(x+w,y+w) - v_I(x+w,y+w)\,m \quad \cdots (14)$$

ただし、ｕＩ（ｘ＋ｗ、ｙ＋ｗ）とｖＩ（ｘ＋ｗ、ｙ＋ｗ）は整数であり、画素の動きベクトルの整数部を表している。一方、ｕＦ（ｘ＋ｗ、ｙ＋ｗ）とｖＦ（ｘ＋ｗ、ｙ＋ｗ)はそれぞれ０以上ｍ未満の値を持つ整数であり、画素の動きベクトルの小数部をｍ倍したものである。なお、上の

例と同様にmは２のｈｍ乗であり（ｈｍは負ではない整数）、ＬとＭはシフトされる値を負の値ではなくするための十分に大きな整数である。

輝度値の内挿方式として共１次内挿が用いられる場合には、さらに以下の処理によってフレーム間予測画像内の画素の輝度値が求められる。ｘ’＝ｘ＋ｗ＋ｕＩ（ｘ＋ｗ、ｙ＋ｗ）、ｙ’＝ｙ＋ｗ＋ｖＩ（ｘ＋ｗ、ｙ＋ｗ）として、参照画像の（ｘ’，ｙ’）、（ｘ’＋１，ｙ’）、（ｘ’，ｙ’＋１）、（ｘ’＋１，ｙ’＋１）に位置する画素の輝度値をそれぞれＹａ、Ｙｂ、Ｙｃ、Ｙｄとすれば、フレーム間予測画像において（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の輝度値Ｙ（ｘ＋ｗ，ｙ＋ｗ）は

$$Y(x+w,y+w) = \big((m-v_F)\big((m-u_F)\,Y_a + u_F\,Y_b\big) + v_F\big((m-u_F)\,Y_c + u_F\,Y_d\big) + \big(m^2>>1\big)\big) >> (2h_m) \quad \cdots (15)$$

によって求められる。ただし、ｕＦ、ｖＦはそれぞれｕＦ(ｘ＋ｗ，ｙ＋ｗ)、ｖＦ(ｘ＋ｗ，ｙ＋ｗ)の略号である。

式(１２)と式(１３)では、それぞれα＋ｈｚ＋ｈｗ－ｈｍビットとｈｍビットの右シフトが行われる。このことは、式(１０)の計算の際に（α＋ｈｚ＋ｈｗ－ｈｍ）＋ｈｍ＝α＋ｈｚ＋ｈｗビットのシフトを行えば一気にｕＩ（ｘ＋ｗ、ｙ＋ｗ）とｖＩ（ｘ＋ｗ、ｙ＋ｗ）を求めることができることを意味する。このとき、α＋ｈｚ＋ｈｗを８の整数倍とすると、実装上便利である。一般的にプロセッサのレジスタは８ビット単位の大きさを持っており、８ビットのレジスタを２個（上位ビットのレジスタと下位ビットのレジスタ）つなげて１６ビットのレジスタとして使用したり、８ビットのレジスタを４個、又は１６ビットのレジスタを２個つなげて３２ビットのレジスタとして使用することができるようになっている場合が多い。ここで例えば１６ビットのシフト演算によってｕＩ（ｘ＋ｗ、ｙ＋ｗ）とｖＩ（ｘ＋ｗ、ｙ＋ｗ）の値が計算されるのであれば、わざわざシフト演算を行う必要はなくなる。つまり、シフトされ

る前の値を３２ビットのレジスタに格納しておき、その上位１６ビットを独立したレジスタとして使用すれば、その１６ビットレジスタにｕＩ（ｘ＋ｗ、ｙ＋ｗ）又はｖＩ（ｘ＋ｗ、ｙ＋ｗ）の値が格納されていることになる。

もちろん、シフトされるビット数を８の整数倍とすることは、式(１０)の処理だけでなく本明細書でこれまで述べてきたあらゆるシフト演算に対し、実装を容易にする効果を持つ。しかし、特に実行される回数の多いシフト演算（例えば画素ごとに実行されるシフト演算）に対して実装を容易にすることは重要である。また、シフトされるビット数が８の整数倍ではない場合でも、分母と分子に事前に同じビット数だけの左シフトを加えておくことにより、除算による右シフトを増やすことは可能である。例えば、６ビットの右シフトによって実現される演算があった場合に、シフトされる数値にあらかじめ４を掛けておく（これは２ビットの左シフトを行ったことに相当する）ことにより、同じ演算を８ビットの右シフトとして実現することが可能となる式(５のｕ（ｘ＋ｗ，ｙ＋ｗ）に関する式を例にとれば、あらかじめｕ０、ｕ１、ｕ２、ｕ３を４倍しておくことにより、この処理を実現することが可能となる）。ただし、このような処理を行う際には、シフトされる数に関してオーバーフローが発生しないように注意する必要がある。

画像符号化装置及び画像復号化装置には、複数の画像サイズに対応できるようになっているものが多い。この場合、例えば式(１２、１３、１４を用いたグローバル動き補償を実行したときには、画像サイズの変化に応じてシフトされるビット数が変化する現象が起こり、シフトされるビット数を８の整数倍に固定しておくことができなくなる。このような場合、次に述べるような対処法がある。例えば、上の例のようにｕＩ（ｘ＋ｗ、ｙ＋ｗ）とｖＩ（ｘ＋ｗ、ｙ＋ｗ）を求めるためにα＋ｈｚ＋ｈｗ

ビットの右シフトが必要であり、αが７～１１の値をとり得る場合を考える。このとき、αが１０より小さいときはｈ ｚ＝５、ｈ ｗ＝１、α＝１１のときはｈ ｚ＝４、ｈ ｗ＝１とすれば、シフトされるビット数を常に１６以下とすることができる。上で述べたように、シフトされるビット数が１６より小さい場合には、あらかじめシフトされる数に定数を掛けておくことにより、シフトされるビット数を擬似的に１６ビットとすることが可能である。このように、画像サイズが変化したときに他のパラメータ（例えば動きベクトルの量子化ステップサイズ）もこれに合わせて変化させることにより、シフトされるビット数が都合の良い値となるように制御することができる。しかし、上記の方法を使う場合には、復号画像の画質に著しい劣化を生じさせるほど、動きベクトルの量子化ステップサイズを大きくしてしまわないように注意する必要がある。

一般的なグローバル動き補償に本明細書で示したアルゴリズムを適用した場合には、まず１／ｎ画素精度の画像の隅の点の動きベクトルを用いて代表点の動きベクトルを１／ｋ画素精度で求め、続いて代表点の動きベクトルを用いて仮代表点の動きベクトルを１／ｚ画素精度で求めた後に、この仮代表点の動きベクトルを用いて画素の動きベクトルが１／ｍ画素精度で求められる。画像の隅の点の動きベクトルが動きパラメータとして伝送される場合には、このパラメータによる共１次内・外挿を正確に近似するという意味で、ｋをできるだけ大きな値にすることが望ましい。しかし、いずれにせよ代表点の動きベクトルの水平・垂直成分には、量子化の影響で１／（２ｋ）以下の絶対値をもつ誤差が含まれることになる。近似を正確にするという意味からは、仮代表点の動きベクトルも、なるべく精度を高くすることが望ましい。しかし、仮代表点の動きベクトルは代表点の動きベクトルを用いて求められるため、代表点の動きベクトル以上の精度を持たせて計算してもあまり意味がない。従

って、演算に必要なビット数を抑える意味でz≦kとすることが望ましい。また、同様の理由により、m≦zとすることが望ましい。

これまで共一次内・外挿を用いたグローバル動き補償に関して説明してきたが、線形内・外挿を用いた場合も同様の処理を導入することによって、シフトされるビット数を制御することができる。例えば、（i，j）、（i＋p，j）、（i，j＋q）に存在する（i、j、p、qは整数）代表点の動きベクトルの水平・垂直成分をk倍したものをそれぞれ（u0，v0）、（u1，v1）、（u2，v2）とする（u0、v0、u1、v1、u2、v2は整数）。このとき、画素（x＋w，y＋w）の動きベクトルの水平・垂直成分をm倍したもの（u(x＋w，y＋w)，v(x＋w，y＋w)）は以下の式（16）で表すことができる（ただし、x、y、u(x＋w，y＋w)、v(x＋w，y＋w)は整数、wの定義は上と同じ）。

$$
\begin{aligned}
u(x+w,y+w) &= (((u_1-u_0)(w_dx+w_n-w_di)\,q \\
&\quad + (u_2-u_0)(w_dy+w_n-w_dj)\,p + u_0w_dpq)\,m) \\
&\quad // (w_dpqk) \\
v(x+w,y+w) &= (((v_1-v_0)(w_dx+w_n-w_di)\,q \\
&\quad + (v_2-v_0)(w_dy+w_n-w_dj)\,p + v_0w_dpq)\,m) \\
&\quad // (w_dpqk)
\end{aligned}
\qquad \cdots (16)
$$

この場合もp、q、k、m、wdがそれぞれ2のα乗、β乗、hk乗、hm乗、hw乗であり（α、β、hk、hm、hwは負ではない整数）、さらにα≧βであるとすれば、この式は

$$
\begin{aligned}
u(x+w,y+w) &= \left(\left((u_1-u_0)(w_dx+w_n-w_di)\,2^{\alpha-\beta}\right.\right. \\
&\quad \left.\left. + (u_2-u_0)(w_dy+w_n-w_dj) + u_0w_dp\right)m\right) // (w_dpk) \\
v(x+w,y+w) &= \left(\left((v_1-v_0)(w_dx+w_n-w_di)\,2^{\alpha-\beta}\right.\right. \\
&\quad \left.\left. + (v_2-v_0)(w_dy+w_n-w_dj) + v_0w_dp\right)m\right) // (w_dpk)
\end{aligned}
\qquad \cdots (17)
$$

と書き換えることができ、共１次内・外挿を用いた場合と同様に、α＋ｈｋ＋ｈｗビットの右シフトにより（ｘ＋ｗ，ｙ＋ｗ）に存在する画素の動きベクトルの整数部を求めることができる。従って、α＋ｈｋ＋ｈｗが８の整数倍となるようにすれば、上と同様の理由により実装を行いやすくすることができる。なお、α＜βの場合には、シフトされるビット数はβ＋ｈｋ＋ｈｗビットとなる。

以下、上記フレーム間予測画像の合成処理を採用する本発明の符号化方法及び復号化方法を実施する画像符号化装置、復号化装置の構成について述べる。

第６図は、本発明によるに画像符号化装置の一実施形態の構成を示す。同図において、動き補償処理部６１６を除いては、従来知られている画像符号化装置と実質的に同じである。

減算器６０２は入力画像（符号化しようとする現フレームの原画像）６０１とフレーム間／フレーム内符号化切り換えスイッチ６１９の出力画像６１３（フレーム間予測画像）との差を計算し、誤差画像６０３を出力する。この誤差画像は、ＤＣＴ変換器６０４でＤＣＴ係数に変換された後に量子化器６０５で量子化され、量子化ＤＣＴ係数６０６となる。この量子化ＤＣＴ係数は伝送情報として通信路に出力されると同時に、符号化器内でもフレーム間予測画像を合成するために使用される。

以下にフレーム予測画像合成の手順を説明する。量子化ＤＣＴ係数６０６は、逆量子化器６０８と逆ＤＣＴ変換器６０９を経て復号誤差画像６１０（受信側で再生される誤差画像と同じ画像）となる。これに、加算器６１１においてフレーム間／フレーム内符号化切り換えスイッチ６１９の出力画像６１３（後述）が加えられ、現フレームの復号画像６１２（受信側で再生される現フレームの復号画像と同じ画像）を得る。この画像は一旦フレームメモリ６１４に蓄えられ、１フレームの時間だけ

遅延される。従って、現時点では、フレームメモリ６１４は前フレームの復号画像６１５を出力している。この前フレームの復号画像と現フレームの入力画像６０１が動き補償処理部６１６に入力される。動き補償処理部６１６は前述のフレーム間予測画像の合成を行う。その構成については後で述べる。

予測画像６１７は、「０」信号６１８と共にフレーム間／フレーム内符号化切り換えスイッチ６１９に入力される。このスイッチは、両入力のいずれかを選択することにより、フレーム間符号化とフレーム内符号化を切り換える。予測画像６１７が選択された場合（第６図はこの場合を表している）には、フレーム間符号化が行われる。一方、「０」信号が選択された場合には、入力画像がそのままＤＣＴ符号化されて通信路に出力されるため、フレーム内符号化が行われる。受信側が正しく復号化画像を得るためには、送信側でフレーム間符号化が行われたかフレーム内符号化が行われたかを知る必要がある。このため、識別フラグ６２１が通信路へ出力される。最終的なＨ．２６１符号化ビットストリーム６２３は多重化器６２２で量子化ＤＣＴ係数、動きベクトル、フレーム内／フレーム間識別フラグの情報を多重化することによって得られる。

第７図は、第６図の符号化器が出力した符号化ビットストリームを受信する復号化器７００の構成例を示す。受信したビットストリーム７１７は、分離器７１６で量子化ＤＣＴ係数７０１、動きベクトル７０２、フレーム内／フレーム間識別フラグ７０３に分離される。量子化ＤＣＴ係数７０１は逆量子化器７０４と逆ＤＣＴ変換器７０５を経て復号化された誤差画像７０６となる。この誤差画像は加算器７０７でフレーム間／フレーム内符号化切り換えスイッチ７１４の出力画像７１５を加算され、復号化画像７０８として出力される。フレーム間／フレーム内符号化切り換えスイッチはフレーム間／フレーム内符号化識別フラグ７０３

に従って、出力を切り換える。フレーム間符号化を行う場合に用いる予測画像７１２は、予測画像合成部７１１において合成される。ここでは、フレームメモリ７０９に蓄えられている前フレームの復号画像７１０に対して、受信した動きベクトル７０２に従って位置を移動させる処理が行われる。一方フレーム内符号化の場合、フレーム間／フレーム内符号化切り換えスイッチは、「０」信号７１３をそのまま出力する。

第８図は、代表点の動きベクトルを伝送する線形内・外挿に基づくグローバル動き補償方式を採用した画像符号化器の動き補償処理部６１６の構成例を示す。第６図と同じ番号は同じものを指す。グローバル動き推定部８０２で前フレームの復号画像６１５と現フレームの原画像６０１との間でグローバル動き補償に関する動き推定が行われ、グローバル動き補償のパラメータ（例えば、上記ｕａ、ｖａ、ｕｂ、ｖｂ、ｕｃ、ｖｃ、ｕｄ、ｖｄの値）が推定される。これらの値に関する情報８０３は動き情報６２０の一部として伝送される。グローバル動き補償の予測画像８０４は式(３)を用いてグローバル動き補償予測画像合成部８０８で合成され、ブロックマッチング部８０５に供給される。ここでは、グローバル動き補償の予測画像と現フレームの原画像との間でブロックマッチングによる動き補償（動き推定と予測画像合成）が行われ、ブロックの動きベクトル情報８０６と最終的な予測画像６１７が得られる。この動きベクトル情報は動きパラメータ情報と多重化部８０７において多重化され、動き情報６２０として出力される。

第１０図は、第７図の予測画像合成部７１１の構成例を示す。他の図と同じ番号は同じものを指す。前フレームの復号画像７１０に対し、動き情報７０２から分割部１００２において抽出されたグローバル動き補償パラメータ８０３を用いて、グローバル動き補償予測画像合成部８０８においてグローバル動き補償の予測画像８０４が合成される。画像８

０４はブロックマッチング予測画像合成部１００１に供給され、動き情報７０２から抽出されたブロックマッチングの動きベクトル情報８０６を用いて最終的な予測画像７１２が合成される。

第９図は、動き補償処理部６１６の他の構成例をに示す。第６図と同じ番号は同じものを指す。この例では、各ブロックに関してグローバル動き補償かブロックマッチングのいずれかが適用される。前フレームの復号画像６１５と現フレームの原画像６０１との間で、グローバル動き推定部９０２、グローバル動き補償予測画像合成部９１１ではグローバル動き補償、ブロックマッチング部９０５ではブロックマッチングにより、それぞれ独立に動き補償の処理が行われる。選択スイッチ９０８は、グローバル動き補償による予測画像９０３とブロックマッチングによる予測画像９０６の間でブロックごとに最適な方式を選択する。グローバル動き補償パラメータ９０４、ブロックごとの動きベクトル９０７、グローバル動き補償／ブロックマッチングの選択情報９０９は多重化部９１０で多重化され、動き情報６２０として出力される。

第 11 図は、動き補償処理部９０１を用いる画像符号化器が生成するビットストリームを復号化する復号化器の、予測画像合成部１１０３の構成例を示す。他の図と同じ番号は同じものを指す。前フレームの復号画像７１０に対し、動き情報７０２から分割部１１０２において抽出されたグローバル動き補償パラメータ９０４を用いて、グローバル動き補償予測画像合成部９１１においてグローバル動き補償の予測画像９０３が合成される。また、これとは独立に前フレームの復号画像７１０に対し、動き情報７０２から抽出されたブロックマッチングの動きベクトル情報９０７を用いてブロックマッチング予測画像合成部１１０１においてブロックマッチングの予測画像９０６が合成される。選択スイッチ１１０４は、グローバル動き補償による予測画像９０３とブロックマッチ

ングによる予測画像９０６の間で、動き情報７０２から抽出された選択情報９０９に基づいて、ブロックごとに一方の方式を選択する。このブロックごとの選択処理を経て、最終的な予測画像７１２が合成される。

第 12 図は、上述の本発明によるグローバル動き補償予測画像合成部の機能的構成を示す。グローバル動き補償パラメータとして、画像の隅の点の動きベクトルが伝送されるとする。この画像の隅の点の動きベクトルに関する情報１２０４を用いて演算処理部１２０５において式(9)、（１０）を用いて代表点の動きベクトルが計算される。この代表点の動きベクトルに関する情報１２０６を用いて演算処理部１２０７では、式(１１)を用いてライン（垂直座標が共通の値である画素）ごとに仮代表点の動きベクトルが計算される。さらにこの仮代表点の動きベクトルに関する情報１２０８を活用して演算処理部１２０９では画素ごとの動きベクトルが式(１２)により計算される。一方、処理部１２１１では、この画素ごとの動きベクトルに関する情報１２１０と、前フレームの復号画像１２０２を用いてグローバル動き補償の予測画像１２０３が合成、出力される。

本発明は、専用回路・専用チップを用いる画像符号化装置、画像復号化装置の他に、汎用プロセッサを用いるソフトウェア画像符号化装置、ソフトウェア画像復号化装置にも適用することができる。

図４及び図５は、それぞれソフトウェア画像符号化装置４００とソフトウェア画像復号化装置５００の例を示す。ソフトウェア符号化装置４００では、入力画像４０１は、入力フレームメモリ４０２に蓄えられ、汎用プロセッサ４０３は入力フレームメモリ４０２から情報を読み込んで符号化の処理を行う。汎用プロセッサ４０３を駆動するためのプログラムはハードディスクやフロッピーディスクなどによる蓄積デバイス４０８から読み出されてプログラム用メモリ４０４に蓄えられる。また、

汎用プロセッサ４０３は、処理用メモリ４０５を活用して符号化の処理を行う。汎用プロセッサ４０３が出力する符号化情報は、一旦出力バッファ４０６に蓄えられた後に符号化ビットストリーム４０７として出力される。

図１３は、図４に示したソフトウェア符号化器上で動作する符号化ソフトウェアのフローチャートを示す。まずステップ１３０１で画像符号化処理が開始され、ステップ１３０２で変数Ｎに０が代入される。続いてステップ１３０３、ステップ１３０４でＮの値が１００である場合には、０が代入される。Ｎはフレーム数のカウンタであり、１枚のフレームの処理が終了する度に１が加算され、符号化を行う際には０～９９の値をとることが許される。Ｎの値が０であるときには符号化中のフレームはＩフレーム（動き補償は行わず、全てのブロックでフレーム内符号化が行われるフレーム）であり、それ以外のときはＰフレーム（動き補償を行うブロックの存在するフレーム）となる。Ｎの値が１００であることは、Ｐフレーム が９９枚符号化された後にＩフレームが１枚符号化されることを意味している。Ｎの最適な値は符号化器の性能や符号化器が使用される環境により変化する。この例では１００という値を使用したが、これはＮの値が必ず１００でなければならいことを意味しているわけではない。フレームタイプ（Ｉ又はＰ）の決定と出力はステップ１３０５で行われる。Ｎの値が０である場合にはフレームタイプの識別情報として’Ｉ’が出力バッファに出力され、これから符号化処理を行うフレームはＩフレームとなる。なお、ここで「出力バッファに出力される」とは、出力バッファ（第４図の４０６）に蓄えられた後に符号化ビットストリームの一部として符号化装置から外部に出力される。Ｎが０ではない場合には、フレームタイプの識別情報として’Ｐ’が出力バッファに出力され、これから符号化処理を行うフレームはＰフレームとな

る。ステップ１３０６では入力画像はフレームメモリＡに蓄えられる。なお、ここで述べたフレームメモリＡとは、ソフトウェア符号化器のメモリ領域（例えば、第４図のメモリ４０５内にこのメモリ領域が確保される）の一部を意味している。ステップ１３０７では、現在符号化中のフレームがＩフレームであるか否かが判定される。そして、Ｉフレームではない場合にはステップ１３０８で動き推定・動き補償処理が行われる。このステップ１３０８における処理の詳細を表すフローチャートの例を第１４図に示す。まず、ステップ１４０１でフレームメモリＡとＢ（フレームメモリＢには前フレームの復号画像が格納されている）に蓄えられた画像の間でグローバル動き推定が行われ、グローバル動きパラメータとして、画像の隅の点の動きベクトルが出力バッファに出力される。ステップ１４０２では、この画像の隅の点の動きベクトルを用いて式(9)、（１０）により代表点の動きベクトルが計算される。続いてステップ１４０３では、変数Ｍに０が代入される。Ｍは画像内のラインの番号を表し、Ｍが０であることは、画像の最も上のラインを処理中であることを意味し、Ｍが画像のライン数から１を引いた値であるときには、画像の最も下のラインを処理中であることを意味する。ステップ１４０２で計算された代表点の動きベクトルを用いて、ステップ１４０４では式(１１)により第Ｍラインの仮代表点の動きベクトルが計算される。そしてこの仮代表点の動きベクトルを活用してステップ１４０５では第Ｍラインに含まれる画素全ての動きベクトルが式(１２)により計算され、求められた動きベクトルに従って、フレームメモリＢに格納されている前フレームの復号画像を用いてグローバル動き補償予測画像の第Ｍラインが合成され、フレームメモリＦに蓄えられる。ステップ１４０６ではＭの値に１が加えられ、ステップ１４０７ではＭの値が画像のライン数に等しければステップ１４０８へ、等しく無ければステップ１４０４に

移動する。ステップ１４０８の処理が開始される時点では、フレームメモリＤには、グローバル動き補償による予測画像が蓄えられている。ステップ１４０８以降では、ブロックマッチングの処理が行われる。まずステップ１４０８では、フレームメモリＦとフレームメモリＡ（入力画像）との間でブロックごとに動き推定の処理が行われ、各ブロックの動きベクトルが求められ、その動きベクトルは出力バッファに出力される。続いてこの動きベクトルと、フレームメモリＦに蓄えられた画像を用いてステップ１４０９ではブロックマッチングによる予測画像が合成され、これが最終的な予測画像となってフレームメモリＣに蓄えられる。そしてステップ１４１０ではフレームメモリＡとＣの差分画像が求められ、これがフレームメモリＡに蓄えられる。

ここで第 13 図に戻る。ステップ１３０８における処理が開始される直前、フレームメモリＡには、現フレームがＩフレームである場合には入力画像が、現フレームがＰフレームである場合には入力画像と予測画像の差分画像が蓄えられている。ステップ１３０８では、このフレームメモリＡに蓄えられた画像に対してＤＣＴが適用され、ここで計算されたＤＣＴ係数は量子化された後に出力バッファに出力される。さらにステップ１３１０で、この量子化ＤＣＴ係数には逆量子化され、逆ＤＣＴが適用され、この結果得られた画像はフレームメモリＢに格納される。続いてステップ１３１１では、再び現フレームがＩフレームであるか否かが判定され、Ｉフレームではない場合にはステップ１３１２でフレームメモリＢとＣの画像が加算され、この結果がフレームメモリＢに格納される。ここで、１フレーム分の符号化処理が終了することになる。そして、ステップ１３１３の処理が行われる直前にフレームメモリＢに格納されている画像は、符号化処理が終了したばかりのフレームの再生画像（復号側で得られるものと同じ）である。ステップ１３１３では、符

号化が終了したフレームが最後のフレームであるか否かが判定され、最後のフレームであれば、符号化処理が終了する。最後のフレームではない場合には、ステップ１３１４でNに１が加算され、再びステップ１３０３に戻って次のフレームの符号化処理が開始される。なお、ここで説明したフローチャートはグローバル動き補償を行なった結果合成されたグローバル動き補償予測画像に対してブロックマッチングを適用する方法（第８図の動き補償処理部８０１を使用する装置に対応する方法）に関するものであるが、グローバル動き補償とブロックマッチングを並列に行う方法（第９図の動き補償処理部９０１を使用する装置に対応する方法）に関するフローチャートもわずかの変更を加えるのみで作成できることは明らかである。

一方、ソフトウェア復号化装置５００では、入力された符号化ビットストリーム５０１は、一旦入力バッファ５０２に蓄えられた後に、汎用プロセッサ５０３に読み込まれる。汎用プロセッサ５０３はハードディスクやフロッピーディスクなどによる蓄積デバイス５０８から読み出されたプログラムを蓄えるプログラム用メモリ５０４および処理用メモリ５０５を活用して復号化処理を行う。この結果得られた復号化画像は、一旦出力フレームメモリ５０６に蓄えられた後に、出力画像５０７として出力される。

第 15 図は、第５図に示したソフトウェア復号化装置上で動作する復号化ソフトウェアのフローチャートを示す。１５０１で処理が開始され、まずステップ１５０２で入力情報があるか否かが判定される。ここで入力情報が無ければステップ１５０３で復号化の処理を終了する。入力情報がある場合には、まず、ステップ１５０４でフレームタイプ情報が入力される。なお、この「入力される」とは、入力バッファ５０２に蓄えられた情報を読み込むことを意味している。ステップ１５０５では、読

み込んだフレームタイプ情報が’Ｉ’であるか否かが判定される。そして、’Ｉ’ではない場合には、ステップ１５０６で予測画像合成処理が行われる。このステップ１５０６で行われる処理の詳細をフローチャートを第16図に示す。

まず、ステップ16０１で画像の隅の点の動きベクトルが入力される。ステップ16０２では、この画像の隅の点の動きベクトルを用いて式(9)、（１０）により代表点の動きベクトルが計算される。続いてステップ16０３では、変数Mに0が代入される。Mは画像内のラインの番号を表し、Mが0であることは、画像の最も上のラインを処理中であることを意味し、Mが画像のライン数から１を引いた値であるときには、画像の最も下のラインを処理中であることを意味する。ステップ16０２で計算された代表点の動きベクトルを用いて、ステップ16０４では式(１１)により第Mラインの仮代表点の動きベクトルが計算される。そしてこの仮代表点の動きベクトルを活用してステップ16０５では第Mラインに含まれる画素すべての動きベクトルが式(１２)により計算され、求められた動きベクトルに従って、フレームメモリＥに格納されている前フレームの復号画像を用いてグローバル動き補償予測画像の第Mラインが合成され、フレームメモリＧに蓄えられる。なお、ここで述べたフレームメモリＧとは、ソフトウェア復号化器のメモリ５０５の領域の一部を意味している。ステップ16０６ではMの値に１が加えられ、ステップ16０７ではMの値が画像のライン数に等しければステップ16０８へ、等しく無ければステップ16０４に移動する。ステップ16０８の処理が開始される時点では、フレームメモリＧには、グローバル動き補償による予測画像が蓄えられている。ステップ16０８では、ブロックマッチングの処理が行われる。ブロックごとの動きベクトル情報が入力され、この動きベクトルとフレームメモリＧに格納された画像を用いてブ

ロックマッチングによる予測画像が合成され、この予測画像はフレームメモリＤに格納される。

ここで第 15 図に戻る。ステップ１５０７では量子化ＤＣＴ係数が入力され、これに逆量子化、逆ＤＣＴを適用して得られた画像がフレームメモリＥに格納される。ステップ１５０８では、再び現在復号化中のフレームがＩフレームであるか否かが判定される。そして、Ｉフレームではない場合には、ステップ１５０９でフレームメモリＤとＥに格納された画像が加算され、この結果の画像がフレームメモリＥに格納される。ステップ１５１０の処理を行う直前にフレームメモリＥに格納されている画像が、再生画像となる。ステップ１５１０では、このフレームメモリＥに格納された画像が出力フレームメモリ５０６に出力され、そのまま出力画像として復号化器から出力される。こうして１フレーム分の復号化処理が終了し、処理は再びステップ１５０２に戻る。

第４図と第５図に示したソフトウェア画像符号化装置、ソフトウェア画像復号化装置に本明細書で示したフレーム間予測画像の合成方法を実行するプログラムを実行させると、グローバル動き補償やワーピング予測の処理をより少ない演算量で実現することが可能となる。このため、本発明を用いない場合と比較して、消費電力の低減、装置の低価格化、より大きな画像を実時間で処理できるようになる、画像符号化・復号化以外の処理を含む同時並列処理を行うことが可能となる、等の効果を期待することができる。また、本明細書で示したアルゴリズムを用いることにより、従来の画像復号化装置では演算能力の限界から実時間で再生できなかったような圧縮画像データを、実時間で再生することが可能となる。

以上本発明の実施形態について述べたが、以下のような実施形態も本発明に含まれる。

（１）従来型の画像符号化方法では、フレーム間予測を行った後に離散コサイン変換などによる誤差符号化が行われるが、フレーム間予測画像をそのまま再生画像として使用する画像符号化方法・復号化方法に対しても、本発明は有効である。

（２）本明細書では、画像の形状は長方形であることを仮定したが、長方形以外の任意の形状を持つ画像にも、本発明は適用可能である。この場合、まず任意形状の画像を囲む長方形に対して本発明の処理を適用し、任意形状画像に含まれる画素に対してのみ動きベクトルを求める演算を行えば良い。

（３）本明細書では、ｐ又はｑの値が２の負ではない整数乗であることを前提として２段階の処理による動きベクトルの内・外挿アルゴリズムを示した。しかし、ｐ及びｑが２の負ではない整数乗ではない場合でも、この２段階処理アルゴリズムは除算における分母の値を小さくするという効果を持っており、レジスタのオーバーフローを防ぐ意味で有効である。

産業上の利用可能性

図１７に、本明細書で示した予測画像合成方法を用いる符号化・復号化装置の具体例を示す。（a）は、パソコン１７０１に画像符号化・復号化用のソフトウェアを組み込むことにより、画像符号化・復号化装置として活用する場合を示す。このソフトウェアは何らかの蓄積メディア（ＣＤ－ＲＯＭ、フロッピーディスク、ハードディスクなど）に記録されており、これをパソコンが読み込んで使用する。また、さらに何らかの通信回線にこのパソコンを接続することにより、映像通信端末として活用することも可能となる。

（b）は本発明による符号化方法による動画像情報を蓄積メディア１

７０２に記録した符号化ビットストリームを読み取り、本発明による装置を持つ再生装置１７０３で再生し、再生された映像信号をテレビモニタ１７０４に表示する場合を示す。再生装置１７０３は符号化ビットストリームを読み取るだけであり、テレビモニタ１７０４内に復号化装置が組み込まれている場合もある。

(c)は、ディジタル放送用のテレビ受信機１７０５に本発明の復号化装置を組み込んだ場合を示す。また、（ｄ）は、ケーブルテレビ用のケーブル１７０８又は衛星／地上波放送のアンテナに接続されたセットトップボックス１７０９内に復号化装置を実装し、これをテレビモニタ１７１０で再生する場合を示す。このとき、（ｂ）の１７０４と同様に、セットトップボックスではなく、テレビモニタ内に符号化装置を組み込んでも良い。

（ｅ）は、ディジタル携帯端末１７０６に本発明の符号化器、復号化器を組み込んだ場合を示す。ディジタル携帯端末の場合、符号器・復号化器を両方持つ送受信型の端末の他に、符号化器のみの送信端末、復号化器のみの受信端末の３通りの実装形式のいずれでもよい。

（ｆ）は、動画像撮影用のカメラ１７０７の中に符号化装置を組み込む場合を示す。また、カメラ１７０７は映像信号を取り込むのみであり、これを専用の符号化装置１７１１に組み込む構成でもよい。この図に示したいずれの装置・システムに関しても、本明細書に示した方法を実装することにより、従来の技術を活用した場合と比較して、装置を簡略化することが可能となる。

請求の範囲

1．　第１のフレーム画像と時間的に異なる第２のフレーム画像から第１のフレーム画像の予測画像を合成する方法において、

第１のフレーム画像と第２のフレーム画像から第１のフレーム画像の座標（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）に（ｉ、ｊ、ｐ、ｑは整数）を持つ４つの代表点の動きベクトル動きベクトル（動きベクトルの水平・垂直成分が１／ｋの整数倍の値をとり（ただし、ｋは2のｈｋ乗、かつｈｋは負ではない整数）を求める第１ステップと、

画素のサンプリング間隔を水平、垂直方向共に１として、サンプリング点が座の水平、垂直成分が、共に整数にｗを加えた数である点の上に存在している画像を対象として（ただし、ｗ＝ｗｎ／ｗｄ、かつｗｎは負ではない整数、かつｗｄは２のｈｗ乗、かつｈｗは負ではない整数、かつｗｎ＜ｗｄ）、４個の代表点における動きベクトルに対し、共１次内・外挿を行うことによって座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルを計算する第２ステップを有し、

上記第２ステップが、座標（ｉ，ｊ）と（ｉ，ｊ＋ｑ）に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標（ｉ，ｙ＋ｗ）に位置する点の動きベクトルの水平・垂直成分をそれぞれ１／ｚの整数倍をとる数値として（ただし、ｚは2のｈｚ乗、かつｈｚは負ではない整数）求め、さらに座標（ｉ＋ｐ，ｊ）と（ｉ＋ｐ，ｊ＋ｑ）に位置する代表点の動きベクトルに対して

線形内・外挿を行うことにより、座標（ｉ＋ｐ，ｙ＋ｗ）に位置する点の動きベクトルの水平・垂直成分をそれぞれ１／ｚの整数倍をとる

数値として求める第３ステップと、

その後に、座標（ｉ，ｙ＋ｗ）と（ｉ＋ｐ，ｙ＋ｗ）に位置する上記２個の動きベクトルに対して線形内・外挿を行うことにより、座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルの水平・垂直成分をそれぞれ１／ｍの整数倍をとる数値として（ただし、ｍは2のｈｍ乗、かつｈｍは負ではない整数）求める第４ステップ途をもつことを特徴とするフレーム間予測画像の合成方法。

２．　第１のフレーム画像と時間的に異なる第２のフレーム画像から第１のフレーム画像の予測画像を合成する方法において、

第１のフレーム画像と第２のフレーム画像から第１のフレーム画像の座標（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）に（ｉ、ｊ、ｐ、ｑは整数）を持つ４つの代表点の動きベクトル動きベクトル（動きベクトルの水平・垂直成分が１／ｋの整数倍の値をとり（ただし、ｋは2のｈｋ乗、かつｈｋは負ではない整数）を求める第１ステップと、

画素のサンプリング間隔を水平、垂直方向共に１として、サンプリング点が座標の水平、垂直成分が、共に整数にｗを加えた数である点の上に存在している画像を対象として（ただし、ｗ＝ｗｎ／ｗｄ、かつｗｎは負ではない整数、かつｗｄは２のｈｗ乗、かつｈｗは負ではない整数、かつｗｎ＜ｗｄ）、４個の代表点における動きベクトルに対し、共１次内・外挿を行うことによって座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルを計算する第２ステップを有し、

上記第２ステップが、座標（ｉ，ｊ）と（ｉ＋ｐ，ｊ）に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標

（ｘ＋ｗ，ｊ）に位置する点の動きベクトルの水平・垂直成分をそれぞれ１／ｚの整数倍をとる数値として（ただし、ｚは2のｈｚ乗、かつｈｚは負ではない整数）求め、さらに座標（ｉ，ｊ＋ｑ）と（ｉ＋ｐ，ｊ＋ｑ）に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標（ｘ＋ｗ，ｊ＋ｑ）に位置する点の動きベクトルの水平・垂直成分をそれぞれ１／ｚの整数倍をとる数値として求める第３ステップと、

その後に、座標（ｘ＋ｗ，ｊ）と（ｘ＋ｗ，ｊ＋ｐ）に位置する上記２個の動きベクトルに対して線形内・外挿を行うことにより、座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルの水平・垂直成分をそれぞれ１／ｍの整数倍をとる数値として（ただし、ｍは2のｈｍ乗、かつｈｍは負ではない整数）求める第４ステップとをもつことを特徴とするフレーム間予測画像の合成方法。

３．　上記座標（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）に位置する代表点の動きベクトルの水平・垂直成分をｋ倍したものである（ｕ0，ｖ0）、（ｕ1，ｖ1）、（ｕ2，ｖ2）、（ｕ3，ｖ3）を用いて座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルを求めるときに、座標（ｉ，ｙ＋ｗ）に位置する点の動きベクトルの水平・垂直成分をそれぞれｚ倍したものである（ｕＬ(ｙ＋ｗ)，ｖＬ(ｙ＋ｗ)）を、

ｕＬ(ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｕ0＋（ｙ・ｗｄ＋ｗｎ）ｕ2）ｚ）////（ｑ・ｋ・ｗｄ），

ｖＬ(ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｖ0＋（ｙ・ｗｄ＋ｗｎ）ｖ2）ｚ）////（ｑ・ｋ・ｗｄ）

を計算することにより（ただし、「////」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求め、

更に座標（ｉ＋ｐ，ｙ＋ｗ）に位置する点の動きベクトルの水平・垂直成分をそれぞれｚ倍したものである（ｕＲ(ｙ＋ｗ)，ｖＲ(ｙ＋ｗ)）を、

ｕＲ(ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｕ１＋（ｙ・ｗｄ＋ｗｎ）ｕ３）ｚ）////（ｑ・ｋ・ｗｄ）、

ｖＲ(ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｖ１＋（ｙ・ｗｄ＋ｗｎ）ｖ３）ｚ）////（ｑ・ｋ・ｗｄ）、

を計算することにより求めた後に、座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルの水平・垂直成分をそれぞれｍ倍したものである（ｕ(ｘ＋ｗ，ｙ＋ｗ)，ｖ(ｘ＋ｗ，ｙ＋ｗ)）を

ｕ(ｘ＋ｗ，ｙ＋ｗ)＝（（（ｐ・ｗｄ－ｘ・ｗｄ－ｗｎ）ｕＬ(ｙ＋ｗ)＋（ｘ・ｗｄ＋ｗｎ）ｕＲ(ｙ＋ｗ)）ｍ）//（ｐ・ｚ・ｗｄ）

ｖ(ｘ＋ｗ，ｙ＋ｗ)＝（（（ｐ・ｗｄ－ｘ・ｗｄ－ｗｎ）ｖＬ(ｙ＋ｗ)＋（ｘ・ｗｄ＋ｗｎ）ｖＲ(ｙ＋ｗ)）ｍ）//（ｐ・ｚ・ｗｄ）

を計算することによって（ただし、「//」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求めることを特徴とする請求項１に記載のフレーム間予測画像の合成方法。

４．　座標（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）に位置する代表点の動きベクトルの水平・垂直成分をｋ倍したものである（ｕ０，ｖ０）、（ｕ１，ｖ１）、（ｕ２，ｖ２）、（ｕ３，ｖ

3）を用いて座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルを求めるときに、座標（ｘ＋ｗ，ｊ）に位置する点の動きベクトルの水平・垂直成分をそれぞれｚ倍したものである（ｕＴ(ｘ＋ｗ)，ｖＴ(ｘ＋ｗ)）を、

ｕＴ(ｘ＋ｗ)＝（（（ｐ・ｗｄ－ｘ・ｗｄ－ｗｎ）ｕ0＋（ｘ・ｗｄ＋ｗｎ）ｕ1）ｚ）////（ｐ・ｋ・ｗｄ）、

ｖＴ(ｘ＋ｗ)＝（（（ｐ・ｗｄ－ｘ・ｗｄ－ｗｎ）ｖ0＋（ｘ・ｗｄ＋ｗｎ）ｖ1）ｚ）////（ｐ・ｋ・ｗｄ）、

を計算することにより（ただし、「////」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求め、

さらに座標（ｘ＋ｗ，ｊ＋ｐ）に位置する点の動きベクトルの水平・垂直成分をそれぞれｚ倍したものである（ｕＢ(ｙ＋ｗ)，ｖＢ(ｙ＋ｗ)）を、

ｕＢ(ｘ＋ｗ)＝（（（ｐ・ｗｄ－ｘ・ｗｄ－ｗｎ）ｕ2＋（ｘ・ｗｄ＋ｗｎ）ｕ3）ｚ）////（ｐ・ｋ・ｗｄ），

ｖＢ(ｘ＋ｗ)＝（（（ｐ・ｗｄ－ｘ・ｗｄ－ｗｎ）ｖ2＋（ｘ・ｗｄ＋ｗｎ）ｖ3）ｚ）////（ｐ・ｋ・ｗｄ），

を計算することにより求めた後に、座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルの水平・垂直成分をそれぞれｍ倍したものである（ｕ(ｘ＋ｗ，ｙ＋ｗ)，ｖ(ｘ＋ｗ，ｙ＋ｗ)）を

ｕ(ｘ＋ｗ，ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｕＴ(ｘ＋ｗ)＋（ｙ・ｗｄ＋ｗｎ）ｕＢ(ｘ＋ｗ)）ｍ）//（ｑ・ｚ・ｗｄ），

ｖ(ｘ＋ｗ，ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｖＴ(ｘ＋ｗ)＋

（y・wd+wn）vB(x+w)）m）//（q・z・wd），

を計算することによって（ただし、「//」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求めることを特徴とする請求項２に記載のフレーム間予測画像の合成方法。

５．　pの絶対値が２のα乗（αは負ではない整数）であることを特徴とする請求項１に記載のフレーム間予測画像の合成方法。

６．　qの絶対値が２のβ乗（βは負ではない整数）であることを特徴とする請求項２又は４に記載のフレーム間予測画像の合成方法。

７．　pとqの絶対値がそれぞれ２のα乗とβ乗（α、βは負ではない整数）であることを特徴とする請求項１に記載のフレーム間予測画像の合成方法。

８．　pとqの絶対値がそれぞれ２のα乗とβ乗（α、βは負ではない整数）であることを特徴とする請求項２に記載のフレーム間予測画像の合成方法。

９．　α+hzが８の正の整数倍であり、かつwが０であることを特徴とする請求項５に記載のフレーム間予測画像の合成方法。

１０．　β+hzが８の正の整数倍であり、かつwが０であることを特徴とする請求項６に記載のフレーム間予測画像の合成方法。

１１．　α+hz+hwが８の正の整数倍であり、かつw＞０であることを特徴とする請求項５記載のフレーム間予測画像の合成方法。

１２．　β+hz+hwが８の正の整数倍であり、かつw＞０であることを特徴とする請求項６に記載のフレーム間予測画像の合成方法。

１３．　複数の異なるαの値に対応し、α+hzが１６以下となるよう

にｈｚの値をαの値に応じて変化させることを特徴とする請求項９に記載のフレーム間予測画像の合成方法。

１４．　複数の異なるβの値に対応し、β＋ｈｚが１６以下となるようにｈｚの値をβの値に応じて変化させることを特徴とする請求項１０に記載のフレーム間予測画像の合成方法。

１５．　複数の異なるαの値に対応し、α＋ｈｚ＋ｈｗが１６以下となるようにｈｚの値をαの値に応じて変化させることを特徴とする請求項１１に記載のフレーム間予測画像の合成方法。

１６．　複数の異なるβの値に対応し、β＋ｈｚ＋ｈｗが１６以下となるようにｈｚの値をβの値に応じて変化させることを特徴とする請求項１２に記載のフレーム間予測画像の合成方法。

１７．　ｚ≧ｍであることを特徴とする請求項１ないし１６に記載のフレーム間予測画像の合成方法。

１８．　ｋ≧ｚであることを特徴とする請求項１ないし１６に記載のフレーム間予測画像の合成方法。

１９．　ｐとｑの絶対値がそれぞれ画像の水平と垂直の画素数と異なることを特徴とする請求項１ないし１６に記載のフレーム間予測画像の合成方法。

２０．　ｒを画像の水平方向の画素数、ｓを画像の垂直方向の画素数として（ただし、ｒとｓは正の整数）、ｐの絶対値を１／２倍した値はｒより小さく、かつｐの絶対値はｒ以上で、かつｑの絶対値を１／２倍した値はｓより小さく、かつｑの絶対値はｓ以上であることを特徴とする請求項１ないし１６に記載のフレーム間予測画像の合成方法。

２１．　ｒを画像の水平方向の画素数、ｓを画像の垂直方向の画素数と

し（ただし、ｒとｓは正の整数）て、ｐの絶対値はｒ以下であり、かつｐの絶対値を２倍した値はｒより大きく、かつｑの絶対値はｓ以下であり、かつｑの絶対値を２倍した値はｓより大きいことを特徴とする請求項１ないし１６に記載のフレーム間予測画像の合成方法。

２２．　画像の水平方向と垂直方向の画素数がそれぞれｒとｓであり（ただし、ｒとｓは正の整数）、かつ画像の画素が水平座標が０以上ｒ未満、垂直座標が０以上ｓ未満の範囲に存在しているときに、

座標（－ｃ，－ｃ）、（ｒ－ｃ，－ｃ）、（－ｃ，ｓ－ｃ）、（ｒ－ｃ，ｓ－ｃ）に位置する画像の隅の点上に存在し（ただし、ｃ＝ｃｎ／ｃｄ、かつｃｎは負ではない整数、かつｃｄは正の整数、かつｃｎ＜ｃｄ）、水平・垂直成分が１／ｎの整数倍の値をとる動きベクトル（ただし、ｎは正の整数）をｎ倍したものである（ｕ００，ｖ００）、（ｕ０１，ｖ０１）、（ｕ０２，ｖ０２）、（ｕ０３，ｖ０３）（ただし、ｕ００、ｖ００、ｕ０１、ｖ０１、ｕ０２、ｖ０２、ｕ０３、ｖ０３は整数）を用いて、

ｕ’（ｘ，ｙ）＝（（（ｓ・ｃｄ－ｃｎ－ｙ・ｃｄ）（（ｒ・ｃｄ－ｃｎ－ｘ・ｃｄ）ｕ００＋（ｘ・ｃｄ＋ｃｎ）ｕ０１）＋（ｙ・ｃｄ＋ｃｎ）（（ｒ・ｃｄ－ｃｎ－ｘ・ｃｄ）ｕ０２＋（ｘ・ｃｄ＋ｃｎ）ｕ０３））ｋ）／／／（ｒ・ｓ・ｎ・ｃｄ），

ｖ’（ｘ，ｙ）＝（（（ｓ・ｃｄ－ｃｎ－ｙ・ｃｄ）（（ｒ・ｃｄ－ｃｎ－ｘ・ｃｄ）ｖ００＋（ｘ・ｃｄ＋ｃｎ）ｖ０１）＋（ｙ・ｃｄ＋ｃｎ）（（ｒ・ｃｄ－ｃｎ－ｘ・ｃｄ）ｖ０２＋（ｘ・ｃｄ＋ｃｎ）ｖ０３））ｋ）／／／（ｒ・ｓ・ｎ・ｃｄ・ｃｄ），

ｕ０＝ｕ’（ｉ，ｊ），

ｖ０＝ｖ’（ｉ，ｊ），

u1＝u’（i＋p，j），

v1＝v’（i＋p，j），

u2＝u’（i，j＋q），

v2＝v’（i，j＋q），

u3＝u’（i＋p，j＋q），

v3＝v’（i＋p，j＋q），

で表される（u0，v0）、（u1，v1）、（u2，v2）、（u3，v3）を（ただし、「///」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）、代表点（i，j）、（i＋p，j）、（i，j＋q）、（i＋p，j＋q）の動きベクトルの水平・垂直成分をk倍したものとして使用することを特徴とする請求項1ないし21に記載のフレーム間予測画像の合成方法。

23．　符号化しようとする現フレームの画像信号とフレーム間予測画像との差を誤差画像としてを出力する第1ステップと

上記誤差画像を信号変換し、変換信号を得てそれを符号化する第2ステップと

上記変換信号を逆変換して上記誤差画像の復号誤差画像を作る第4ステップと

上記復号誤差画像と上記フレーム間予測画像を用いて上記現フレームの画像信号の次の現フレームの画像に対応するフレーム間予測画像信号作る第5ステップとを有し、

上記第5ステップが請求項1ないし16のいずれかのフレーム間予測画像の合成方法で行うことを特徴とする画像符号化方法。

２４．　上記第５ステップが代表点の動きベクトルに関する情報を検出し符号化するステップを含む請求項２３記載の画像符号化方法。

２５．　上記第５ステップが代表点が画像の隅の点である請求項２３に記載の画像の符号化方法。

２６．　復号すべき画像フレームのフレーム間符号化信号と上記画像フレームの動きベクトルの情報と入力する第１ステップと、

上記フレーム間符号化信号を復号誤差信号に変換する第２ステップと、

上記復号すべき画像フレームと時間的に異なる他の画像フレームの復号画像信号と上記動きベクトルの情報からフレーム間予測画像を作る第３ステップと、

復号誤差信号と上記フレーム間予測画像の信号とを加算して、上記復号すべき画像フレームの復号画像信号を得る第４ステップともち、

上記第３ステップが請求項１ないし１６のいずれかのフレーム間予測画像の合成方法で行う画像復号化方法。

２７．　上記複数の代表点が符号化データとして直接符号化されている代表点の動きベクトルに関する情報を再生して用いる上記画像の隅の点である請求項２６に記載の画像の復号化方法。

２８．　上記代表点が上記画像の隅の点である請求項２６に記載の画像の復号化方法。

２９．　符号化しようとする現フレームの画像信号とフレーム間予測像と像を信号変換する第１変換部の出力の一部を符号化する符号化器と、

第１変換部の出力の一部を逆変換して上記誤差画像の復号誤差画像

を得るる第2変換部と、

上記復号誤差画像と上記フレーム間予測画像とから上記現フレームの画像信号の復号画像信号を得る復号手段と、

上記前フレームの復号画像と現フレームの入力画像が入力され前述のフレーム間予測画像の合成を行う動き補償処理部をもつ符号化装置であって、

動き補償処理部が上記前フレームの復号画像と現フレームの入力画像から上記前フレームの復号画像の座標（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）に（ｉ、ｊ、ｐ、ｑは整数）を持つ４つの代表点の動きベクトル（動きベクトルの水平・垂直成分が１／ｋの整数倍の値をとり（ただし、ｋは2のｈｋ乗、かつｈｋは負ではない整数）を求めるグローバル動きベクトル推定部と、

上記動きベクトルと上記前フレームの復号画像から上記符号化しようとする現フレームの画像の信号を予測したフレーム間予測画像を作る予測画像合成部と持ち、

上記予測画像合成部が、画素のサンプリング間隔を水平、垂直方向共に１として、サンプリング点が座標の水平、垂直成分が、共に整数にｗを加えた数である点の上に存在している画像を対象として（ただし、ｗ＝ｗｎ／ｗｄ、かつｗｎは負ではない整数、かつｗｄは２のｈｗ乗、かつｈｗは負ではない整数、かつｗｎ＜ｗｄ）、座標（ｉ，ｊ）と（ｉ，ｊ＋ｑ）に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標（ｉ，ｙ＋ｗ）に位置する点の動きベクトルの水平・垂直成分をそれぞれ１／ｚの整数倍をとる数値として（ただし、ｚは2のｈｚ乗、かつｈｚは負ではない整数）求め、 さらに座標（ｉ＋

ｐ，ｊ）と（ｉ＋ｐ，ｊ＋ｑ）に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標（ｉ＋ｐ，ｙ＋ｗ）に位置する点の動きベクトルの水平・垂直成分をそれぞれ１／ｚの整数倍をとる数値として求めた後に、（ｉ，ｙ＋ｗ）と（ｉ＋ｐ，ｙ＋ｗ）に位置する上記２個の動きベクトルに対して線形内・外挿を行うことにより、座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルの水平・垂直成分をそれぞれ１／ｍの整数倍をとる数値として（ただし、ｍは2のｈｍ乗、かつｈｍは負ではない整数）求める演算部と、

上記座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルと上記前フレームの復号画像から予測画像を合成する合成部とをもつ画像符号化装置。

３０．　符号化しようとする現フレームの画像信号とフレーム間予測画像信号との差を誤差画像としてを出力する減算器と、

上記誤差画像を信号変換する第１変換部の出力の一部を符号化する符号化器と、

第１変換部の出力の一部を逆変換して上記誤差画像の復号誤差画像を得るる第2変換部と、

上記復号誤差画像と上記フレーム間予測画像とから上記現フレームの画像信号の復号画像信号を得る復号手段と、

上記前フレームの復号画像と現フレームの入力画像６０１が入力され上記フレーム間予測画像の合成を行う動き補償処理部をもつ符号化装置であって、

動き補償処理部が上記前フレームの復号画像と現フレームの入力画像から上記前フレームの復号画像の座標（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、

（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）に（ｉ、ｊ、ｐ、ｑは整数）を持つ４つの代表点の動きベクトル（動きベクトルの水平・垂直成分が１／ｋの整数倍の値をとり（ただし、ｋは2のｈｋ乗、かつｈｋは負ではない整数）を求めるグローバル動きベクトル推定部と、

上記動きベクトルと上記前フレームの復号画像から上記符号化しようとする現フレームの画像の信号を予測したフレーム間予測画像を作る予測画像合成部と持ち、

上記予測画像合成部が、画素のサンプリング間隔を水平、垂直方向共に１として、サンプリング点が座標の水平、垂直成分が、共に整数にｗを加えた数である点の上に存在している画像を対象として（ただし、ｗ＝ｗｎ／ｗｄ、かつｗｎは負ではない整数、かつｗｄは２のｈｗ乗、かつｈｗは負ではない整数、かつｗｎ＜ｗｄ）、座標（ｉ，ｊ）と（ｉ＋ｐ，ｊ）に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標（ｘ＋ｗ，ｊ）に位置する点の動きベクトルの水平・垂直成分をそれぞれ１／ｚの整数倍をとる数値として（ただし、ｚは2のｈｚ乗、かつｈｚは負ではない整数）求め、

さらに座標（ｉ，ｊ＋ｑ）と（ｉ＋ｐ，ｊ＋ｑ）に位置する代表点の動きベクトルに対して線形内・外挿を行うことにより、座標（ｘ＋ｗ，ｊ＋ｑ）に位置する点の動きベクトルの水平・垂直成分をそれぞれ１／ｚの整数倍をとる数値として求めた後に、

（ｘ＋ｗ，ｊ）と（ｘ＋ｗ，ｊ＋ｐ）に位置する上記２個の動きベクトルに対して線形内・外挿を行うことによ、座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルの水平・垂直成分をそれぞれ１／ｍの整数倍をとる数値として（ただし、ｍは2のｈｍ乗、かつｈｍは負ではな

い整数）求める演算部と、

上記（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルと上記前フレームの復号画像から予測画像を合成する合成部とをもつ画像符号化装置。

３１．　座標（ｉ，ｊ）、（ｉ＋ｐ，ｊ）、（ｉ，ｊ＋ｑ）、（ｉ＋ｐ，ｊ＋ｑ）に位置する代表点の動きベクトルの水平・垂直成分をｋ倍したものである（ｕ0，ｖ0）、（ｕ1，ｖ1）、（ｕ2，ｖ2）、（ｕ3，ｖ3）を用いて座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルを求めるときに、座標（ｉ，ｙ＋ｗ）に位置する点の動きベクトルの水平・垂直成分をそれぞれｚ倍したものである（ｕL(ｙ＋ｗ)，ｖL(ｙ＋ｗ)）を、

ｕL(ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｕ0＋（ｙ・ｗｄ＋ｗｎ）ｕ2）ｚ）////（ｑ・ｋ・ｗｄ），

ｖL(ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｖ0＋（ｙ・ｗｄ＋ｗｎ）ｖ2）ｚ）////（ｑ・ｋ・ｗｄ）

を計算することにより（ただし、「////」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求め、

更に座標（ｉ＋ｐ，ｙ＋ｗ）に位置する点の動きベクトルの水平・垂直成分をそれぞれｚ倍したものである（ｕR(ｙ＋ｗ)，ｖR(ｙ＋ｗ)）を、

ｕR(ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｕ1＋（ｙ・ｗｄ＋ｗｎ）ｕ3）ｚ）////（ｑ・ｋ・ｗｄ），

ｖR(ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｖ1＋（ｙ・ｗｄ＋ｗｎ）

v3) z) //// (q・k・wd)、

を計算することにより求めた後に、座標（x＋w，y＋w）に位置する画素の動きベクトルの水平・垂直成分をそれぞれm倍したものである（u(x＋w，y＋w)，v(x＋w，y＋w)）を

u(x＋w，y＋w)＝(((p・wd－x・wd－wn) uL(y＋w)＋(x・wd＋wn) uR(y＋w)) m) // (p・z・wd)

v(x＋w，y＋w)＝(((p・wd－x・wd－wn) vL(y＋w)＋(x・wd＋wn) vR(y＋w)) m) // (p・z・wd)

を計算することによって（ただし、「//」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求めることを特徴とする請求項29に記載の符号化装置。

32．　座標（i，j）、（i＋p，j）、（i，j＋q）、（i＋p，j＋q）に位置する代表点の動きベクトルの水平・垂直成分をk倍したものである（u0，v0）、（u1，v1）、（u2，v2）、（u3，v3）を用いて座標（x＋w，y＋w）に位置する画素の動きベクトルを求めるときに、

座標（x＋w，j）に位置する点の動きベクトルの水平・垂直成分をそれぞれz倍したものである（uT(x＋w)，vT(x＋w)）を、

uT(x＋w)＝(((p・wd－x・wd－wn) u0＋(x・wd＋wn) u1) z) //// (p・k・wd)、

vT(x＋w)＝(((p・wd－x・wd－wn) v0＋(x・wd＋wn) v1) z) //// (p・k・wd)、

を計算することにより（ただし、「////」は通常の除算による演算結

果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求め、

さらに座標（ｘ＋ｗ，ｊ＋ｐ）に位置する点の動きベクトルの水平・垂直成分をそれぞれｚ倍したものである（ｕＢ(ｙ＋ｗ)，ｖＢ(ｙ＋ｗ)）を、

ｕＢ(ｘ＋ｗ)＝（（（ｐ・ｗｄ－ｘ・ｗｄ－ｗｎ）ｕ2＋（ｘ・ｗｄ＋ｗｎ）ｕ3）ｚ）////（ｐ・ｋ・ｗｄ），

ｖＢ(ｘ＋ｗ)＝（（（ｐ・ｗｄ－ｘ・ｗｄ－ｗｎ）ｖ2＋（ｘ・ｗｄ＋ｗｎ）ｖ3）ｚ）////（ｐ・ｋ・ｗｄ），

を計算することにより求めた後に、座標（ｘ＋ｗ，ｙ＋ｗ）に位置する画素の動きベクトルの水平・垂直成分をそれぞれｍ倍したものである（ｕ(ｘ＋ｗ，ｙ＋ｗ)，ｖ(ｘ＋ｗ，ｙ＋ｗ)）を

ｕ(ｘ＋ｗ，ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｕＴ(ｘ＋ｗ)＋（ｙ・ｗｄ＋ｗｎ）ｕＢ(ｘ＋ｗ)）ｍ）//（ｑ・ｚ・ｗｄ），

ｖ(ｘ＋ｗ，ｙ＋ｗ)＝（（（ｑ・ｗｄ－ｙ・ｗｄ－ｗｎ）ｖＴ(ｘ＋ｗ)＋（ｙ・ｗｄ＋ｗｎ）ｖＢ(ｘ＋ｗ)）ｍ）//（ｑ・ｚ・ｗｄ），

を計算することによって（ただし、「//」は通常の除算による演算結果が整数ではない場合にこれを近隣の整数に丸め込む除算で、演算子としての優先順位は乗除算と同等）求めること請求項３０に記載のフレーム間予測画像の符号化装置。

３３．　ｐの絶対値が２のα乗（αは負ではない整数）であるとする請求項２９に記載のフレーム間予測画像の符号化装置。

３４．　ｑの絶対値が２のβ乗（βは負ではない整数）である請求項３０に記載のフレーム間予測画像の符号化装置。

３５．　ｐとｑの絶対値がそれぞれ２のα乗とβ乗（α、βは負ではない整数）である請求項２９に記載の符号化装置。

３６．　ｐとｑの絶対値がそれぞれ２のα乗とβ乗（α、βは負ではない整数）である請求項３０に記載の符号化装置。

３７．　α＋ｈｚが８の正の整数倍であり、かつｗが０である請求項３３に記載の符号化装置。

３８．　β＋ｈｚが８の正の整数倍であり、かつｗが０である請求項３４に記載の符号化装置。

３９．　α＋ｈｚ＋ｈｗが８の正の整数倍であり、かつｗ＞０である請求項３３記載の符号化装置。

４０．　β＋ｈｚ＋ｈｗが８の正の整数倍であり、かつｗ＞０である請求項３４に記載の符号化装置。

４１．　複数の異なるαの値に対応し、α＋ｈｚが１６以下となるようにｈｚの値をαの値に応じて変化する請求項３７に記載の符号化装置。

４２．　複数の異なるβの値に対応し、β＋ｈｚが１６以下となるようにｈｚの値をβの値に応じて変化する請求項３８に記載の符号化装置。

４３．　複数の異なるαの値に対応し、α＋ｈｚ＋ｈｗが１６以下となるようにｈｚの値をαの値に応じて変化する請求項３９に記載の符号化装置。

４４．　複数の異なるβの値に対応し、β＋ｈｚ＋ｈｗが１６以下となるようにｈｚの値をβの値に応じて変化する請求項４０に記載の符号化装置。

４５．　上記動き補償処理部が更に上記代表点の動きベクトルに関する情報を符号化する手段をもつ請求項２９ないし４０のいずれか記載

の符号化装置。

４６．　上記代表点が画像の隅の位置の点である請求項２９ないし４０のいずれかに記載の符号化装置。

４７．　上記第１変換部及び第２変換部がそれぞれ上記誤差誤差画像の信号をＤＣＴ変換して量子化する回路及び逆量子化して逆ＤＣＴ変換する回路である請求項２９ないし４０のいずれか記載の符号化装置。

４８．符号化された画像信号のフレーム間誤差の符号を誤差画像の信号に変換する変換回路と、復号されたフレーム画像信号を記憶するフレームメモリと、上記符号化された画像信号の動きベクトルと上記フレームメモリの復号されたフレーム画像信号を入力し、予測画像を合成する予測画像合成部と、上記予測画像合成部の出力と上記変換回路の出力を加算して復号画像を作る加算部と、上記加算部の出力を上記フレームメモリに記録する手段を持つ画像復号化装置において、上記予測画像合成部が請求項1ないし16のいずれかに記載された記載されたフレーム間予測画像の合成方法を行う手段で構成された画像復号化装置。

４９．　請求項１ないし２２に記載のフレーム間予測画像の合成方法を実行するためのソフトウェアを記録した蓄積メディア。

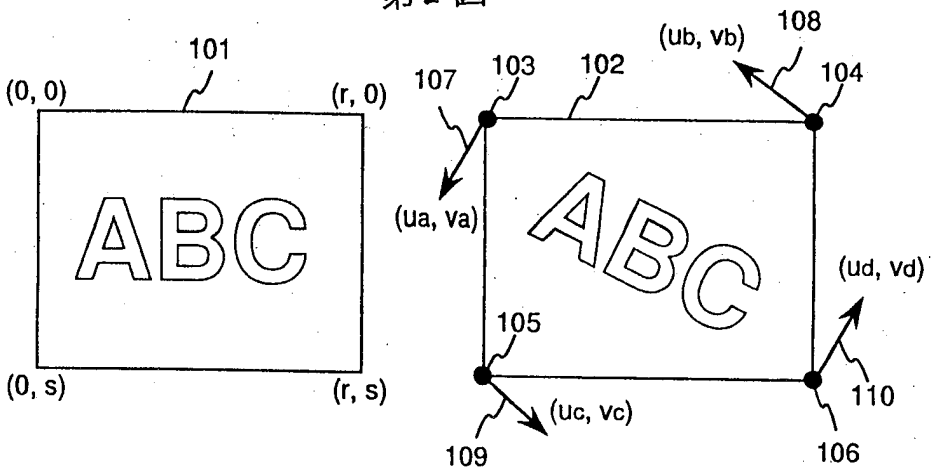
５０．　請求項４８に記載の画像復号化装置を駆動するするためのソフトウェアを記録した蓄積メディア。

５１．　請求項２３、２４、又は２５に記載の符号化方法によっ生成された符号化されたビットストリームを記録した蓄積メディア。

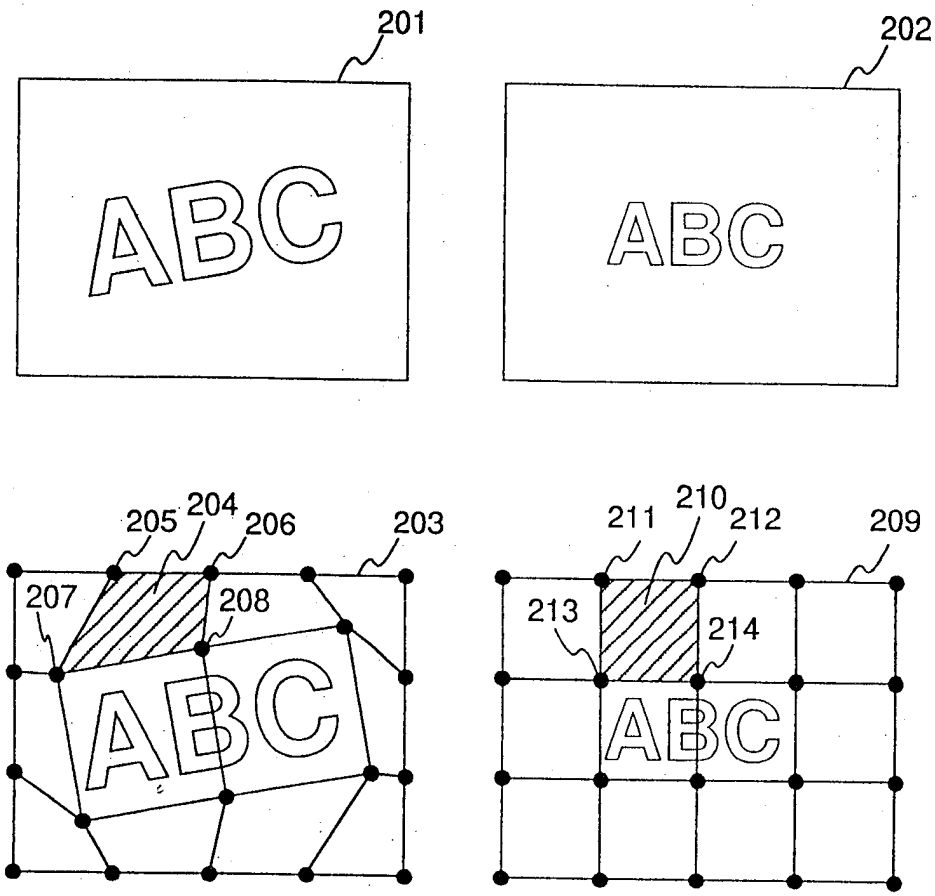
５２．請求項２６、２７又は２８に記載の画像復号化方法によって復号化することができる符号化された圧縮ビットストリームを記録した

蓄積メディア。

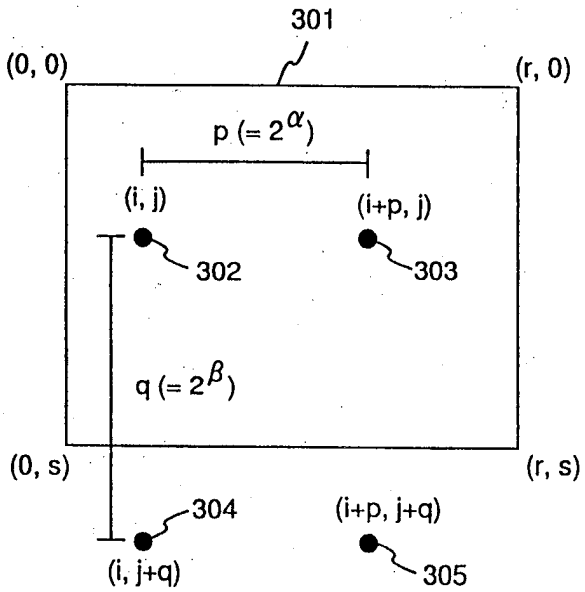
第 1 図


101
(0, 0)
(r, 0)
ABC
(0, s)
(r, s)
107
103
102
(ub, vb)
108
104
(ua, va)
ABC
(ud, vd)
105
110
(uc, vc)
109
106

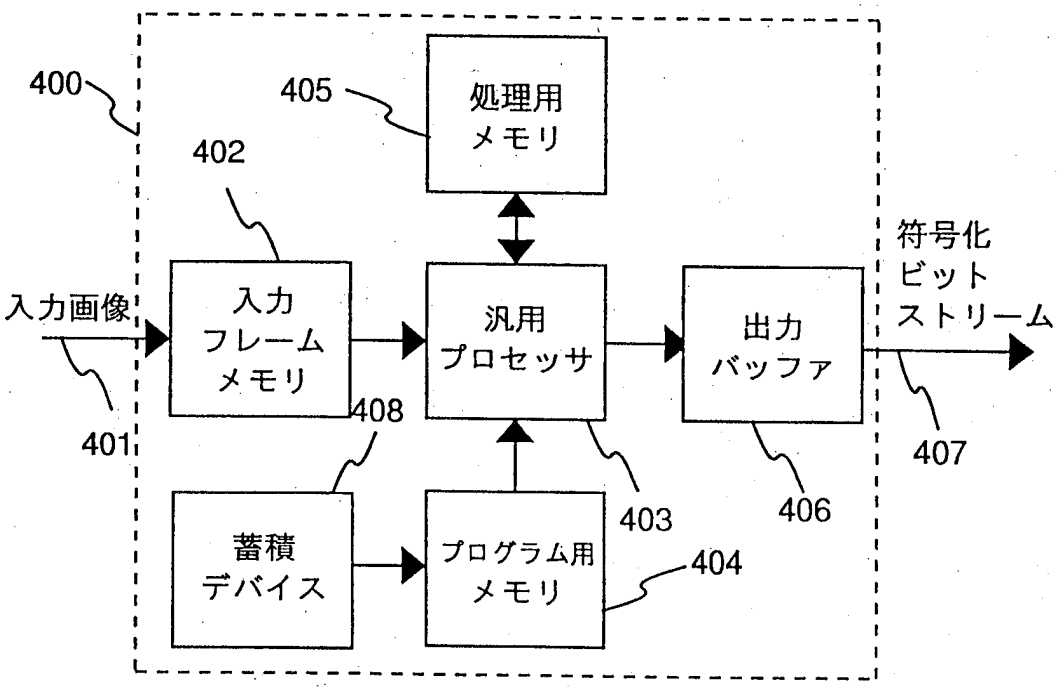
第 2 図


201
202
ABC
ABC
205
204
206
203
207
208
ABC
211
210
212
209
213
214
ABC

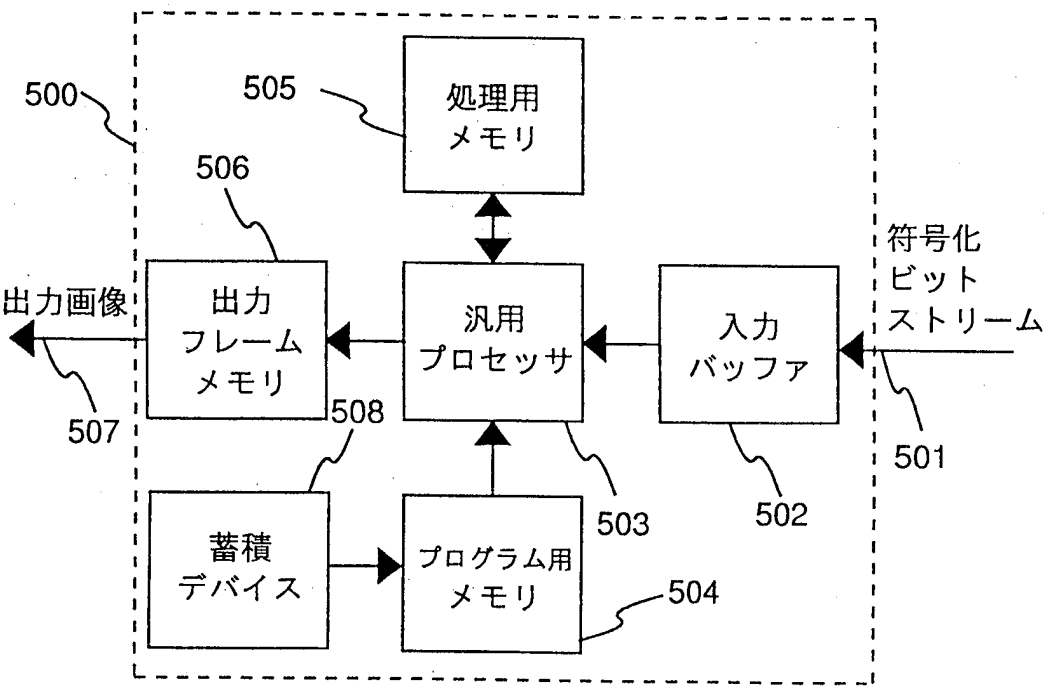
第 3 図



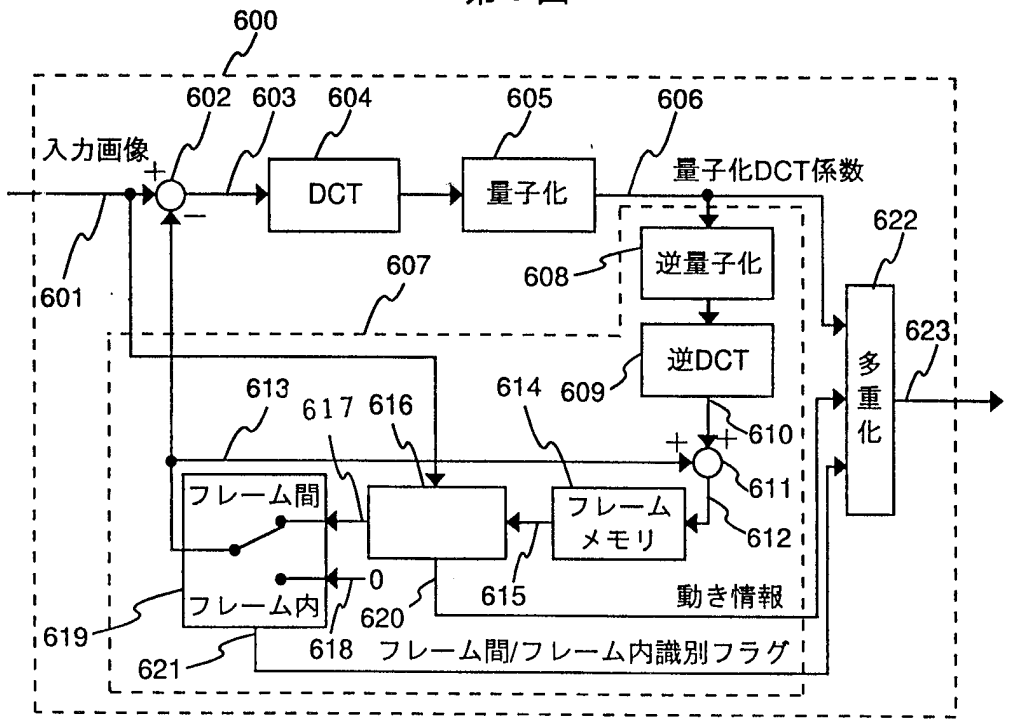
第 4 図


400
405
処理用
メモリ
402
入力画像
入力
フレーム
メモリ
汎用
プロセッサ
出力
バッファ
符号化
ビット
ストリーム
401
408
403
406
407
蓄積
デバイス
プログラム用
メモリ
404

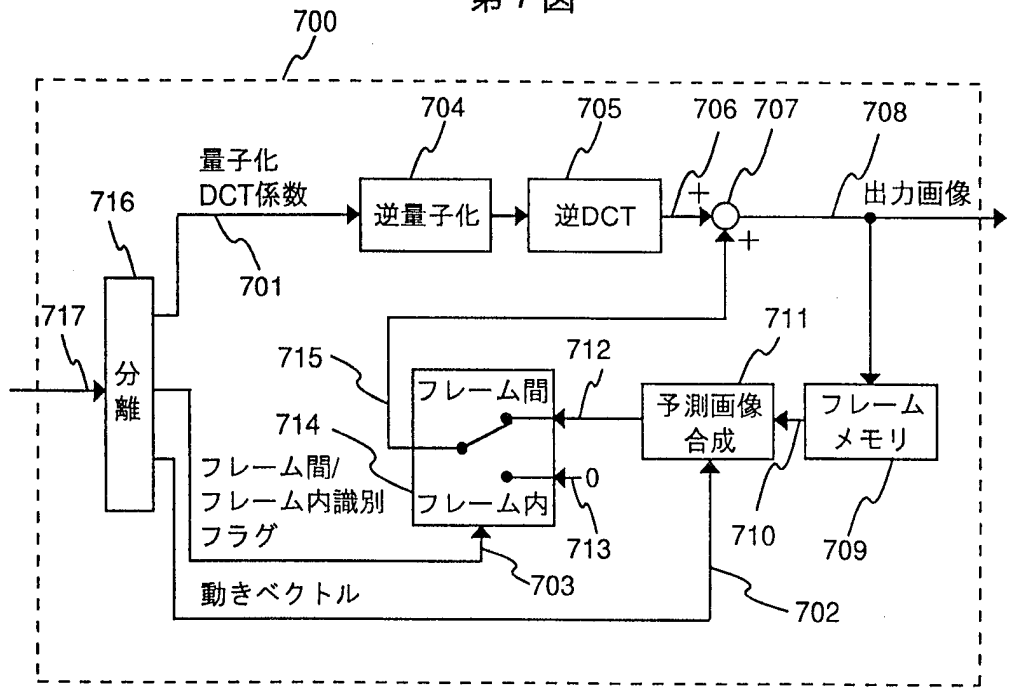
第 5 図


500
505
処理用
メモリ
506
出力画像
出力
フレーム
メモリ
汎用
プロセッサ
入力
バッファ
符号化
ビット
ストリーム
507
508
503
502
501
蓄積
デバイス
プログラム用
メモリ
504

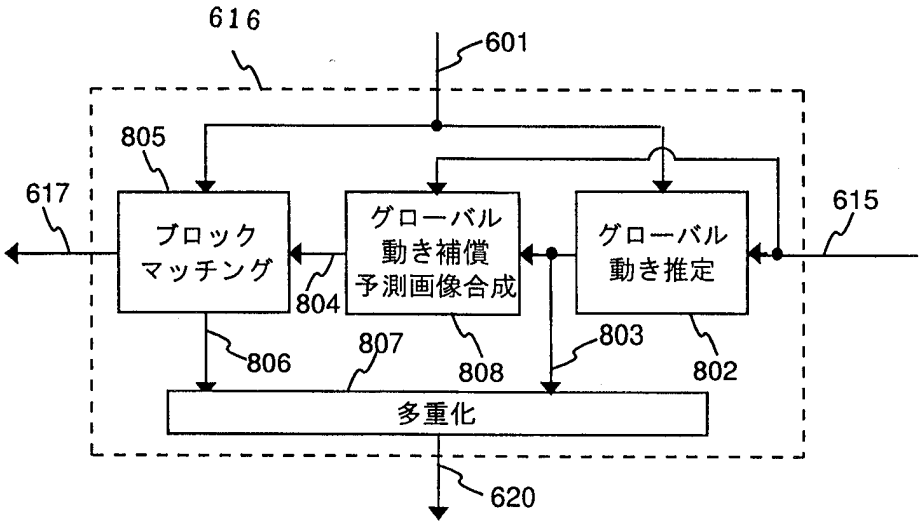
第6図


600
602
603
604
605
606
入力画像
DCT
量子化
量子化DCT係数
601
607
608
逆量子化
622
逆DCT
623
多重化
613
617
616
614
609
610
611
612
フレーム間
フレームメモリ
0
フレーム内
615
動き情報
619
620
621
618
フレーム間/フレーム内識別フラグ

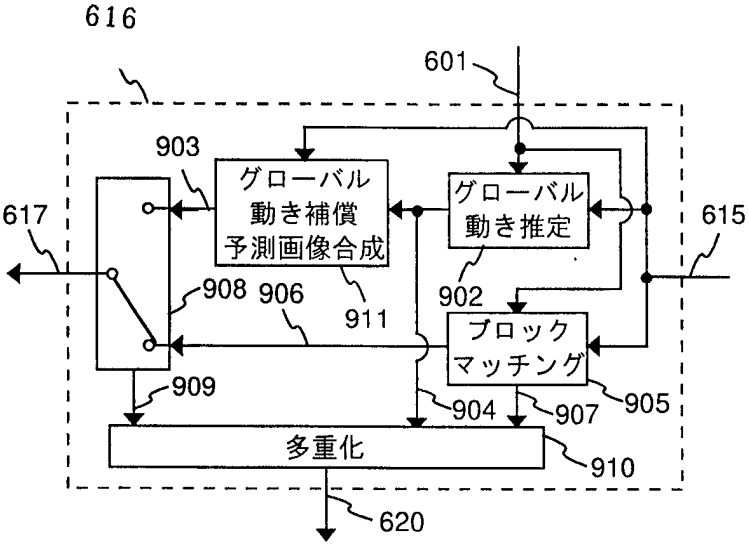
第7図


700
704
705
706
707
708
量子化
DCT係数
逆量子化
逆DCT
出力画像
716
701
717
分離
711
715
712
フレーム間
予測画像合成
フレームメモリ
714
フレーム間/
フレーム内識別
フラグ
フレーム内
0
713
710
709
703
動きベクトル
702

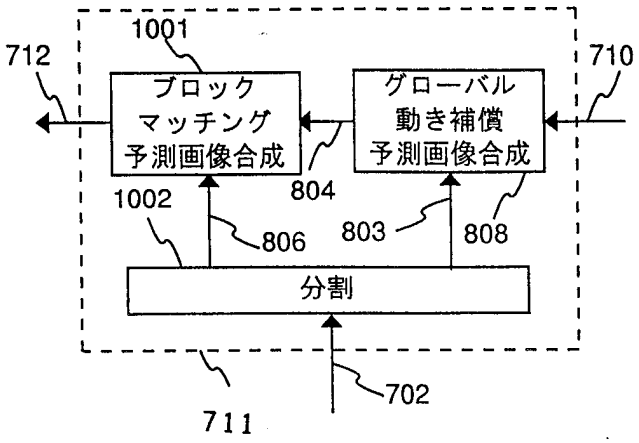
第８図


616
601
805
617
615
ブロック
マッチング
グローバル
動き補償
予測画像合成
グローバル
動き推定
804
806
807
808
803
802
多重化
620

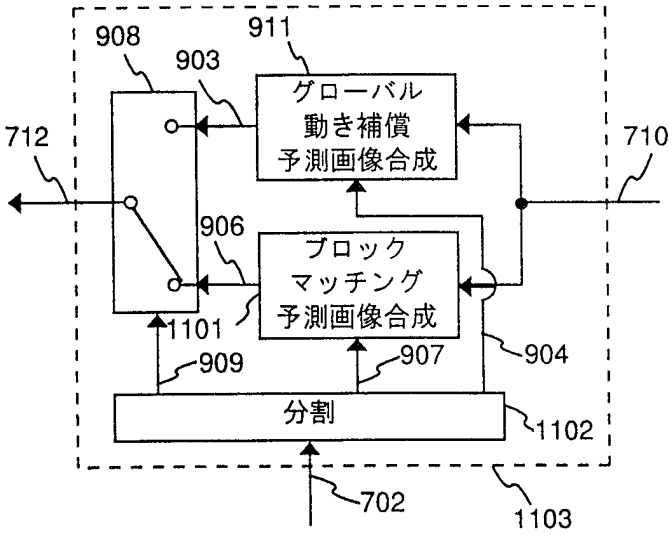
第９図


616
601
903
617
615
グローバル
動き補償
予測画像合成
グローバル
動き推定
908
906
911
902
ブロック
マッチング
909
904
907
905
多重化
910
620

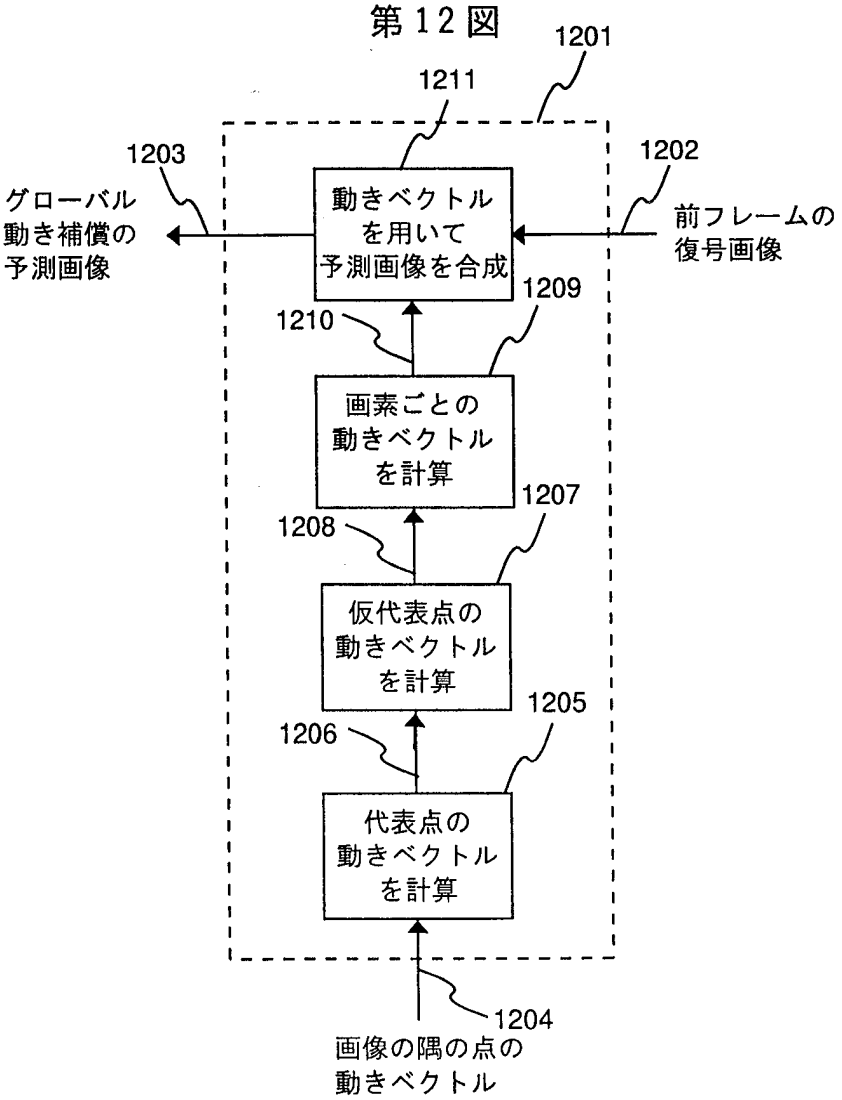
第１０図


1001
712
710
ブロック
マッチング
予測画像合成
グローバル
動き補償
予測画像合成
804
1002
806
803
808
分割
702
711

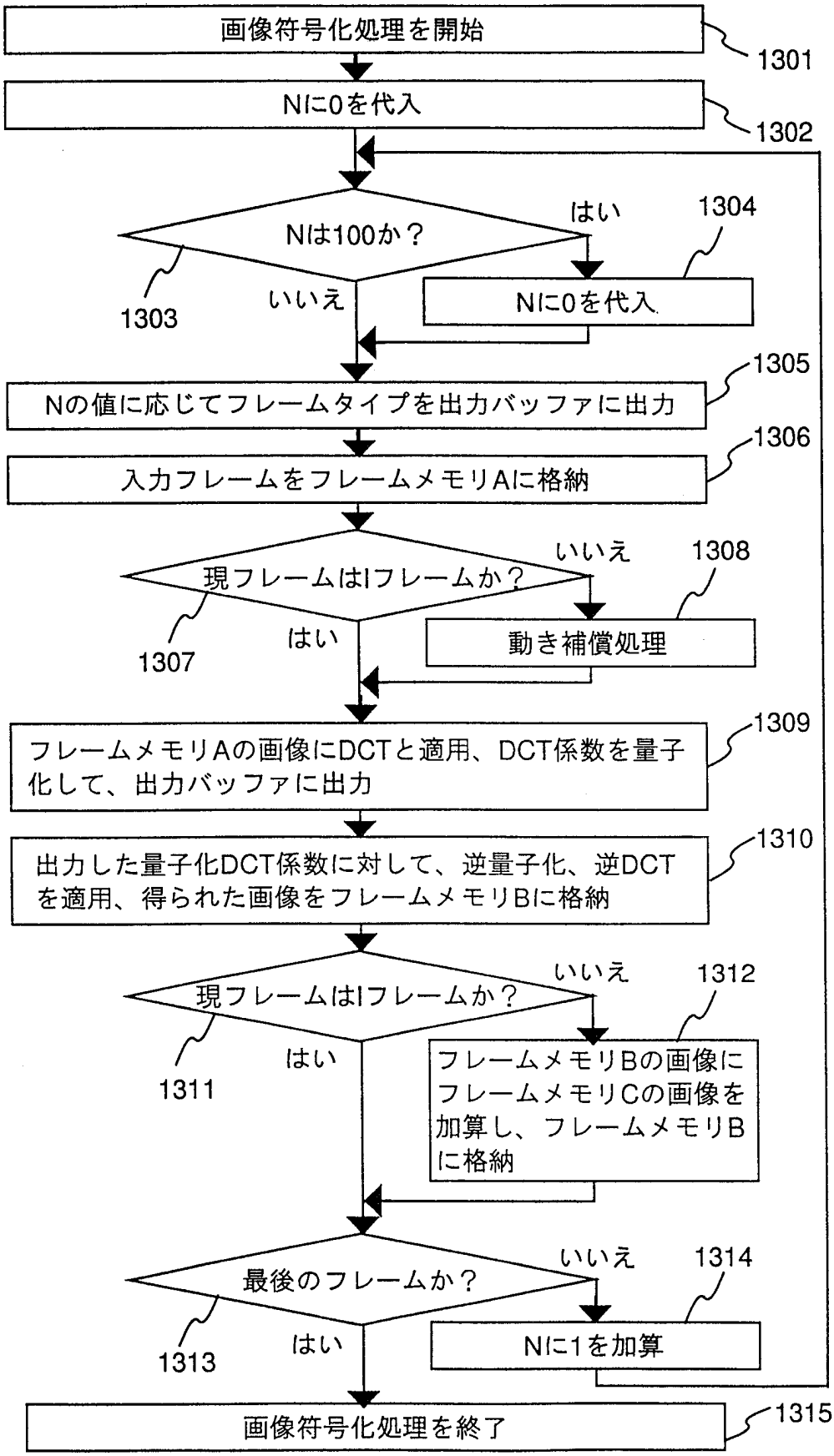
第１１図


908
911
903
グローバル
動き補償
予測画像合成
712
710
906
ブロック
マッチング
予測画像合成
1101
909
907
904
分割
1102
702
1103

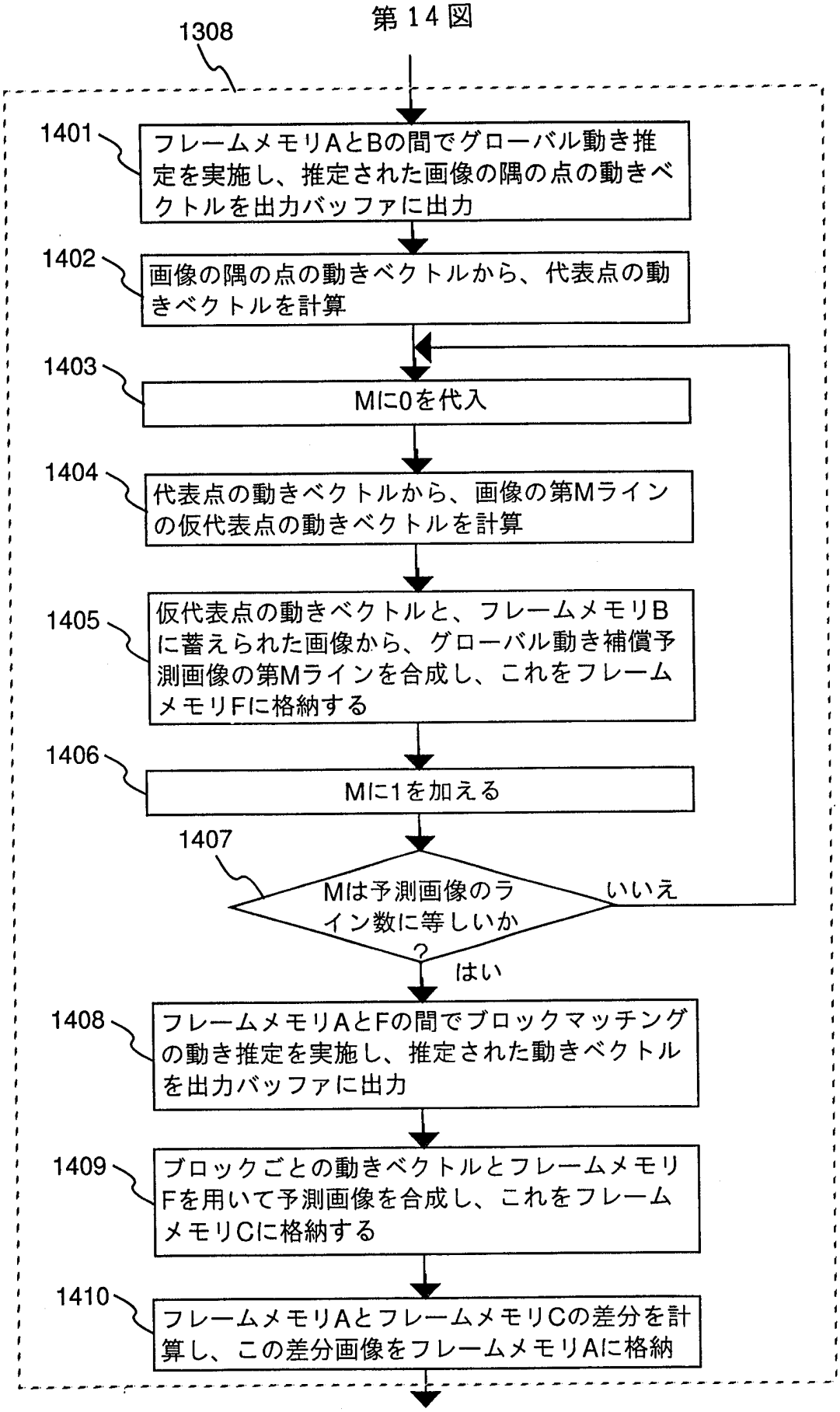
第12図


1201
1211
1203
1202
グローバル
動き補償の
予測画像
動きベクトル
を用いて
予測画像を合成
前フレームの
復号画像
1209
1210
画素ごとの
動きベクトル
を計算
1207
1208
仮代表点の
動きベクトル
を計算
1205
1206
代表点の
動きベクトル
を計算
1204
画像の隅の点の
動きベクトル

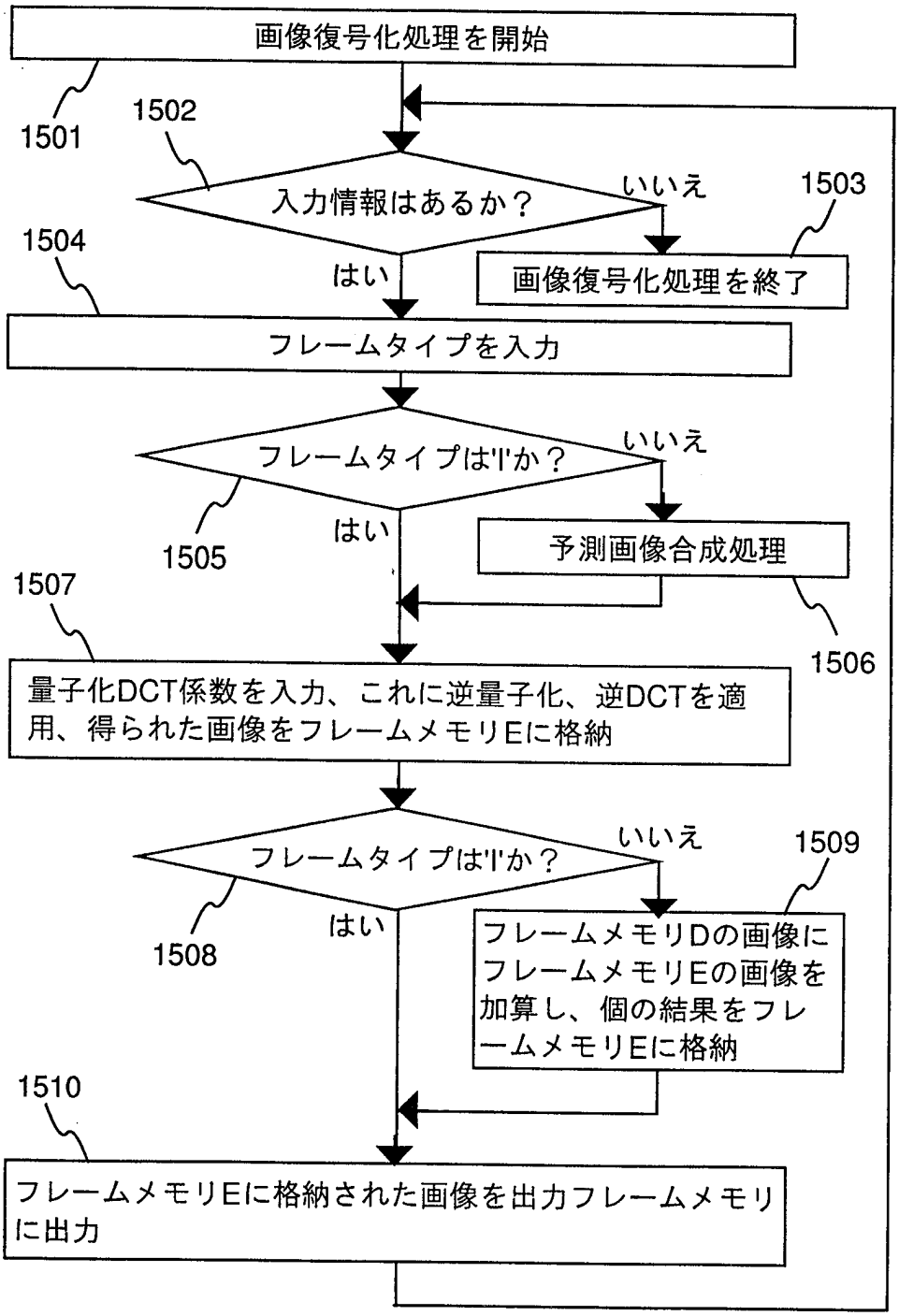
第13図


画像符号化処理を開始
1301
Nに0を代入
1302
Nは100か？
はい
1304
いいえ
1303
Nに0を代入
1305
Nの値に応じてフレームタイプを出力バッファに出力
1306
入力フレームをフレームメモリAに格納
現フレームはIフレームか？
いいえ
1308
はい
1307
動き補償処理
1309
フレームメモリAの画像にDCTと適用、DCT係数を量子化して、出力バッファに出力
1310
出力した量子化DCT係数に対して、逆量子化、逆DCTを適用、得られた画像をフレームメモリBに格納
現フレームはIフレームか？
いいえ
1312
はい
1311
フレームメモリBの画像にフレームメモリCの画像を加算し、フレームメモリBに格納
最後のフレームか？
いいえ
1314
はい
1313
Nに1を加算
画像符号化処理を終了
1315

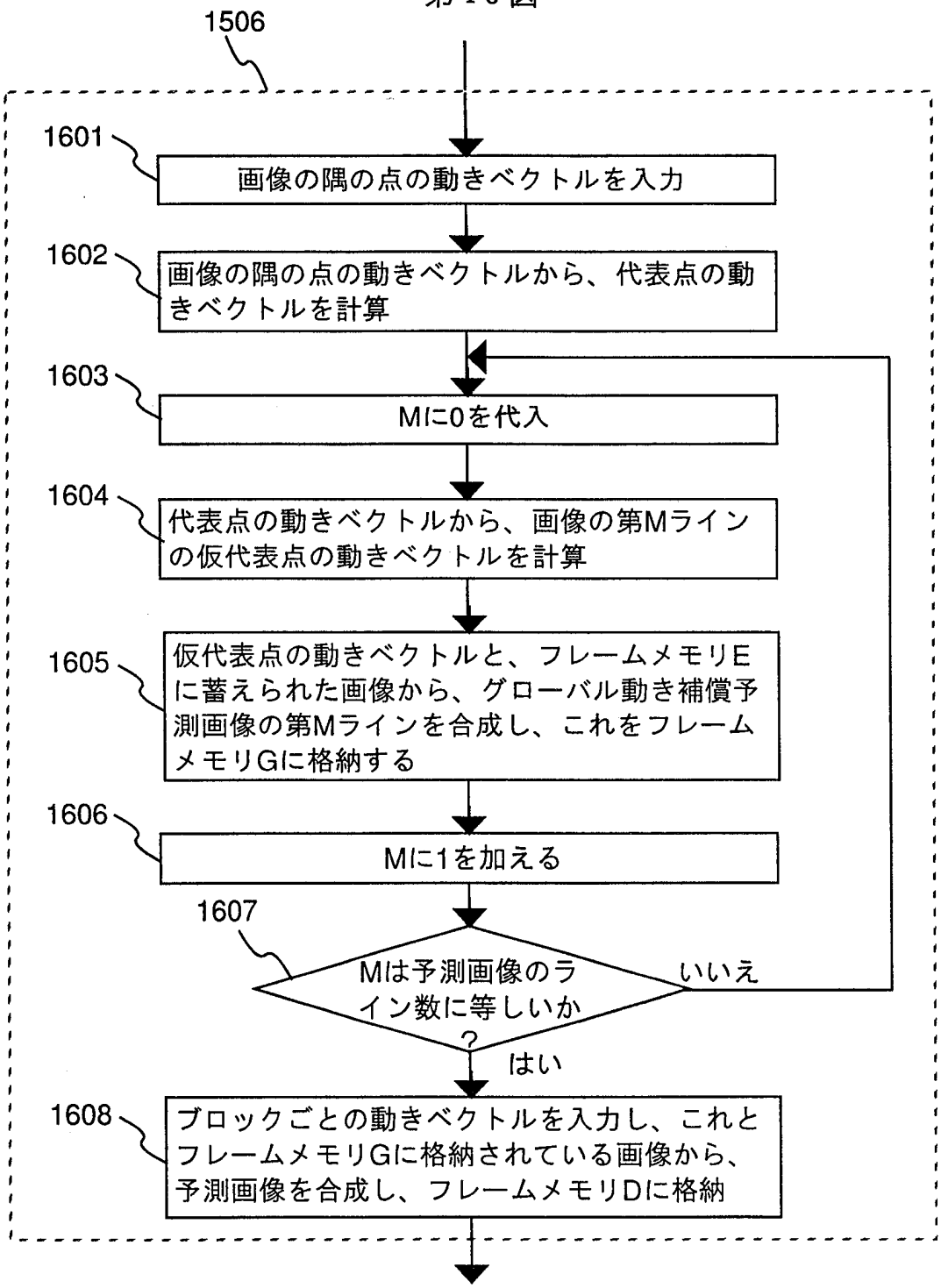
第14図


1308
1401
フレームメモリAとBの間でグローバル動き推定を実施し、推定された画像の隅の点の動きベクトルを出力バッファに出力
1402
画像の隅の点の動きベクトルから、代表点の動きベクトルを計算
1403
Mに0を代入
1404
代表点の動きベクトルから、画像の第Mラインの仮代表点の動きベクトルを計算
1405
仮代表点の動きベクトルと、フレームメモリBに蓄えられた画像から、グローバル動き補償予測画像の第Mラインを合成し、これをフレームメモリFに格納する
1406
Mに1を加える
1407
Mは予測画像のライン数に等しいか？
いいえ
はい
1408
フレームメモリAとFの間でブロックマッチングの動き推定を実施し、推定された動きベクトルを出力バッファに出力
1409
ブロックごとの動きベクトルとフレームメモリFを用いて予測画像を合成し、これをフレームメモリCに格納する
1410
フレームメモリAとフレームメモリCの差分を計算し、この差分画像をフレームメモリAに格納

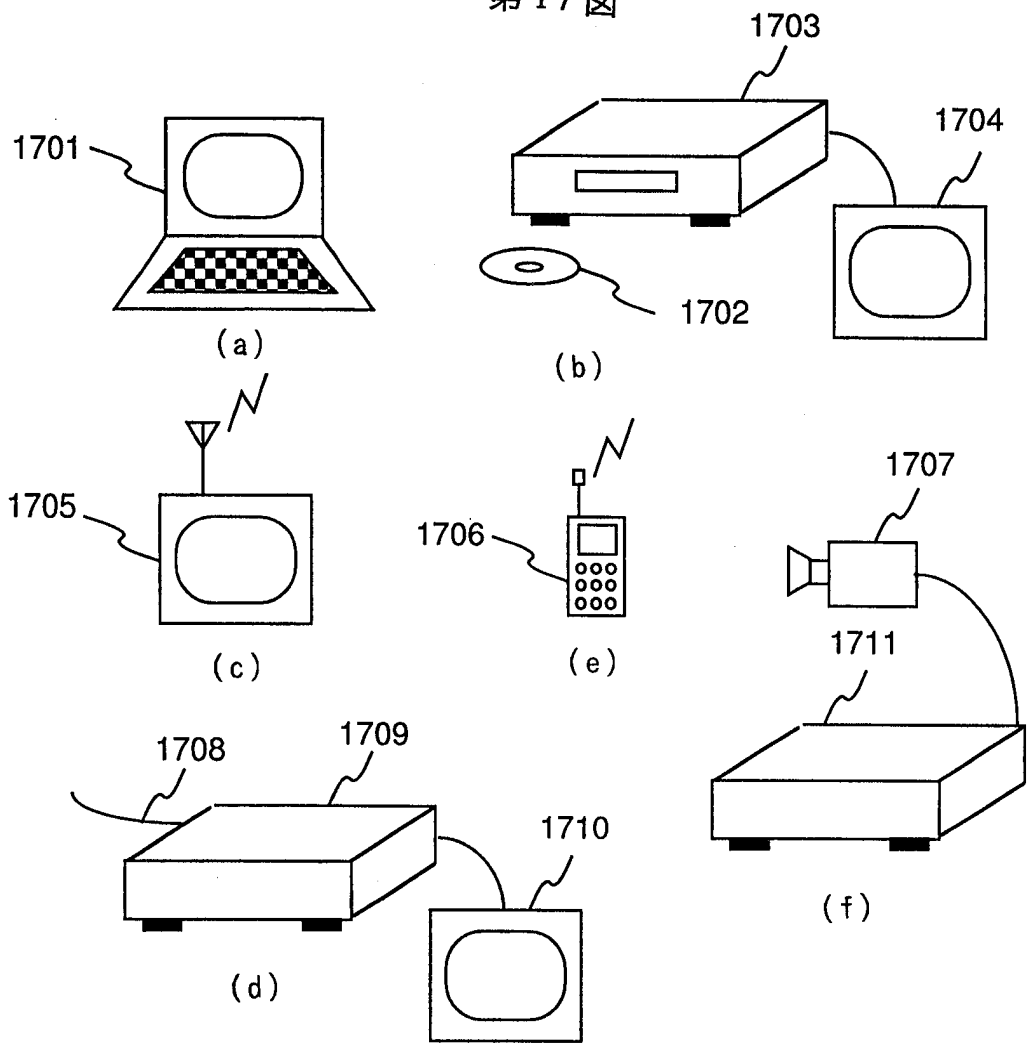
第15図


画像復号化処理を開始
1501
1502
入力情報はあるか？
いいえ
1503
画像復号化処理を終了
1504
はい
フレームタイプを入力
フレームタイプは'I'か？
いいえ
1505
はい
予測画像合成処理
1506
1507
量子化DCT係数を入力、これに逆量子化、逆DCTを適用、得られた画像をフレームメモリEに格納
フレームタイプは'I'か？
いいえ
1509
1508
はい
フレームメモリDの画像にフレームメモリEの画像を加算し、個の結果をフレームメモリEに格納
1510
フレームメモリEに格納された画像を出力フレームメモリに出力

第16図


1506
1601
画像の隅の点の動きベクトルを入力
1602
画像の隅の点の動きベクトルから、代表点の動きベクトルを計算
1603
Mに0を代入
1604
代表点の動きベクトルから、画像の第Mラインの仮代表点の動きベクトルを計算
1605
仮代表点の動きベクトルと、フレームメモリEに蓄えられた画像から、グローバル動き補償予測画像の第Mラインを合成し、これをフレームメモリGに格納する
1606
Mに1を加える
1607
Mは予測画像のライン数に等しいか？
いいえ
はい
1608
ブロックごとの動きベクトルを入力し、これとフレームメモリGに格納されている画像から、予測画像を合成し、フレームメモリDに格納

第17図


1701
(a)
1703
1704
1702
(b)
1705
(c)
1706
(e)
1707
1711
(f)
1708
1709
1710
(d)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP98/02435

A. CLASSIFICATION OF SUBJECT MATTER
Int.Cl^6 H04N7/32

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
Int.Cl^6 H04N7/24-7/68

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Jitsuyo Shinan Koho 1957-1998
Kokai Jitsuyo Shinan Koho 1975-1998

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	C. Haung, et al., "A New Motion Compensation Method for Image Sequence Coding Using Hierarchical Grid Interpolation" IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, Vol. 4, No. 1 (1994), p.42-52	1-52
A	Seiji Kimura et al., "Motion Compensation Based on Variable Size Variable Shape Block (in Japanese)", Technical Report of the Institute of Television Engineers of Japan, Vol. 20, No. 64 (1996) (Tokyo) p.31-38	1-52

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier document but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
25 August, 1998 (25. 08. 98)	8 September, 1998 (08. 09. 98)
Name and mailing address of the ISA/ Japanese Patent Office	Authorized officer
Facsimile No.	Telephone No.

Form PCT/ISA/210 (second sheet) (July 1992)

国際調査報告

国際出願番号　ＰＣＴ／ＪＰ９８／０２４３５

A.　発明の属する分野の分類（国際特許分類（ＩＰＣ））

Ｉｎｔ．Ｃｌ[6]　Ｈ０４Ｎ７／３２

B.　調査を行った分野

調査を行った最小限資料（国際特許分類（ＩＰＣ））

Ｉｎｔ．Ｃｌ[6]　Ｈ０４Ｎ７／２４－７／６８

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報　　１９５７－１９９８年
日本国公開実用新案公報　１９７５－１９９８年

国際調査で使用した電子データベース（データベースの名称、調査に使用した用語）

C.　関連すると認められる文献

引用文献のカテゴリー＊	引用文献名　及び一部の箇所が関連するときは、その関連する箇所の表示	関連する請求の範囲の番号
A	C.Haung,et al「A New Motion Compensation Method for Image Sequence Coding Using Hierarchical Grid Interpolation」IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY,第４巻，第１号（１９９４），ｐ．４２－５２	１－５２
A	木村青司，他「可変サイズ可変形状ブロックに基づいた動き補償法」テレビジョン学会技術報告，第２０巻，第６４号（１９９６）（東京）ｐ．３１－３８	１－５２

☐ Ｃ欄の続きにも文献が列挙されている。　　☐ パテントファミリーに関する別紙を参照。

＊　引用文献のカテゴリー
「Ａ」特に関連のある文献ではなく、一般的技術水準を示すもの
「Ｅ」先行文献ではあるが、国際出願日以後に公表されたもの
「Ｌ」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献（理由を付す）
「Ｏ」口頭による開示、使用、展示等に言及する文献
「Ｐ」国際出願日前で、かつ優先権の主張の基礎となる出願の日の後に公表された文献
「Ｔ」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの
「Ｘ」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの
「Ｙ」特に関連のある文献であって、当該文献と他の１以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの
「＆」同一パテントファミリー文献

国際調査を完了した日	国際調査報告の発送日
２５．０８．９８	08.09.98

国際調査機関の名称及びあて先	特許庁審査官（権限のある職員）	５Ｃ	９０７０
日本国特許庁（ＩＳＡ／ＪＰ） 郵便番号１００－８９１５ 東京都千代田区霞が関三丁目４番３号	國分　直樹 電話番号　０３－３５８１－１１０１　内線　３５４２		

様式ＰＣＴ／ＩＳＡ／２１０（第２ページ）（１９９２年７月）