

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4879558号
(P4879558)

(45) 発行日 平成24年2月22日(2012.2.22)

(24) 登録日 平成23年12月9日(2011.12.9)

(51) Int.Cl. F 1
H04N 7/32 (2006.01) H04N 7/137 Z

請求項の数 1 (全 38 頁)

(21) 出願番号	特願2005-320189 (P2005-320189)	(73) 特許権者	000005821
(22) 出願日	平成17年11月2日(2005.11.2)		パナソニック株式会社
(65) 公開番号	特開2007-129498 (P2007-129498A)		大阪府門真市大字門真1006番地
(43) 公開日	平成19年5月24日(2007.5.24)	(74) 代理人	100109210
審査請求日	平成20年10月2日(2008.10.2)		弁理士 新居 広守
		(72) 発明者	西郷 賀津雄
			大阪府門真市大字門真1006番地 松下
			電器産業株式会社内
		(72) 発明者	重里 達郎
			大阪府門真市大字門真1006番地 松下
			電器産業株式会社内
		(72) 発明者	大古瀬 秀之
			大阪府門真市大字門真1006番地 松下
			電器産業株式会社内

最終頁に続く

(54) 【発明の名称】 動きベクトル検出装置

(57) 【特許請求の範囲】

【請求項 1】

参照ピクチャを蓄積する蓄積部と、

前記蓄積部に蓄積された同じ参照ピクチャを参照して、複数のピクチャに亘って空間的に同じ位置のブロックについてそれぞれ動きベクトルを検出した後、前記複数のピクチャ内の別の空間位置におけるブロックについてそれぞれ動きベクトルを検出する動きベクトル検出部と、を備え、

前記動き検出部は、複数のピクチャに亘って同じ空間位置のブロックについてそれぞれ動きベクトルを検出する際、前記複数のピクチャのうちいずれかのピクチャ内のブロックについて動きベクトルを検出する前に、既に他のピクチャ内の同じ空間位置のブロックについて動きベクトルを検出していた場合、検出済みの動きベクトルに基づいて、動きベクトルの検出対象であるブロックにおける動きベクトルの予測値を算出し、当該予測値に基づいて前記参照ピクチャ内の参照領域を決定し、前記決定された参照領域内のデータを用いて動きベクトルを検出し、

動きベクトルの検出対象である前記複数のピクチャのそれぞれは、前記同じ参照ピクチャを参照するピクチャであって、

前記検出済みの動きベクトルは、前記複数のピクチャのうち表示順で中間に位置するピクチャにおいて検出された動きベクトルである、

動きベクトル検出装置。

【発明の詳細な説明】

10

20

【技術分野】

【0001】

本発明は、複数のピクチャからなる動画像のピクチャ間予測符号化に用いられる動きベクトル検出装置であって、ピクチャの一部であるブロックが他のピクチャのどの位置から動いてきたかを検出し、その動きを動きベクトルで表現する動きベクトル検出装置に関する。

【背景技術】

【0002】

動画像符号化においては、一般に動画像が有する空間方向および時間方向の冗長性を利用して情報量の圧縮を行う。ここで、時間方向の冗長性を利用する方法として、ピクチャ間予測符号化が用いられる。ピクチャ間予測符号化では、あるピクチャを符号化する際に、表示順序で前方または後方にあるピクチャを参照ピクチャとする。そして、その参照ピクチャからの動きベクトルを検出し、動き補償を行ったピクチャと符号化対象のピクチャとの差分値に対して空間方向の冗長度を取り除くことにより情報量の圧縮を行う。

10

【0003】

MPEGなどの動画像符号化方式では、参照ピクチャを持たずに符号化対象ピクチャのみを用いてピクチャ内予測符号化を行うピクチャをIピクチャと呼ぶ。ここでピクチャとは、フレームおよびフィールドの両者を包含する1つの符号化の単位を意味する。また、既に符号化済みの1枚のピクチャを参照してピクチャ間予測符号化するピクチャをPピクチャと呼び、既に符号化済みの2枚のピクチャを同時に参照してピクチャ間予測符号化するピクチャをBピクチャと呼ぶ。

20

【0004】

図22は、動画像符号化方式における各ピクチャの予測関係を示す模式図である。

図22において、縦線は1枚のピクチャを示しており、各ピクチャの右下にピクチャタイプ(I、P、B)を示している。また図22中の矢印は、矢印の始端にあるピクチャが、矢印の終端にあるピクチャを参照ピクチャとして用いてピクチャ間予測符号化されることを示している。例えば、先頭から2枚目のBピクチャは、先頭のIピクチャと先頭から4枚目のPピクチャを参照ピクチャとして用いることにより符号化される。

【0005】

図23は、ピクチャの表示順序および符号化順序を示す図である。

30

図23の(a)に示すように、各ピクチャの表示順序は、PピクチャP1、BピクチャB2、BピクチャB3、PピクチャP4、BピクチャB5、BピクチャB6、PピクチャP7、BピクチャB8、BピクチャB9、PピクチャP10となる。一方、これらピクチャの符号化順序は、図23の(b)に示すように、BピクチャB0、PピクチャP4、BピクチャB2、BピクチャB3、PピクチャP7、BピクチャB5、BピクチャB6、PピクチャP10、BピクチャB8、BピクチャB9となる。

【0006】

なお、従来のMPEG-2のような動画像符号化方式に対し、最新の動画像符号化方式H.264では、Bピクチャは3枚以上のピクチャを参照してもよい。

【0007】

40

図24は、H.264における参照関係を示す図である。

この図24に示すように、Bピクチャは、例えば、前方の2枚のPピクチャと、後方の1枚のPピクチャとを参照する。そしてBピクチャにおける動きベクトルが検出される。このように、H.264では、同じ1枚のBピクチャに対して、動きベクトル検出のために参照可能な参照ピクチャの枚数が、MPEG2より多くなっている(例えば、非特許文献1参照)。

【0008】

ところで、ピクチャ間予測符号化における動きベクトルの検出は、ブロック単位に行なわれる。符号化対象ピクチャに含まれるブロックごとに、そのブロックの画像に最も近い画像を有するブロックが、参照ピクチャの中から探索される。そして、動きベクトルの検

50

出では、通常、演算量の負荷と動きベクトルの精度を考慮し、動きベクトル検出の探索範囲が事前に設定される。このとき、時間の経過とともに対象が移動する場合を想定し、符号化対象のピクチャと参照ピクチャとの間の距離に比例して、動きベクトルの探索範囲を拡大しておく必要がある。

【0009】

図25は、動きベクトルの検出における探索範囲を示す図である。

例えば、符号化対象ピクチャと参照ピクチャとの間の距離（ピクチャ間距離）が1のときに、動きベクトルの探索範囲が $\pm S \times \pm S$ である場合、そのピクチャ間距離が d になると、動きベクトルの探索範囲は $(d \times \pm S) \times (d \times \pm S)$ となる。このように、探索範囲は $d \times d$ 倍に拡大する。すなわち、ピクチャ間距離が長くなる場合、探索範囲は、その距離の変化の割合の自乗に比例して拡大する。

10

【0010】

図25に示すように、 $(n+3)$ 番目の符号化対象ピクチャに含まれるブロックの動きベクトル $MV1$ 、 $MV2$ 、 $MV3$ を、 n 番目の参照ピクチャ、 $(n+1)$ 番目の参照ピクチャ、または $(n+2)$ 番目の参照ピクチャを用いて検出する場合、 $(n+1)$ 番目の参照ピクチャの探索範囲は、 $(n+2)$ 番目の参照ピクチャの探索範囲の 2×2 倍であり、 n 番目の参照ピクチャの探索範囲は、 $n+2$ 番目の参照ピクチャの探索範囲の 3×3 倍となる。

【0011】

このように、ピクチャ間距離が増大すると、動きベクトルの探索範囲は急激に拡大し、動きベクトル検出の演算回数も膨大になる。このため、演算回数を削減するために、テレスコピックサーチにより動きベクトルを検出する動きベクトル検出装置が提案されている（例えば、特許文献1、2及び3参照）。

20

【0012】

テレスコピックサーチは、参照ピクチャと符号化対象ピクチャとの間に存在するピクチャに対して順次動き探索を行うことにより、動きベクトルを検出する手法である。この方法では、ピクチャ間距離が長くなっても、探索範囲は、その距離の変化の割合の自乗ではなく、その割合に比例して拡大する。

【0013】

図26は、テレスコピックサーチを示す図である。

30

例えば、 $(n+3)$ 番目の符号化対象ピクチャに含まれるブロック（対象ブロック）の動きベクトルを、 n 番目の参照ピクチャを用いて検出する。この場合、テレスコピックサーチでは、まず、 $(n+2)$ 番目のピクチャの対象ブロックと同一位置を中心とする探索範囲 $(\pm S \times \pm S)$ を用いて、 $(n+3)$ 番目のピクチャから $(n+2)$ 番目のピクチャに対する動きベクトル $v1$ を検出する。続いて、 $(n+1)$ 番目のピクチャにある、動きベクトル $v1$ の示す位置と同一位置を中心とする探索範囲 $(\pm S \times \pm S)$ を用いて、 $(n+3)$ 番目のピクチャから $(n+1)$ 番目のピクチャに対する動きベクトル $v2 + v1$ を検出する。同様に、 n 番目のピクチャにある、動きベクトル $v2 + v1$ の示す位置と同一位置を中心とする探索範囲 $(\pm S \times \pm S)$ を用いて、 $n+3$ 番目のピクチャから n 番目のピクチャへの動きベクトル $v3 + v2 + v1$ を検出する。この動きベクトルを対象ブロックの動きベクトル $v0$ とする。つまり、全体の探索範囲は $3 \times (\pm S \times \pm S)$ となる。

40

【非特許文献1】Draft of Version 4 of H.264/AVC (ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4 part 10) Advanced Video Coding), Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, Document: JVT-N050d1, 2005-01-28

【特許文献1】特許第2830183号公報

【特許文献2】特許第3335137号公報

【特許文献3】特開平10-341440号公報

50

【発明の開示】

【発明が解決しようとする課題】

【0014】

しかしながら、テレスコピックサーチを行なう特許文献1～3の動きベクトル検出装置であっても、動きベクトルを検出するときの演算量が多いという問題がある。

【0015】

つまり、テレスコピックサーチでは、参照ピクチャと符号化対象ピクチャの間に存在するピクチャに対して順次動き探索を行うことにより、各ピクチャにおける探索範囲は、それぞれピクチャ間距離に関係なく一定となる。しかし、テレスコピックサーチでは、参照ピクチャと符号化対象ピクチャの間に存在するピクチャに対しても順次動き探索を行うため、参照ピクチャとは無関係な途中のピクチャまで参照しなければならず、その参照回数はピクチャ間距離に比例する。したがって、全体的な探索範囲は、ピクチャ間距離に比例して大きくなり、その結果、動きベクトルを検出するときの演算量が多くなるのである。

【0016】

さらに、テレスコピックサーチを行なう動きベクトル検出装置では、回路規模が増大するという問題がある。つまり、この動きベクトル検出装置では、参照ピクチャを格納しているメモリから、動きベクトルの検出のために読み出される探索範囲のデータ量が多い。その結果、この動きベクトル検出装置では、メモリの転送動作クロックを高速化したりメモリバンド幅(ビット幅)を広げたりするために回路規模が増大するのである。

【0017】

図27は、一枚の参照ピクチャにおける探索範囲の変化を示す図である。

一般に、対象ブロック(通常は、マクロブロックであって、例えば、16画素×16ラインから構成されるブロック)に対する参照ピクチャRP1の探索範囲TA1は、参照ピクチャRP1において対象ブロックと同一位置にあるブロックTB1を中心とする範囲である。ここで、参照ピクチャRP1は、水平H画素および垂直Vラインから構成され、探索範囲TA1は、水平h画素および垂直vラインから構成される。したがって、対象ブロックの動きベクトルを検出するときには、水平h画素および垂直vラインのデータがメモリから読み出される。

【0018】

次に、対象ブロックが水平方向にシフトすると、その対象ブロックに対する参照ピクチャRP1の探索範囲TA2は、参照ピクチャRP1において対象ブロックと同一位置にあるブロックTB2を中心とする範囲となる。したがって、対象ブロックが水平方向にシフトすると、探索範囲もブロックの幅wだけシフトする。この場合、対象ブロックの動きベクトルを検出するときメモリから読み出されるデータは、探索範囲TA2のデータのうち探索範囲TA1に含まれていなかったデータのみとなる。即ち、新たにメモリから読み出されるデータは、16画素および垂直vラインのデータとなる。

【0019】

このように、対象ブロックが水平方向にシフトすると、新たな探索範囲に含まれる全てのデータがメモリから読み出されるのではなく、直前の探索範囲に含まれていなかったデータのみがメモリから読み出される。

【0020】

ここで、テレスコピックサーチの場合、互いに隣接するピクチャごとに上述のような探索範囲のデータがメモリから読み出される。しかし、この場合には、各ピクチャの探索範囲が常にブロックの幅だけシフトするとは限らず、大きくシフトする可能性がある。そのため、探索範囲のデータをメモリから読み出して動きベクトルの検出を行い、そのデータを次の動きベクトルの検出用に蓄積しておいても、ほとんど重複するデータがないので、新たな探索範囲のデータの多くをメモリから読み込む必要がある。したがって、テレスコピックサーチの場合には、メモリからの転送データ量が多くなるのである。また、その結果、消費電力が上昇してしまう。

【0021】

そこで、本発明は、かかる問題に鑑みてなされたものであって、動きベクトルの検出のための演算量を低減して回路規模の縮小化を図った動きベクトル検出装置を提供することを目的とする。

【課題を解決するための手段】

【0022】

上記目的を達成するために、本発明に係る動きベクトル検出装置は、参照ピクチャを蓄積する蓄積部と、前記蓄積部に蓄積された同じ参照ピクチャを参照して、複数のピクチャに亘って空間的に同じ位置のブロックについてそれぞれ動きベクトルを検出した後、前記複数のピクチャ内の別の空間位置におけるブロックについてそれぞれ動きベクトルを検出する動きベクトル検出部と、を備え、前記動き検出部は、複数のピクチャに亘って同じ空間位置のブロックについてそれぞれ動きベクトルを検出する際、前記複数のピクチャのうちいずれかのピクチャ内のブロックについて動きベクトルを検出する前に、既に他のピクチャ内の同じ空間位置のブロックについて動きベクトルを検出していた場合、検出済みの動きベクトルに基づいて、動きベクトルの検出対象であるブロックにおける動きベクトルの予測値を算出し、当該予測値に基づいて前記参照ピクチャ内の参照領域を決定し、前記決定された参照領域内のデータを用いて動きベクトルを検出し、動きベクトルの検出対象である前記複数のピクチャのそれぞれは、前記同じ参照ピクチャを参照するピクチャであって、前記検出済みの動きベクトルは、前記複数のピクチャのうち表示順で中間に位置するピクチャにおいて検出された動きベクトルである。

10

上記目的を達成するために、本発明に係る動きベクトル検出装置は、参照ピクチャを蓄積する蓄積部と、前記蓄積部に蓄積された同じ参照ピクチャを参照して、複数のピクチャに亘って空間的に同じ位置のブロックについてそれぞれ動きベクトルを検出した後、前記複数のピクチャ内の別の空間位置におけるブロックについてそれぞれ動きベクトルを検出する動きベクトル検出部と、を備える。

20

また、上記目的を達成するために、本発明に係る動きベクトル検出装置は、ピクチャに含まれる対象ブロックの動きベクトルを検出する動きベクトル検出装置であって、第1のピクチャに含まれる対象ブロックと近似する画像を有するブロックを、参照ピクチャから探索することにより、前記第1のピクチャに含まれる前記対象ブロックの第1の動きベクトルを検出する第1の動き検出手段と、前記第1の動きベクトルに基づいて、前記参照ピクチャの探索範囲を特定する探索範囲特定手段と、第2のピクチャに含まれる対象ブロックと近似する画像を有するブロックを、前記探索範囲から探索することにより、前記第2のピクチャに含まれる前記対象ブロックの第2の動きベクトルを検出する第2の動き検出手段とを備えることを特徴とする。例えば、前記探索範囲特定手段は、前記第1および第2のピクチャと前記参照ピクチャとの間での表示時刻の時間差に応じて、前記第1の動きベクトルをスケールリングすることにより、前記第2のピクチャの前記対象ブロックの予測動きベクトルを算出する予測演算手段と、前記予測演算手段により算出された予測動きベクトルにより示される前記参照ピクチャの範囲を前記探索範囲として特定する特定手段とを備える。

30

【0023】

第1および第2のピクチャの表示時刻が互いに近く、それぞれのピクチャに含まれる対象ピクチャが空間的に同一の位置にあれば、第1および第2の動きベクトルの向きは等しくなる傾向にある。したがって、本発明では、例えば、第1の動きベクトルから予測動きベクトルが算出されて探索範囲が特定されるように、第1の動きベクトルに基づいて探索範囲が特定されるため、第2の動きベクトルを検出するための探索範囲を適切に特定することができる。その結果、従来のように、第2のピクチャが参照ピクチャから表示時間的に離れていても、参照ピクチャにおける探索範囲を広げることなく、適切な探索範囲で第2のピクチャの対象ブロックの動きベクトルを検出することができる。また、本発明では、テレスコピックサーチのように、直接必要のない動きベクトルまで検出する必要がない。その結果、本発明では全体的な探索範囲を小さくすることができる。さらに、本発明では、第2のピクチャの対象ブロックをずらしながら各対象ブロックの第2の動きベクトル

40

50

を検出するときにも、テレスコピックサーチと比べて、参照ピクチャを格納しているメモリ（ピクチャメモリ）からの転送データ量を少なくすることができる。即ち、本発明では、表示順序で参照ピクチャと第2のピクチャとの間に他のピクチャがあっても、他のピクチャにおける探索範囲のデータをメモリから読み出す必要がなく、さらに、対象ブロックをずらしながら各対象ブロックの第2の動きベクトルを検出するときには、前回に読み出した探索範囲を有効に利用して、新たにメモリから読み出すデータ量を減らすことができる。

【0024】

したがって、本発明では、探索範囲を適切な範囲とすることができるため、動きベクトルの検出のための演算量を低減することができるとともに、メモリからの転送データ量を少なくすることができるため、回路規模の縮小化を図ることができる。

10

【0025】

また、前記予測演算手段は、前記第2のピクチャにおいて前記第1のピクチャの前記対象ブロックと空間的同一位置にある同位置ブロックを、前記第2のピクチャの前記対象ブロックとして、当該対象ブロックの前記予測動きベクトルを算出し、前記第2の動き検出手段は、前記同位置ブロックを前記第2のピクチャの前記対象ブロックとして、当該対象ブロックの前記第2の動きベクトルを検出することを特徴としてもよい。

【0026】

これにより、第1のピクチャの対象ブロックと第2のピクチャの対象ブロックとがそれぞれ、各ピクチャ内において空間的に同一の位置にあるため、予測動きベクトルを第2の動きベクトルに近づけることができる。即ち、予測動きベクトルの精度を向上することができ、その結果、探索範囲をより適切に特定することができる。

20

【0027】

また、前記第1の動き検出手段は、前記第1のピクチャごとに、当該第1のピクチャに含まれる前記対象ブロックの前記第1の動きベクトルを検出し、前記探索範囲特定手段は、複数の前記第1のピクチャのうち、前記第2のピクチャとの間の表示時刻の時間差が最も短いピクチャの第1の動きベクトルに基づいて、前記参照ピクチャの探索範囲を特定することを特徴としてもよい。

【0028】

これにより、第1の動きベクトルが複数あるときには、第2のピクチャと表示時間的に近い第1のピクチャの第1の動きベクトルに基づいて探索範囲が特定されるため、第1の動きベクトルが複数あっても、適切な探索範囲を特定することができる。

30

【0029】

また、前記特定手段は、前記予測動きベクトルにより示される前記参照ピクチャの範囲と、前記参照ピクチャにおいて前記第2のピクチャの前記対象ピクチャと空間的同一位置にあるブロックとを、前記探索範囲として特定することを特徴としてもよい。

【0030】

これにより、前記参照ピクチャにおいて第2のピクチャの対象ピクチャと空間的に同一位置にあるブロックが探索範囲に含まれるため、予測動きベクトルにより示される参照ピクチャの範囲に、第2のピクチャの対象ピクチャと近似する画像を有するブロックがなくとも、その他の範囲からそのブロックを探索することができ、第2動きベクトルを適切に検出することができる。

40

【0031】

また、前記第1または第2の動き検出手段は、既に検出した前記第1または第2の動きベクトルを用いた前記第1または第2のピクチャの前記対象ブロックに対する符号化処理が行なわれている間に、前記第1または第2のピクチャに含まれる他の対象ブロックの第1または第2の動きベクトルを検出することを特徴としてもよい。

【0032】

これにより、この動きベクトル検出装置が動画像符号化装置に組み込まれたときには、符号化処理と動き検出処理とがパイプライン処理されるため、全体的な処理時間の短縮化

50

を図ることができる。

【0033】

なお、本発明は、このような動きベクトル検出装置として実現することができるだけでなく、その方法やプログラム、そのプログラムを格納する記憶媒体、集積回路としても実現することができる。

【発明の効果】

【0034】

本発明の動きベクトル検出装置は、動きベクトルの探索範囲を小さくして動きベクトルの演算量を低減することができるとともに、外部のメモリから読み出すデータ量を低減することで、低消費電力化および回路規模の縮小化を図ることができるという作用効果を奏する。

10

【発明を実施するための最良の形態】

【0035】

以下、本発明の実施の形態における動きベクトル検出装置を備えた動画像符号化装置について図面を参照しながら説明する。

【0036】

図1は、本発明の実施の形態における動きベクトル検出装置を備えた動画像符号化装置のブロック図である。

【0037】

動画像符号化装置100は、図1に示すように、ピクチャメモリ112、120、ブロック選択部111、差分演算部113、スイッチ114、127、符号化部115、整列化バッファ116、符号列生成部117、復号化部118、加算演算部119、動きベクトル検出部121、記憶部122、予測演算部123、探索範囲制御部124、モード選択部125および動き補償部126を備えている。

20

【0038】

ピクチャメモリ112は、図23の(a)に示すように、動画像を示す複数の入力画像(ピクチャ)を表示順序で取得して格納する。そして、ピクチャメモリ112は、ブロック選択部111で選択された各ピクチャの対象ブロック(動きベクトルの検出対象や符号化対象となるブロック)を差分演算部113、スイッチ114および動きベクトル検出部121へ出力する。

30

【0039】

ブロック選択部111は、ピクチャメモリ112に格納されたピクチャから、連続する複数枚のピクチャ単位、例えば、図23の(a)に示すピクチャP1、B2、B3、P4のような単位を選択する。さらに、ブロック選択部111は、それら各ピクチャから所定の位置の対象ブロックを選択して出力するようにピクチャメモリ112を制御する。また、ブロック選択部111は、選択した対象ブロックの位置情報とその対象ブロックが含まれるピクチャ(対象ピクチャ)の表示順情報を、動きベクトル検出部121、記憶部122、予測演算部123、探索範囲制御部124および整列化バッファ116に出力する。

【0040】

差分演算部113は、入力画像に含まれる対象ブロックの画像と、動き補償部126から出力された予測画像との差分を算出し、その算出結果を予測残差画像として出力する。

40

【0041】

スイッチ114は、対象ブロックに対してピクチャ内予測符号化が行われるときには、対象ブロックを符号化部115に出力し、対象ブロックに対してピクチャ間予測符号化が行なわれるときには、差分演算部113からの予測残差画像を符号化部115に出力する。

【0042】

スイッチ127は、対象ブロックに対してピクチャ間処理が行なわれるときにのみ、動き補償部126からの予測画像を加算演算部119に出力する。

【0043】

50

符号化部 115 は、スイッチ 114 から出力された対象ブロックまたは予測残差画像に対して周波数変換や量子化などの符号化処理を行なうことで符号化信号を生成する。

【0044】

整列化バッファ 116 は、符号化信号に含まれるピクチャを規格に従った並びに配列し、その符号化信号を符号列生成部 117 に出力する。

【0045】

符号列生成部 117 は、整列化バッファ 116 から出力された符号化信号に対して可変長符号化等を行う。さらに、符号列生成部 117 は、モード選択部 125 から動きベクトルや、動き予測のモードを示す情報（後述するダイレクトモード選択情報）などを取得し、可変長符号化された符号化信号に付加することにより、符号列を生成する。

10

【0046】

復号化部 118 は、符号化部 115 から出力された符号化信号に対して逆量子化や逆周波数変換等の復号化処理を行うことで復号化画像を生成する。

【0047】

加算演算部 119 は、復号化部 118 から出力された復号化画像と、動き補償部 126 からスイッチ 127 を介して出力された予測画像とを加算することで、再構築画像を生成する。

【0048】

ピクチャメモリ 120 は、加算演算部 119 で生成されて出力された再構築画像を順次格納する。その結果、ピクチャメモリ 120 には参照ピクチャが格納される。

20

【0049】

動きベクトル検出部 121 は、ピクチャメモリ 112 から出力された対象ブロックを取得する。また、動きベクトル検出部 121 は、その対象ブロックに対する参照ピクチャの探索範囲をピクチャメモリ 120 から読み出す。そして、動きベクトル検出部 121 は、対象ブロックの画像に近似する画像を有するブロックをその探索範囲から探索することにより、対象ブロックの画像の動きを示す動きベクトル（参照ピクチャに対する対象ブロックの動きベクトル）を検出する。

【0050】

予測演算部 123 は、後述する時間ダイレクトモードにより、対象ブロックの動きベクトルを予測して予測動きベクトルを算出する。

30

【0051】

探索範囲制御部 124 は、予測演算部 123 により算出された予測動きベクトルに基づいて、参照ピクチャの探索範囲を特定してその探索範囲を動きベクトル検出部 121 に設定する。

【0052】

記憶部 122 は、動きベクトル検出部 121 で検出された動きベクトルを記憶するための領域を有する。

【0053】

モード選択部 125 は、動き予測のモードを選択する。即ち、モード選択部 125 は、時間ダイレクトモードと非時間ダイレクトモードの何れかを選択する。そして、モード選択部 125 は、非時間ダイレクトモードを選択したときには、動きベクトル検出部 121 で検出された動きベクトルを符号列生成部 117 に出力し、時間ダイレクトモードを選択したときには、時間ダイレクトモードが選択されたことを通知するためのダイレクトモード選択情報を符号列生成部 117 に出力する。

40

【0054】

また、本実施の形態における動きベクトル検出装置 100 a は、動きベクトルの検出のための演算量を低減して回路規模の縮小化を図ることができるという点に特徴があり、ブロック選択部 111 と、モード選択部 125 と、動きベクトル検出部 121 と、記憶部 122 と、予測演算部 123 と、探索範囲制御部 124 とを備えて構成されている。さらに、本実施の形態では、動きベクトル検出部 121 が、第 1 の動きベクトルを検出する第 1

50

の動き検出手段と、第2の動きベクトルを検出する第2の動き検出手段とを備えて構成され、予測演算部123と探索範囲制御部124とが、第1の動きベクトルに基づいて参照ピクチャの探索範囲を特定する探索範囲特定手段として構成されている。また、本実施の形態では、予測演算部123が、予測演算手段として構成され、探索範囲制御部124が、予測動きベクトルにより示される参照ピクチャの範囲を探索範囲として特定する特定手段として構成されている。

【0055】

ここで、一般的な時間ダイレクトモードについて詳細に説明する。

H.264方式では、Bピクチャの符号化において、動きベクトルの情報を削減するため、ダイレクトモードという符号化モードを選択することができる。このダイレクトモードには、時間的方法と空間的方法との2種類の方法がある。時間ダイレクトモードでは、対象ブロック自体は動きベクトルを持たず、符号化済みの他のピクチャの動きベクトルを参照動きベクトルとして、ピクチャ間の表示時間的位置関係に基づいてスケーリングすることによって、対象ブロックの動きベクトルを予測して生成している。

10

【0056】

図2は、時間ダイレクトモードにおける動きベクトルの予測生成方法を示す模式図である。なお、この図2に示すP1, B2などの符号中、PはPピクチャ、BはBピクチャを示し、数字は各ピクチャの表示順序を示している。また、各ピクチャP1, B2, B3, P4は、それぞれ表示順情報T1, T2, T3, T4を有している。つまり、各ピクチャは、ピクチャP1、ピクチャB2、ピクチャB3、ピクチャP4の順に表示される。

20

【0057】

ここで、ピクチャB3のブロックBL0は、時間ダイレクトモードで符号化される。この場合、ピクチャB3の表示時間的に近傍に位置する既に符号化済みのピクチャであるピクチャP4中の、ブロックBL0と同じ位置にあるブロックBL1の動きベクトルMV1が利用される。この動きベクトルMV1は、ブロックBL1が符号化される際に用いられた動きベクトルであり、ピクチャP1を参照ピクチャとしている。

【0058】

つまり時間ダイレクトモードでは、動きベクトルMV1を下記の(式1a)および(式1b)に適用することにより、ブロックBL0のピクチャP1に対する動きベクトル(前方動きベクトル)MVfと、ブロックBL0のピクチャP4に対する動きベクトル(後方動きベクトル)MVBとを算出する。なお、(式1a)および(式1b)中、Mvは動きベクトルMV1の大きさを示し、Mvfは動きベクトルMVfの大きさを示し、Mvbは動きベクトルMVBの大きさを示す。

30

【0059】

$$Mvf = (T3 - T1) / (T4 - T1) \times Mv \quad \dots \quad (\text{式 } 1a)$$

$$Mvb = (T4 - T3) / (T4 - T1) \times Mv \quad \dots \quad (\text{式 } 1b)$$

【0060】

このように、時間ダイレクトモードでは、動きベクトルMV1をスケーリングすることによって得られる動きベクトルMVfおよび動きベクトルMVBを用いて、ピクチャP1とピクチャP4を参照ピクチャとするブロックBL0の動き補償を行う。そのため、時間ダイレクトモードであることを示す情報を符号列に含めるだけで、ブロックBL0の動きベクトルを符号化して符号列に含める必要がなく、その結果、符号列のビットレートを削減することができる。

40

【0061】

本実施の形態の動きベクトル検出装置100aでは、上述のような時間ダイレクトモードにより対象ブロックの動きベクトルを予測する。つまり、本実施の形態の動きベクトル検出装置100aは、対象ブロックの予測動きベクトルを算出して、その予測動きベクトルから参照ピクチャの探索範囲を特定し、その探索範囲から動きベクトルを検出する。

【0062】

図3は、本実施の形態における予測演算部123によって算出される前方向の予測動き

50

ベクトルを説明するための説明図である。なお、この図3中の符号P1およびP4はPピクチャを示し、符号B2およびB3はBピクチャを示す。また、これらの符号中の数字は、各ピクチャの表示順序を示している。また、各ピクチャP1, B2, B3, P4は、それぞれ表示順情報T1, T2, T3, T4を有している。この表示順情報は、各ピクチャの表示順序や表示時刻を示している。

【0063】

動きベクトル検出部121は、まず、第1のピクチャたるピクチャB2の対象ブロックBL0の動きベクトルMV0F(第1の動きベクトル)を検出する。即ち、動きベクトル検出部121は、参照ピクチャP1においてピクチャB2の対象ブロックBL0と同じ位置を中心とするS画素×Sラインの探索範囲(部分範囲)から、対象ブロックBL0の画

10

【0064】

予測演算部123は、その動きベクトルMV0Fを記憶部122から読み出して、動きベクトルMV0Fをスケーリングすることにより、第2のピクチャたるピクチャB3およびピクチャP4において対象ブロックBL0と同じ位置(空間的同一位置)にある対象ブロックBL1, BL2に対して、それぞれ前方向の予測動きベクトルMV1F*と予測動きベクトルMV2*とを算出する。

【0065】

具体的に、予測演算部123は、ピクチャP1, B2, B3, P4の表示順情報T1, T2, T3, T4を使い、下記の(式2a)および(式2b)によって、ピクチャB3の対象ブロックBL1の参照ピクチャP1に対する予測動きベクトルMV1F*と、ピクチャP4の対象ブロックBL2の参照ピクチャP1に対する予測動きベクトルMV2*とを算出する。なお、(式2a)および(式2b)中、Mv1f*およびMv2*はそれぞれ予測動きベクトルMV1F*, MV2*の大きさを示し、Mv0fは動きベクトルMV0Fの大きさを示す。

20

【0066】

$$Mv1f* = (T3 - T1) / (T2 - T1) \times Mv0f \quad \dots (式2a)$$

$$Mv2* = (T4 - T1) / (T2 - T1) \times Mv0f \quad \dots (式2b)$$

30

【0067】

ここで、各ピクチャP1, B2, B3, P4の表示時間間隔(表示時刻の時間差)が等間隔である場合には、参照ピクチャP1の探索可能最大範囲は、(3×S)画素×(3×S)ラインの範囲となる。つまり、ピクチャB2と参照ピクチャP1との表示時間間隔が(T2 - T1) = 1の場合、参照ピクチャP1とピクチャP4との表示時間間隔は(T4 - T1) = 3となる。したがって、ピクチャB2の動きベクトルMV0Fの探索範囲がS画素×Sラインである場合には、探索可能最大範囲は、(3×S)画素×(3×S)ラインの範囲となる。このように、探索可能最大範囲は、各対象ピクチャと参照ピクチャP1との間の表示時刻の時間差のうち最も長い時間差に応じた大きさとなる。

【0068】

予測演算部123は、このように算出した予測動きベクトルMV1F*, MV2F*を探索範囲制御部124に出力する。

40

【0069】

探索範囲制御部124は、参照ピクチャP1において予測動きベクトルMV1F*により示される探索範囲を特定して、動きベクトル検出部121に設定する。即ち、探索範囲制御部124は、予測動きベクトルMV1F*により示される参照ピクチャP1の位置を中心とするS画素×Sラインの探索範囲を設定する。また、探索範囲制御部124は、参照ピクチャP1において予測動きベクトルMV2*により示される探索範囲を特定して、動きベクトル検出部121に設定する。即ち、探索範囲制御部124は、予測動きベクトルMV2*により示される参照ピクチャP1の位置を中心とするS画素×Sラインの探索

50

範囲を設定する。このように、探索範囲制御部 124 は、対象ブロック B L 1 , B L 2 の動きベクトルを検出するために、対象ブロック B L 0 の動きベクトル M V 0 F の探索範囲（部分範囲）と同じ大きさの S 画素 × S ラインの探索範囲を探索可能最大範囲から特定している。

【 0 0 7 0 】

動きベクトル検出部 121 は、探索範囲制御部 124 によって設定された参照ピクチャ P 1 の各探索範囲をピクチャメモリ 120 から読み出し、その探索範囲のそれぞれから、対象ブロック B L 1 の前方動きベクトル（第 2 の動きベクトル）と対象ブロック B L 2 の動きベクトル（第 2 の動きベクトル）とを検出する。つまり、動きベクトル検出部 121 は、参照ピクチャ P 1 に対する対象ブロック B L 0 の前方動きベクトルを検出した後、参照ピクチャ P 1 に対する対象ブロック B L 1 の前方動きベクトルを検出し、その次に、参照ピクチャ P 1 に対する対象ブロック B L 2 の前方動きベクトルを検出する。そして、動きベクトル検出部 121 は、対象ブロック B L 1 および対象ブロック B L 2 のそれぞれの動きベクトルを記憶部 122 に格納する。

10

【 0 0 7 1 】

図 4 は、動きベクトルの検出の順序を示す図である。

この図 4 では、各ブロックの動きベクトルの検出順序を、各ブロック内の数字（ 1 , 2 , ... , N , N + 1 , ... ）で示している。即ち、動きベクトル検出装置 100 a は、まず、ピクチャ B 2 の左上端のブロックの動きベクトルを検出し、その動きベクトルを用いて、ピクチャ B 3 の左上端のブロックの動きベクトルを検出し、その後、ピクチャ P 4 の左上端のブロックの動きベクトルを検出する。次に、動きベクトル検出装置 100 a は、ピクチャ B 2 の左上端から右に 2 番目のブロックの動きベクトルを検出し、その動きベクトルを用いて、ピクチャ B 3 の左上端から右に 2 番目のブロックの動きベクトルを検出し、その後、ピクチャ P 4 の左上端から右に 2 番目のブロックの動きベクトルを検出する。このように、動きベクトル検出装置 100 a は、各ピクチャにおいて対象ブロックを右にずらしながら、ピクチャ B 2、ピクチャ B 3、ピクチャ P 4 の順に、対象ブロックの動きベクトルを検出する。

20

【 0 0 7 2 】

そして、各ピクチャにおける最上段の全てのブロックの動きベクトルが検出されると、動きベクトル検出装置 100 a は、その段の下段にある各ブロックを対象ブロックとし、上述と同様の順序で動きベクトルを検出する。動きベクトル検出装置 100 a は、このように動きベクトルの検出の対象となるブロックを左から右に、上から下にずらしながら、ピクチャ B 2、ピクチャ B 3、ピクチャ P 4 の順に、全てのブロックに対して動きベクトルを検出する。その結果、ピクチャ B 2 およびピクチャ B 3 の全てのブロックの参照ピクチャ P 1 に対する前方動きベクトルが検出され、ピクチャ P 4 の全てのブロックの参照ピクチャ P 1 に対する動きベクトルが検出される。

30

【 0 0 7 3 】

ここで、ピクチャ P 4 の対象ブロックの参照ピクチャ P 1 に対する動きベクトルが検出されると、動き補償部 126 は、その検出された動きベクトルと参照ピクチャ P 1 を用いて動き補償を行い、ピクチャ P 4 の各ブロックの予測画像を生成する。そして、差分演算部 113、符号化部 115、復号化部 118 および加算演算部 119 は、その予測画像から符号化信号を経て再構築画像を生成し、その再構築画像をピクチャメモリ 120 に格納する。その結果、ピクチャ P 4 の全てのブロックの動きベクトルが検出された後には、ピクチャ P 4 が参照ピクチャとしてピクチャメモリ 120 に格納される。

40

【 0 0 7 4 】

このように本実施の形態における動きベクトル検出装置 100 a は、対象ピクチャに含まれる 1 つのブロックの動きベクトルを検出すると、表示順序で次の対象ピクチャに含まれる 1 つのブロックの動きベクトルを検出するように、表示順序に従った順序で動きベクトルを検出する。つまり、本実施の形態では、従来のように、P ピクチャに含まれる全てのブロックの動きベクトルを順次検出した後に、表示順序で前方の B ピクチャに含まれる

50

全てのブロックの動きベクトルを順次検出するような、表示順序に逆らった順序で動きベクトルを検出しな

【 0 0 7 5 】

次に、動きベクトル検出装置 1 0 0 a は、ピクチャ B 2 およびピクチャ B 3 の後方動きベクトルを検出する。このときにも上述と同様に、動きベクトル検出装置 1 0 0 a は、ピクチャ B 2 およびピクチャ B 3 の対象ブロックごとに後方向の予測動きベクトルを時間ダイレクトモードにより算出し、それらの予測動きベクトルにより示される位置を中心とする探索範囲から後方動きベクトルを検出する。

【 0 0 7 6 】

図 5 は、本実施の形態における予測演算部 1 2 3 によって算出される後方向の予測動きベクトルを説明するための説明図である。

10

【 0 0 7 7 】

予測演算部 1 2 3 は、ピクチャ B 2 およびピクチャ B 3 のそれぞれ同じ位置にある対象ブロック B L 0 と対象ブロック B L 1 の後方向の予測動きベクトルを算出するときには、まず、参照ピクチャ P 4 において対象ブロック B L 0 , B L 1 と同じ位置にあるブロック B L 2 の動きベクトル M V 2 を記憶部 1 2 2 から読み出す。そして、予測演算部 1 2 3 は、動きベクトル M V 2 をスケーリングすることにより、対象ブロック B L 0 , B L 1 に対して、予測動きベクトル M V 0 B * と予測動きベクトル M V 1 B * とを算出する。

【 0 0 7 8 】

具体的に、予測演算部 1 2 3 は、ピクチャ P 1 , B 2 , B 3 , P 4 の表示順情報 T 1 , T 2 , T 3 , T 4 を使い、下記の (式 3 a) および (式 3 b) によって、ピクチャ B 2 の対象ブロック B L 0 の参照ピクチャ P 4 に対する予測動きベクトル M V 0 B * と、ピクチャ B 3 の対象ブロック B L 1 の参照ピクチャ P 4 に対する予測動きベクトル M V 1 B * とを算出する。なお、(式 3 a) および (式 3 b) 中、M v 0 b * および M v 1 b * はそれぞれ予測動きベクトル M V 0 B * , M V 1 B * の大きさを示し、M v 2 は動きベクトル M V 2 の大きさを示す。

20

【 0 0 7 9 】

$$M v 0 b * = (T 2 - T 4) / (T 4 - T 1) \times M v 2 \dots \quad (式 3 a)$$

$$M v 1 b * = (T 3 - T 4) / (T 4 - T 1) \times M v 2 \dots \quad (式 3 b)$$

【 0 0 8 0 】

ここで、各ピクチャ P 1 , B 2 , B 3 , P 4 の表示時間間隔が等間隔である場合には、参照ピクチャ P 4 の探索可能最大範囲は、(2 x S) 画素 x (2 x S) ラインの範囲となる。

30

【 0 0 8 1 】

予測演算部 1 2 3 は、このように算出した予測動きベクトル M V 0 B * , M V 1 B * を探索範囲制御部 1 2 4 に出力する。

【 0 0 8 2 】

探索範囲制御部 1 2 4 は、参照ピクチャ P 4 において予測動きベクトル M V 0 B * により示される探索範囲を動きベクトル検出部 1 2 1 に設定する。即ち、探索範囲制御部 1 2 4 は、予測動きベクトル M V 0 B * により示される参照ピクチャ P 4 の位置を中心とする S 画素 x S ラインの探索範囲を設定する。また、探索範囲制御部 1 2 4 は、参照ピクチャ P 4 において予測動きベクトル M V 1 B * により示される探索範囲を動きベクトル検出部 1 2 1 に設定する。即ち、探索範囲制御部 1 2 4 は、予測動きベクトル M V 1 B * により示される参照ピクチャ P 4 の位置を中心とする S 画素 x S ラインの探索範囲を設定する。

40

【 0 0 8 3 】

動きベクトル検出部 1 2 1 は、探索範囲制御部 1 2 4 によって設定された参照ピクチャ P 4 の探索範囲から、対象ブロック B L 0 の後方動きベクトルと対象ブロック B L 1 の後方動きベクトルとを検出する。つまり、動きベクトル検出部 1 2 1 は、ピクチャ P 4 のブロック B L 2 の動きベクトル M V 2 を検出した後、参照ピクチャ P 4 に対する対象ブロック B L 0 の後方動きベクトルを検出し、その次に、参照ピクチャ P 4 に対する対象ブロッ

50

ク B L 1 の後方動きベクトルを検出する。そして、動きベクトル検出部 1 2 1 は、対象ブロック B L 0 および対象ブロック B L 1 のそれぞれの後方動きベクトルを記憶部 1 2 2 に格納する。

【 0 0 8 4 】

動きベクトル検出装置 1 0 0 a は、このようなピクチャ B 2 およびピクチャ B 3 に含まれる全てのブロックの後方動きベクトルの検出を、左上端のブロックから順に、対象ブロックを左から右に上から下にずらしながら行なう。

【 0 0 8 5 】

図 6 は、本実施の形態の動きベクトル検出装置 1 0 0 a が動きベクトルを検出する動作を示すフローチャートである。

【 0 0 8 6 】

まず、動きベクトル検出装置 1 0 0 a のブロック選択部 1 1 1 は、ピクチャメモリ 1 1 2 から対象ブロックを抽出する（ステップ S 1 0 0）。例えば、ブロック選択部 1 1 1 は、図 3 に示すように、対象ピクチャ B 2, B 3, P 4 のそれぞれから、ブロック B L 0, B L 1, B L 2 を対象ブロックとして抽出する。そして、ブロック選択部 1 1 1 は、それらの対象ブロックから最初に処理すべきブロック、例えばブロック B L 0 を選択する（ステップ S 1 0 2）。

【 0 0 8 7 】

動きベクトル検出部 1 2 1 は、ブロック選択部 1 1 1 によって選択された対象ブロック B L 0 の参照ピクチャ P 1 に対する動きベクトル（前方動きベクトル）M V 0 F を検出する（ステップ S 1 0 4）。そして、動きベクトル検出部 1 2 1 は、その検出した動きベクトル M V 0 F を記憶部 1 2 2 に格納する（ステップ S 1 0 6）。

【 0 0 8 8 】

予測演算部 1 2 3 は、時間ダイレクトモードにより、つまり、動きベクトル検出部 1 2 1 によって検出された動きベクトル M V 0 F をスケールリングすることにより、対象ブロック B L 1 の参照ピクチャ P 1 に対する前方向の予測動きベクトル M V 1 F * を演算（算出）する（ステップ S 1 0 8）。同様に、予測演算部 1 2 3 は、動きベクトル検出部 1 2 1 によって検出された動きベクトル M V 0 F をスケールリングすることにより、対象ブロック B L 2 の参照ピクチャ P 1 に対する予測動きベクトル M V 2 * を演算（算出）する（ステップ S 1 1 0）。

【 0 0 8 9 】

探索範囲制御部 1 2 4 は、予測演算部 1 2 3 によって算出された対象ブロック B L 1 の予測動きベクトル M V 1 F * により示される参照ピクチャ P 1 の探索範囲を特定して動きベクトル検出部 1 2 1 に設定する（ステップ S 1 1 2）。同様に、探索範囲制御部 1 2 4 は、予測演算部 1 2 3 によって算出された対象ブロック B L 2 の予測動きベクトル M V 2 * により示される参照ピクチャ P 1 の探索範囲を特定して動きベクトル検出部 1 2 1 に設定する（ステップ S 1 1 4）。

【 0 0 9 0 】

動きベクトル検出部 1 2 1 は、対象ブロック B L 1 に対して探索範囲制御部 1 2 4 により設定された探索範囲から、その対象ブロック B L 1 に近似する画像を有するブロックを探索することにより、対象ブロック B L 1 の動きベクトル（前方動きベクトル）M V 1 F を検出する（ステップ S 1 1 6）。同様に、動きベクトル検出部 1 2 1 は、対象ブロック B L 2 に対して探索範囲制御部 1 2 4 により設定された探索範囲から、その対象ブロック B L 2 に近似する画像を有するブロックを探索することにより、対象ブロック B L 2 の動きベクトル M V 2 を検出する（ステップ S 1 1 8）。そして、動きベクトル検出部 1 2 1 は、検出した動きベクトル M V 1 F と動きベクトル M V 2 を記憶部 1 2 2 に格納する（ステップ S 1 2 0）。

【 0 0 9 1 】

ステップ S 1 0 0 ~ S 1 2 0 までの処理により、ブロック選択部 1 1 1 により抽出された全ての対象ブロック B L 0, B L 1, B L 2 の前方動きベクトルが検出される。

10

20

30

40

50

【 0 0 9 2 】

次に、ブロック選択部 1 1 1 は、改めて対象ブロックを抽出する。例えば、ブロック選択部 1 1 1 は、図 5 に示すように、ピクチャメモリ 1 1 2 にある対象ピクチャ B 2 , B 3 のそれぞれから、ブロック B L 0 , B L 1 を対象ブロックとして抽出する。そして、ブロック選択部 1 1 1 は、参照ピクチャ P 4 において対象ブロック B L 0 , B L 1 と同じ位置にあるブロック B L 2 を選択する。

【 0 0 9 3 】

予測演算部 1 2 3 は、ブロック選択部 1 1 1 によって選択された参照ピクチャ P 4 のブロック B L 2 の動きベクトル M V 2 を記憶部 1 2 2 から読み出す (ステップ S 1 2 4)。そして、予測演算部 1 2 3 は、時間ダイレクトモードにより、つまり、記憶部 1 2 2 から読み出した動きベクトル M V 2 をスケーリングすることにより、対象ブロック B L 0 の参照ピクチャ P 4 に対する後方向の予測動きベクトル M V 0 B * を演算 (算出) する (ステップ S 1 2 6)。同様に、予測演算部 1 2 3 は、記憶部 1 2 2 から読み出した動きベクトル M V 2 をスケーリングすることにより、対象ブロック B L 1 の参照ピクチャ P 4 に対する後方の予測動きベクトル M V 1 B * を演算 (算出) する (ステップ S 1 2 8)。

【 0 0 9 4 】

探索範囲制御部 1 2 4 は、予測演算部 1 2 3 によって算出された対象ブロック B L 0 の予測動きベクトル M V 0 B * により示される参照ピクチャ P 4 の探索範囲を特定して動きベクトル検出部 1 2 1 に設定する (ステップ S 1 3 0)。同様に、探索範囲制御部 1 2 4 は、予測演算部 1 2 3 によって算出された対象ブロック B L 1 の予測動きベクトル M V 1 B * により示される参照ピクチャ P 4 の探索範囲を特定して動きベクトル検出部 1 2 1 に設定する (ステップ S 1 3 2)。

【 0 0 9 5 】

動きベクトル検出部 1 2 1 は、対象ブロック B L 0 に対して探索範囲制御部 1 2 4 により設定された探索範囲から、その対象ブロック B L 0 に近似する画像を有するブロックを探索することにより、対象ブロック B L 0 の動きベクトル (後方動きベクトル) M V 0 B を検出する (ステップ S 1 3 4)。同様に、動きベクトル検出部 1 2 1 は、対象ブロック B L 1 に対して探索範囲制御部 1 2 4 により設定された探索範囲から、その対象ブロック B L 1 に近似する画像を有するブロックを探索することにより、対象ブロック B L 1 の動きベクトル (後方動きベクトル) M V 1 B を検出する (ステップ S 1 3 6)。そして、動きベクトル検出部 1 2 1 は、検出した動きベクトル M V 0 B と動きベクトル M V 1 B を記憶部 1 2 2 に格納する (ステップ S 1 3 8)。

【 0 0 9 6 】

ステップ S 1 2 2 ~ S 1 3 8 までの処理により、ブロック選択部 1 1 1 により抽出された全ての対象ブロック B L 0 , B L 1 の後方動きベクトルが検出される。

【 0 0 9 7 】

このように本実施の形態における動きベクトル検出装置 1 0 0 a では、対象ピクチャの対象ブロックの動きベクトルを検出するために、他の対象ピクチャの対象ブロックに対して検出された動きベクトルをスケーリングして予測動きベクトルを算出し、その予測動きベクトルの示す参照ピクチャにおける S 画素 × S ラインの範囲を探索範囲として特定する。したがって、対象ピクチャと参照ピクチャとの表示時刻の時間差に関わらず、常に一定の大きさの探索範囲が特定される。一方、従来では、その対象ピクチャが参照ピクチャから表示時間的に離れていれば、その時間差に応じて広い範囲が探索範囲とされている。したがって、本実施の形態では、対象ピクチャが参照ピクチャから表示時間的に離れていても、探索範囲を小さくすることができるとともに、動きベクトル M V 0 F のスケーリングにより適切な探索範囲を特定することができる。

【 0 0 9 8 】

また、本実施の形態では、テレスコピックサーチのように、隣接するピクチャ間の動きベクトルを求めていくことがないため、動きベクトルの検出精度の低下を防ぐことができる。さらに、本実施の形態では、テレスコピックサーチのように、直接必要のない動きベ

10

20

30

40

50

クトルまで検出する必要がない。その結果、全体的な探索範囲を小さくすることができる。さらに、本実施の形態では、対象ブロックをずらしながら各対象ブロックの動きベクトルを検出するときにも、テレスコピックサーチと比べて、参照ピクチャを格納しているピクチャメモリ120からの転送データ量を少なくすることができる。即ち、本実施の形態では、表示順で参照ピクチャと対象ピクチャとの間に他のピクチャがあっても、その他のピクチャにおける探索範囲のデータをピクチャメモリ120から読み出す必要がなく、さらに、対象ブロックをずらしながら各対象ブロックの動きベクトルを検出するときには、前回に読み出した探索範囲を有効に利用して、新たにピクチャメモリ120から読み出すデータ量を減らすことができる。

【0099】

10

したがって、本実施の形態における動きベクトル検出装置100aでは、探索範囲を適切な範囲とすることができるため、動きベクトルの検出のための演算量を低減することができるとともに、ピクチャメモリ120からの転送データ量を少なくすることができるため、回路規模の縮小化を図ることができる。

【0100】

モード選択部125は、上述のように対象ブロックの動きベクトルが検出されると、その検出された動きベクトルと、予測演算部123で算出された予測動きベクトルとを用いて、時間ダイレクトモードと非時間ダイレクトモードのうち何れかのモードを選択する。具体的に、モード選択部125は、このような選択を行なうときには、予測残差画像と、動きベクトルおよび予測動きベクトルと、対象ブロックの位置情報と、対象ピクチャの表示順序とを用いて、時間ダイレクトモードでの評価値と、非時間ダイレクトモードでの評価値とを算出する。そして、モード選択部125は、その評価値の比較結果に基づいてモードを選択する。

20

【0101】

図7は、モード選択部125が非時間ダイレクトモードの評価値を算出する動作を示すフローチャートである。

【0102】

まず、モード選択部125は対象ブロックを選択する(ステップS200)。例えば、モード選択部125は、図3および図5に示すピクチャB3のブロックBL1を選択する。モード選択部125は、ブロックBL1の前方動きベクトルMV1Fと後方動きベクトルMV1Bとを記憶部122から読み出す(ステップS202, S204)。

30

【0103】

そして、モード選択部125は、動き補償部126に対して、その前方動きベクトルMV1Fにより示される参照ピクチャP1のブロックを参照データとして抽出させるとともに(ステップS206)、後方動きベクトルMV1Bにより示される参照ピクチャP4のブロックを参照データとして抽出させる(ステップS208)。さらに、モード選択部125は、動き補償部126に対して、参照ピクチャP1の参照データと参照ピクチャP4の参照データとの平均を算出させて平均化参照データを生成させる(ステップS210)。

【0104】

40

また、モード選択部125は、ステップS202で読み出した前方動きベクトルMV1Fの符号量を符号列生成部117に算出させるとともに(ステップS212)、ステップS204で読み出した後方動きベクトルMV1Bの符号量を符号列生成部117に算出させる(ステップS214)。

【0105】

また、モード選択部125は、ステップS200で選択したブロックBL1を、ピクチャメモリ112からブロック選択部111を介して、入力データとして抽出する(ステップS216)。そして、モード選択部125は、ステップS210で生成された平均化参照データと、ステップS216で抽出された入力データとの絶対値差分和を算出する(ステップS218)。

50

【 0 1 0 6 】

モード選択部 1 2 5 は、ステップ S 2 1 8 で算出された絶対値差分和と、ステップ S 2 1 2 , S 2 1 4 で算出された符号量とに基づいて、非時間ダイレクトモードの評価値を算出する (ステップ S 2 2 0)。

【 0 1 0 7 】

図 8 は、モード選択部 1 2 5 が時間ダイレクトモードの評価値を算出する動作を示すフローチャートである。

【 0 1 0 8 】

まず、モード選択部 1 2 5 は対象ブロックを選択する (ステップ S 3 0 0)。例えば、モード選択部 1 2 5 は、図 3 および図 5 に示すピクチャ B 3 のブロック B L 1 を選択する。そして、モード選択部 1 2 5 は、時間ダイレクトモードが選択されたことを通知するためのダイレクトモード選択情報を生成する (ステップ S 3 0 2)。

【 0 1 0 9 】

次に、モード選択部 1 2 5 は、例えば動きベクトル M V 2 のような時間ダイレクトモードに必要な動きベクトルを記憶部 1 2 2 から読み出す (ステップ S 3 0 4)。そして、モード選択部 1 2 5 は、予測演算部 1 2 3 に対して予測動きベクトルを算出させる (ステップ S 3 0 6 , S 3 0 8)。即ち、予測演算部 1 2 3 は、ステップ S 3 0 4 で読み出された動きベクトルをスケーリングすることにより、例えば、ステップ S 3 0 0 で選択されたブロック B L 1 の前方向の予測動きベクトル M V 1 F * と後方向の予測動きベクトル M V 1 B * とを算出する。

【 0 1 1 0 】

そして、モード選択部 1 2 5 は、動き補償部 1 2 6 に対して、その前方向の予測動きベクトル M V 1 F * により示される参照ピクチャ P 1 のブロックを参照データとして抽出させるとともに (ステップ S 3 1 0)、その後方向の予測動きベクトル M V 1 B * により示される参照ピクチャ P 4 のブロックを参照データとして抽出させる (ステップ S 3 1 2)。さらに、モード選択部 1 2 5 は、動き補償部 1 2 6 に対して、参照ピクチャ P 1 の参照データと参照ピクチャ P 4 の参照データとの平均を算出させて平均化参照データを生成させる (ステップ S 3 1 4)。

【 0 1 1 1 】

また、モード選択部 1 2 5 は、ステップ S 3 0 0 で選択したブロック B L 1 を、ピクチャメモリ 1 1 2 からブロック選択部 1 1 1 を介して入力データとして抽出する (ステップ S 3 1 6)。そして、モード選択部 1 2 5 は、ステップ S 3 1 4 で生成された平均化参照データと、ステップ S 3 1 6 で抽出された入力データとの絶対値差分和を算出する (S 3 1 8)。

【 0 1 1 2 】

モード選択部 1 2 5 は、ステップ S 3 1 8 で算出された絶対値差分和と、ステップ S 3 0 2 で生成されたダイレクトモード選択情報とに基づいて、時間ダイレクトモードの評価値を算出する (ステップ S 3 2 0)。

【 0 1 1 3 】

このように、モード選択部 1 2 5 は、ブロック B L 1 に対する非時間ダイレクトモードの評価値と時間ダイレクトモードの評価値とを算出すると、それらの評価値を比較することにより、全体的な符号量が小さくなるような評価値の高いモードを選択する。

【 0 1 1 4 】

そして、モード選択部 1 2 5 は、非時間ダイレクトモードを選択すると、動きベクトル検出部 1 2 1 によって検出されて記憶部 1 2 2 に格納されている対象ブロックの動きベクトルを符号列生成部 1 1 7 に出力する。その結果、符号列生成部 1 1 7 は、その動きベクトルを可変長符号化して符号列中に含める。一方、モード選択部 1 2 5 は、時間ダイレクトモードを選択すると、ダイレクトモード選択情報を符号列生成部 1 1 7 に出力する。その結果、符号列生成部 1 1 7 は、符号列中にダイレクトモード選択情報を含める。

【 0 1 1 5 】

10

20

30

40

50

このように本実施の形態では、評価値に基づいて時間ダイレクトモードと非時間ダイレクトモードとを切り換えるため、符号列の符号量を常に少なくすることができる。また、本実施の形態では、上述のようにテレスコピックサーチの場合と比べて動きベクトルの検出精度が高いため、時間ダイレクトモードが選択される可能性が高くなり、符号量をさらに少なくすることができる。

【0116】

(変形例1)

ここで、本実施の形態における予測動きベクトルの算出方法についての第1の変形例を説明する。

【0117】

上記実施の形態では、複数の対象ピクチャのうち、表示順序が最も早いピクチャに含まれる対象ブロックの動きベクトルをスケーリングした。つまり、図3に示すように、動きベクトル検出装置100aは、表示順序が最も早いピクチャB2に含まれる対象ブロックBL0の動きベクトルMV0Fを検出し、その動きベクトルMV0Fをスケーリングする。その結果、動きベクトル検出装置100aは、ピクチャB3に含まれる対象ブロックBL1の前方向の予測動きベクトルMV1F*と、ピクチャP4に含まれる対象ブロックBL2の前方向の予測動きベクトルMV2*とを算出する。

【0118】

本変形例では、複数の対象ピクチャのうち、表示順序が中間にあるピクチャの対象ブロックの動きベクトルをスケーリングすることにより、そのピクチャの前後にあるピクチャに含まれる各対象ブロックの予測動きベクトルを算出する。

【0119】

図9は、変形例1における予測動きベクトルの算出方法を示す図である。

まず、動きベクトル検出部121は、複数の対象ピクチャのうち、表示順序が中間にあるピクチャB3に含まれる対象ブロックBL1の動きベクトルMV1Fを検出する。そして、動きベクトル検出部121はその動きベクトルMV1Fを記憶部122に格納する。

【0120】

このとき、例えば、ピクチャB2のブロックの参照ピクチャP1に対する動きベクトルの探索範囲はS画素×Sラインであって、参照ピクチャP1とピクチャB3との間の表示時間間隔は、参照ピクチャP1とピクチャB2との間の表示時間間隔の2倍である。このような場合、対象ブロックBL1の動きベクトルMV1Fを検出するための参照ピクチャP1上における探索範囲は、(2×S)画素×(2×S)ラインとなる。

【0121】

予測演算部123は、動きベクトルMV1Fを記憶部122から読み出して、動きベクトルMV1Fをスケーリングすることにより、ピクチャB2およびピクチャP4において対象ブロックBL1と同じ位置にある対象ブロックBL0、BL2に対して、予測動きベクトルMV0F*と予測動きベクトルMV2*とを算出する。

【0122】

具体的に、予測演算部123は、ピクチャP1、B2、B3、P4の表示順情報T1、T2、T3、T4を使い、下記の(式4a)および(式4b)によって、ピクチャB2の対象ブロックBL0の参照ピクチャP1に対する予測動きベクトルMV0F*と、ピクチャP4の対象ブロックBL2の参照ピクチャP1に対する予測動きベクトルMV2*とを算出する。なお、(式4a)および(式4b)中、Mv0f*およびMv2*はそれぞれ予測動きベクトルMV0F*、MV2*の大きさを示し、Mv1fは動きベクトルMV1Fの大きさを示す。

【0123】

$$Mv0f* = (T2 - T1) / (T3 - T1) \times Mv1f \quad \dots (式4a)$$

$$Mv2* = (T4 - T1) / (T3 - T1) \times Mv1f \quad \dots (式4b)$$

【0124】

探索範囲制御部124は、予測動きベクトルMV0F*により示される参照ピクチャP

10

20

30

40

50

1 の位置を中心とする S 画素 \times S ラインの探索範囲を特定して動きベクトル検出部 121 に設定するとともに、予測動きベクトル $MV2^*$ により示される参照ピクチャ P1 の位置を中心とする S 画素 \times S ラインの探索範囲を特定して動きベクトル検出部 121 に設定する。

【0125】

動きベクトル検出部 121 は、探索範囲制御部 124 によって設定された参照ピクチャ P1 の探索範囲から、対象ブロック BL0 の前方動きベクトルと対象ブロック BL2 の動きベクトルとを検出する。つまり、動きベクトル検出部 121 は、参照ピクチャ P1 に対する対象ブロック BL1 の前方動きベクトルを検出した後、参照ピクチャ P1 に対する対象ブロック BL0 の前方動きベクトルを検出し、その次に、参照ピクチャ P1 に対する対象ブロック BL2 の動きベクトルを検出する。そして、動きベクトル検出部 121 は、対象ブロック BL0 および対象ブロック BL2 のそれぞれの動きベクトルを記憶部 122 に格納する。

10

【0126】

このように本変形例では、表示順序が中間にあるピクチャの対象ブロックの動きベクトルをスケールリングすることによって、そのピクチャの前後にあるピクチャに含まれる各対象ブロックの予測動きベクトルを高い精度で算出することができる。

【0127】

(変形例 2)

ここで、本実施の形態における予測動きベクトルの算出方法についての第 2 の変形例を説明する。

20

【0128】

上記実施の形態では、複数の対象ピクチャのうち、表示順序が最も早いピクチャに含まれる対象ブロックの動きベクトルをスケールリングした。

【0129】

本変形例では、複数の対象ピクチャのうち、表示順序が最も遅いピクチャの対象ブロックの動きベクトルをスケールリングすることにより、そのピクチャの前方にある対象ピクチャに含まれる各対象ブロックの予測動きベクトルを算出する。

【0130】

図 10 は、変形例 2 における予測動きベクトルの算出方法を示す図である。

30

まず、動きベクトル検出部 121 は、複数の対象ピクチャのうち、表示順序が最も遅いピクチャ P4 に含まれる対象ブロック BL2 の動きベクトル $MV2$ を検出する。そして、動きベクトル検出部 121 はその動きベクトル $MV2$ を記憶部 122 に格納する。

【0131】

このとき、例えば、ピクチャ B2 のブロックの参照ピクチャ P1 に対する動きベクトルの探索範囲は S 画素 \times S ラインであって、参照ピクチャ P1 とピクチャ P4 との間の表示時間間隔は、参照ピクチャ P1 とピクチャ B2 との間の表示時間間隔の 3 倍である。このような場合、対象ブロック BL2 の動きベクトル $MV2$ を検出するための参照ピクチャ P1 上における探索範囲は、 $(3 \times S)$ 画素 \times $(3 \times S)$ ラインとなる。

【0132】

40

予測演算部 123 は、動きベクトル $MV2$ を記憶部 122 から読み出して、動きベクトル $MV2$ をスケールリングすることにより、ピクチャ B2 およびピクチャ B3 において対象ブロック BL2 と同じ位置にある対象ブロック BL0, BL1 に対して、予測動きベクトル $MV0F^*$ と予測動きベクトル $MV1F^*$ とを算出する。

【0133】

具体的に、予測演算部 123 は、ピクチャ P1, B2, B3, P4 の表示順情報 T1, T2, T3, T4 を使い、下記の (式 5a) および (式 5b) によって、ピクチャ B2 の対象ブロック BL0 の参照ピクチャ P1 に対する予測動きベクトル $MV0F^*$ と、ピクチャ B3 の対象ブロック BL1 の参照ピクチャ P1 に対する予測動きベクトル $MV1F^*$ とを算出する。なお、(式 5a) および (式 5b) 中、 $Mv0f^*$ および $Mv1f^*$ はそれ

50

ぞれ予測動きベクトル $MV0F^*$, $MV1F^*$ の大きさを示し、 $Mv2$ は動きベクトル $MV2$ の大きさを示す。

【0134】

$$Mv0f^* = (T2 - T1) / (T4 - T1) \times Mv2 \quad \dots (式5a)$$

$$Mv1f^* = (T3 - T1) / (T4 - T1) \times Mv2 \quad \dots (式5b)$$

【0135】

探索範囲制御部124は、予測動きベクトル $MV0F^*$ により示される参照ピクチャ $P1$ の位置を中心とする S 画素 $\times S$ ラインの探索範囲を特定して動きベクトル検出部121に設定するとともに、予測動きベクトル $MV1F^*$ により示される参照ピクチャ $P1$ の位置を中心とする S 画素 $\times S$ ラインの探索範囲を特定して動きベクトル検出部121に設定する。

10

【0136】

動きベクトル検出部121は、探索範囲制御部124によって設定された参照ピクチャ $P1$ の探索範囲から、対象ブロック $BL0$ の前方動きベクトルと対象ブロック $BL1$ の前方動きベクトルとを検出する。つまり、動きベクトル検出部121は、参照ピクチャ $P1$ に対する対象ブロック $BL2$ の動きベクトル $MV2$ を検出した後、参照ピクチャ $P1$ に対する対象ブロック $BL0$ の前方動きベクトルを検出し、その次に、参照ピクチャ $P1$ に対する対象ブロック $BL1$ の前方動きベクトルを検出する。そして、動きベクトル検出部121は、対象ブロック $BL0$ および対象ブロック $BL1$ のそれぞれの前方動きベクトルを記憶部122に格納する。

20

【0137】

(変形例3)

ここで、本実施の形態における予測動きベクトルの算出方法についての第3の変形例を説明する。

【0138】

上記実施の形態では、後方向の予測動きベクトルを算出するときには、 P ピクチャの動きベクトルをスケーリングした。つまり、図5に示すように、動きベクトル検出装置100aは、 P ピクチャである参照ピクチャ $P4$ に含まれるブロック $BL2$ の動きベクトル $MV2$ を検出し、その動きベクトル $MV2$ をスケーリングする。その結果、動きベクトル検出装置100aは、ピクチャ $B2$ に含まれる対象ブロック $BL0$ の後方向の予測動きベクトル $MV0B^*$ と、ピクチャ $B3$ に含まれる対象ブロック $BL1$ の後方向の予測動きベクトル $MV1B^*$ とを算出する。

30

【0139】

本変形例では、後方向の予測動きベクトルを算出するときには、対象ピクチャのうち、表示順序で参照ピクチャに最も近い B ピクチャの後方動きベクトルをスケーリングすることにより、他の対象ピクチャである B ピクチャの後方向の予測動きベクトルを算出する。

【0140】

図11は、変形例3における予測動きベクトルの算出方法を示す図である。

まず、動きベクトル検出部121は、複数の対象ピクチャのうち、表示順序で参照ピクチャ $P4$ に最も近いピクチャ $B3$ に含まれる対象ブロック $BL1$ の動きベクトル(後方動きベクトル) $MV1B$ を検出する。そして、動きベクトル検出部121は、その動きベクトル $MV1B$ を記憶部122に格納する。

40

【0141】

予測演算部123は、動きベクトル $MV1B$ を記憶部122から読み出して、動きベクトル $MV1B$ をスケーリングすることにより、ピクチャ $B2$ において対象ブロック $BL1$ と同じ位置にある対象ブロック $BL0$ に対して予測動きベクトル $MV0B^*$ を算出する。

【0142】

具体的に、予測演算部123は、ピクチャ $P1$, $B2$, $B3$, $P4$ の表示順情報 $T1$, $T2$, $T3$, $T4$ を使い、下記の(式6)によって、ピクチャ $B2$ の参照ピクチャ $P4$ に

50

対する予測動きベクトル $MV0B^*$ を算出する。なお、(式6)中、 $Mv0b^*$ は予測動きベクトル $MV0B^*$ の大きさを示し、 $Mv1b$ は動きベクトル $MV1B$ の大きさを示す。

【0143】

$$Mv0b^* = (T4 - T2) / (T4 - T3) \times Mv1b \quad \dots (式6)$$

探索範囲制御部124は、予測動きベクトル $MV0B^*$ により示される参照ピクチャP4の位置を中心とするS画素×Sラインの探索範囲を特定して動きベクトル検出部121に設定する。

【0144】

動きベクトル検出部121は、探索範囲制御部124によって設定された参照ピクチャP4の探索範囲から、対象ブロックBL0の後方動きベクトルを検出する。つまり、動きベクトル検出部121は、参照ピクチャP4に対する対象ブロックBL1の後方動きベクトルを検出した後、参照ピクチャP4に対する対象ブロックBL0の後方動きベクトルを検出する。そして、動きベクトル検出部121は、対象ブロックBL0の後方動きベクトル $MV0B$ を記憶部122に格納する。

10

【0145】

このように本変形例では、表示順序で参照ピクチャに最も近いBピクチャの後方動きベクトルをスケールリングすることによって、そのBピクチャの隣のBピクチャに含まれる対象ブロックの予測動きベクトルを高い精度で算出することができる。

【0146】

(変形例4)

ここで、本実施の形態における予測動きベクトルの算出方法についての第4の変形例を説明する。

20

【0147】

上記実施の形態では、後方向の予測動きベクトルを算出するときには、Pピクチャの動きベクトルをスケールリングした。

【0148】

本変形例では、後方向の予測動きベクトルを算出するときには、対象ピクチャのうち、表示順序で参照ピクチャから最も遠いBピクチャの後方動きベクトルをスケールリングすることにより、他の対象ピクチャであるBピクチャの後方向の予測動きベクトルを算出する。

30

【0149】

図12は、変形例4における予測動きベクトルの算出方法を示す図である。

まず、動きベクトル検出部121は、複数の対象ピクチャのうち、表示順序で参照ピクチャP4から最も遠いピクチャB2に含まれる対象ブロックBL0の動きベクトル(後方動きベクトル) $MV0B$ を検出する。そして、動きベクトル検出部121は、その動きベクトル $MV0B$ を記憶部122に格納する。

【0150】

このとき、例えば、ピクチャB3のブロックの参照ピクチャP4に対する動きベクトルの探索範囲はS画素×Sラインであって、参照ピクチャP4とピクチャB2との間の表示時間間隔は、参照ピクチャP4とピクチャB3との間の表示時間間隔の2倍である。このような場合、対象ブロックBL0の動きベクトル $MV0B$ を検出するための参照ピクチャP4上における探索範囲は、(2×S)画素×(2×S)ラインとなる。

40

【0151】

予測演算部123は、動きベクトル $MV0B$ を記憶部122から読み出して、動きベクトル $MV0B$ をスケールリングすることにより、ピクチャB3において対象ブロックBL0と同じ位置にある対象ブロックBL1に対して、予測動きベクトル $MV1B^*$ を算出する。

【0152】

具体的に、予測演算部123は、ピクチャP1, B2, B3, P4の表示順情報T1,

50

T2, T3, T4を使い、下記の(式7)によって、ピクチャB3の対象ブロックBL1の参照ピクチャP4に対する予測動きベクトルMV1B*を算出する。なお、(式7)中、MV1b*は予測動きベクトルMV1B*の大きさを示し、MV0bは動きベクトルMV0Bの大きさを示す。

【0153】

$$MV1b* = (T4 - T3) / (T4 - T2) \times MV0b \quad \dots (式7)$$

探索範囲制御部124は、予測動きベクトルMV1B*により示される参照ピクチャP4の位置を中心とするS画素×Sラインの探索範囲を特定して動きベクトル検出部121に設定する。

【0154】

動きベクトル検出部121は、探索範囲制御部124によって設定された参照ピクチャP4の探索範囲から、対象ブロックBL1の後方動きベクトルを検出する。つまり、動きベクトル検出部121は、参照ピクチャP4に対する対象ブロックBL0の後方動きベクトルを検出した後、参照ピクチャP4に対する対象ブロックBL1の後方動きベクトルを検出する。そして、動きベクトル検出部121は、対象ブロックBL1の後方動きベクトルを記憶部122に格納する。

【0155】

(変形例5)

ここで、本実施の形態における予測動きベクトルの算出方法についての第5の変形例を説明する。

【0156】

上記実施の形態では、複数の対象ピクチャのうち、1枚のピクチャの動きベクトルを検出して、その動きベクトルをスケーリングすることにより、他の対象ピクチャの予測動きベクトルを算出した。つまり、図3に示すように、動きベクトル検出装置100aは、複数の対象ピクチャのうち、表示順序が最も早いピクチャB2に含まれる対象ブロックBL0の動きベクトルMV0Fを検出し、その動きベクトルMV0Fをスケーリングする。その結果、動きベクトル検出装置100aは、ピクチャB3に含まれる対象ブロックBL1の前方向の予測動きベクトルMV1F*と、ピクチャP4に含まれる対象ブロックBL2の前方向の予測動きベクトルMV2*とを算出する。

【0157】

本変形例では、複数の対象ピクチャの動きベクトルを検出し、それらのピクチャのうち、予測動きベクトルの算出対象となるピクチャに最も近いピクチャの動きベクトルをスケーリングする。これにより、予測動きベクトルが算出される。即ち、本変形例では、複数の第1のピクチャのうち、第2のピクチャとの間の表示時刻の時間差が最も短いピクチャの第1の動きベクトルに基づいて、参照ピクチャの探索範囲を特定する。

【0158】

図13は、変形例5における予測動きベクトルの算出方法を示す図である。

まず、動きベクトル検出部121は、対象ピクチャのうち、表示順序で参照ピクチャP1に最も近いピクチャB2に含まれる対象ブロックBL0の動きベクトル(前方動きベクトル)MV0Fを検出する。そして、動きベクトル検出部121は、その動きベクトルMV0Fを記憶部122に格納する。さらに、動きベクトル検出部121は、対象ピクチャのうち、ピクチャB2の次に参照ピクチャP1に近いピクチャB3に含まれる対象ブロックBL1の動きベクトル(前方動きベクトル)MV1Fを検出する。そして、動きベクトル検出部121は、その動きベクトルMV1Fを記憶部122に格納する。

【0159】

このとき、例えば、ピクチャB2のブロックの参照ピクチャP1に対する動きベクトルの探索範囲はS画素×Sラインであって、参照ピクチャP1とピクチャB3との間の表示時間間隔は、参照ピクチャP1とピクチャB2との間の表示時間間隔の2倍である。このような場合、対象ブロックBL1の動きベクトルMV1Fを検出するための参照ピクチャP1上における探索範囲は、(2×S)画素×(2×S)ラインとなる。

10

20

30

40

50

【 0 1 6 0 】

予測演算部 1 2 3 は、記憶部 1 2 2 に格納されている動きベクトル MV_0F と動きベクトル MV_1F のうち、ピクチャ P_4 に最も近いピクチャ B_3 の動きベクトル MV_1F を読み出す。そして、予測演算部 1 2 3 は、動きベクトル MV_1F をスケーリングすることにより、ピクチャ P_4 において対象ブロック BL_0 、 BL_1 と同じ位置にある対象ブロック BL_2 に対して、予測動きベクトル MV_2^* を算出する。

【 0 1 6 1 】

探索範囲制御部 1 2 4 は、予測動きベクトル MV_2^* により示される参照ピクチャ P_1 の位置を中心とする S 画素 $\times S$ ラインの探索範囲を特定して動きベクトル検出部 1 2 1 に設定する。

10

【 0 1 6 2 】

動きベクトル検出部 1 2 1 は、探索範囲制御部 1 2 4 によって設定された参照ピクチャ P_1 の探索範囲から、対象ブロック BL_2 の動きベクトルを検出する。つまり、動きベクトル検出部 1 2 1 は、参照ピクチャ P_1 に対する対象ブロック BL_0 の前方動きベクトルを検出した後、参照ピクチャ P_1 に対する対象ブロック BL_1 の前方動きベクトルを検出し、その次に、参照ピクチャ P_1 に対する対象ブロック BL_2 の動きベクトルを検出する。そして、動きベクトル検出部 1 2 1 は、対象ブロック BL_2 の動きベクトルを記憶部 1 2 2 に格納する。

【 0 1 6 3 】

(変形例 6)

ここで、本実施の形態における予測動きベクトルの算出方法についての第 6 の変形例を説明する。

20

【 0 1 6 4 】

上記実施の形態では、複数の対象ピクチャのうち、1 枚のピクチャの動きベクトルを検出して、その動きベクトルをスケーリングすることにより、他の対象ピクチャの予測動きベクトルを算出した。

【 0 1 6 5 】

本変形例では、複数の対象ピクチャの動きベクトルを検出する。そして、それらの動きベクトルの向きが等しいときには、何れか一方の動きベクトルをスケーリングすることにより、他の対象ピクチャの予測動きベクトルを算出する。

30

【 0 1 6 6 】

図 1 4 は、変形例 6 における予測動きベクトルの算出方法を示す図である。

まず、動きベクトル検出部 1 2 1 は、複数の対象ピクチャのうち、表示順序で参照ピクチャ P_1 に最も近いピクチャ B_2 に含まれる対象ブロック BL_0 の動きベクトル (前方動きベクトル) MV_0F を検出する。

【 0 1 6 7 】

さらに、動きベクトル検出部 1 2 1 は、複数の対象ピクチャのうち、表示順序で参照ピクチャ P_1 に最も遠いピクチャ P_4 に含まれる対象ブロック BL_2 の動きベクトル MV_2 を検出する。そして、動きベクトル検出部 1 2 1 は、それらの動きベクトル MV_0F 、 MV_2 を記憶部 1 2 2 に格納する。

40

【 0 1 6 8 】

このとき、例えば、ピクチャ B_2 のブロックの参照ピクチャ P_1 に対する動きベクトルの探索範囲は S 画素 $\times S$ ラインであって、参照ピクチャ P_1 とピクチャ P_4 との間の表示時間間隔は、参照ピクチャ P_1 とピクチャ B_2 との間の表示時間間隔の 3 倍である。このような場合、対象ブロック BL_2 の動きベクトル MV_2 を検出するための参照ピクチャ P_1 上における探索範囲は、 $(3 \times S)$ 画素 $\times (3 \times S)$ ラインとなる。

【 0 1 6 9 】

予測演算部 1 2 3 は、記憶部 1 2 2 に格納されている動きベクトル MV_0F と動きベクトル MV_2 を読み出して比較する。そして、予測演算部 1 2 3 は、これらの動きベクトルの向きが等しいときには、動きベクトル MV_0F と動きベクトル MV_2 の何れか一方をス

50

ケーリングすることにより、ピクチャ B 3 において対象ブロック B L 0 , B L 2 と同じ位置にある対象ブロック B L 1 に対して、予測動きベクトル M V 1 F * を算出する。

【 0 1 7 0 】

探索範囲制御部 1 2 4 は、予測動きベクトル M V 1 F * により示される参照ピクチャ P 1 の位置を中心とする S 画素 × S ラインの探索範囲を特定して動きベクトル検出部 1 2 1 に設定する。

【 0 1 7 1 】

動きベクトル検出部 1 2 1 は、探索範囲制御部 1 2 4 によって設定された参照ピクチャ P 1 の探索範囲から、対象ブロック B L 1 の動きベクトルを検出する。つまり、動きベクトル検出部 1 2 1 は、参照ピクチャ P 1 に対する対象ブロック B L 0 の前方動きベクトルを検出して、さらに、参照ピクチャ P 1 に対する対象ブロック B L 2 の動きベクトルを検出した後に、参照ピクチャ P 1 に対する対象ブロック B L 1 の前方動きベクトルを検出する。そして、動きベクトル検出部 1 2 1 は、対象ブロック B L 1 の動きベクトルを記憶部 1 2 2 に格納する。

10

【 0 1 7 2 】

このように本変形例では、複数のピクチャのそれぞれの動きベクトルの向きが等しい場合、それらのうち何れか一方の動きベクトルをスケーリングして予測動きベクトルを算出することによって、その予測動きベクトルの精度を高めることができる。

【 0 1 7 3 】

なお、このように算出される予測動きベクトルの精度は高いため、その予測動きベクトルにより示される探索範囲から動きベクトルを検出することを省いてもよい。即ち、その予測動きベクトルを実際の動きベクトルと同様に扱ってもよい。

20

【 0 1 7 4 】

(変形例 7)

ここで、本実施の形態における予測動きベクトルの算出方法についての第 7 の変形例を説明する。

【 0 1 7 5 】

上記実施の形態では、対象ブロックの参照ピクチャに対する予測動きベクトルを算出するときには、その対象ブロックとは異なる他のブロックの動きベクトルをスケーリングした。

30

【 0 1 7 6 】

本変形例では、対象ブロックの参照ピクチャに対する予測動きベクトルを算出するときには、その対象ブロックの他の参照ピクチャに対する動きベクトルをスケーリングする。

【 0 1 7 7 】

図 1 5 は、変形例 7 における予測動きベクトルの算出方法を示す図である。

まず、動きベクトル検出部 1 2 1 は、B ピクチャであるピクチャ B 5 に含まれる対象ブロック B L 3 の参照ピクチャ P 4 に対する動きベクトル M V 3 F を検出する。

【 0 1 7 8 】

予測演算部 1 2 3 は、その動きベクトル検出部 1 2 1 によって検出された動きベクトル M V 3 F をスケーリングすることにより、対象ブロック B L 3 の参照ピクチャ P 1 に対する予測動きベクトル M V 3 F * を算出する。

40

【 0 1 7 9 】

探索範囲制御部 1 2 4 は、予測動きベクトル M V 3 F * により示される参照ピクチャ P 1 の位置を中心とする S 画素 × S ラインの探索範囲を特定して動きベクトル検出部 1 2 1 に設定する。

【 0 1 8 0 】

動きベクトル検出部 1 2 1 は、探索範囲制御部 1 2 4 によって設定された参照ピクチャ P 1 の探索範囲から、対象ブロック B L 3 の参照ピクチャ P 1 に対する動きベクトルを検出する。そして、動きベクトル検出部 1 2 1 は、その対象ブロック B L 3 の参照ピクチャ P 1 に対する動きベクトルを記憶部 1 2 2 に格納する。

50

【0181】

このように本変形例は、複数のピクチャを参照することが可能なH.264で有効である。

【0182】

(変形例8)

ここで、本実施の形態における探索範囲についての第8の変形例を説明する。

【0183】

上記実施の形態では、予測動きベクトルの示す位置を中心とする参照ピクチャの探索範囲だけから探索を行って動きベクトルを検出した。つまり、図3に示すように、動きベクトル検出装置100aは、ピクチャB3に含まれる対象ブロックBL1の前方動きベクトルを検出するときには、その対象ブロックBL1の予測動きベクトルMV1F*の示す位置を中心とする参照ピクチャP1の探索範囲(S画素×Sライン)だけから、対象ブロックBL1に近似する画像を有するブロックを探索する。

10

【0184】

本変形例では、予測動きベクトルの示す位置を中心とする参照ピクチャの探索範囲に、参照ピクチャにおいて対象ブロックと空間的同一位置にあるブロックも含める。

【0185】

図16は、変形例8における探索範囲を示す図である。

本変形例に係る探索範囲制御部124は、予測演算部123によって算出された予測動きベクトルMV1F*により示される位置を中心とする参照ピクチャP1の探索範囲(S画素×Sライン)に、対象ブロックBL1と同位置にある参照ピクチャP1のブロックBLを含める。また、探索範囲制御部124は、予測演算部123によって算出された予測動きベクトルMV2*により示される位置を中心とする参照ピクチャP1の探索範囲(S画素×Sライン)に、対象ブロックBL2と同位置にある参照ピクチャP1のブロックBLを含める。探索範囲制御部124は、このようなブロックBLを含む探索範囲を特定して動きベクトル検出部121に設定する。

20

【0186】

動きベクトル検出部121は、参照ピクチャP1のブロックBLを含む各探索範囲からそれぞれ、対象ブロックBL1の前方動きベクトルと対象ブロックBL2の前方動きベクトルとを検出する。

30

【0187】

このように本変形例では、探索範囲を広げることにより、S画素×Sラインの範囲からでは検出できない適切な動きベクトルを検出することができる。

【0188】

(変形例9)

ここで、本実施の形態における予測動きベクトルの算出方法についての第9の変形例を説明する。

【0189】

上記実施の形態では、Bピクチャについて前方向の予測動きベクトルを算出した後に、後方向の予測動きベクトルを算出した。

40

【0190】

本変形例では、Bピクチャの前方向の予測動きベクトルと後方向の予測動きベクトルとを、同一の動きベクトルをスケールリングすることにより同時に算出する。また、本変形例では、予測動きベクトルを算出する前に、予め2枚のPピクチャを符号化および復号化しておく。

【0191】

図17は、変形例9において予測動きベクトルを算出する前の各ピクチャの状態を示す図である。

【0192】

本変形例では、動画像符号化装置100は、まず、2枚のPピクチャを符号化するとと

50

もに復号化し、それぞれのピクチャを参照ピクチャとしてピクチャメモリ120に格納しておく。

【0193】

図18は、変形例9における予測動きベクトルの算出方法を示す図である。

上述のように本変形例では、動画像符号化装置100は、まず、ピクチャP1とピクチャP4とを符号化するとともに復号化し、それぞれのピクチャを参照ピクチャとしてピクチャメモリ120に格納しておく。したがって、このような符号化および復号化が行われた後には、記憶部122に、参照ピクチャP4の各ブロックの動きベクトルが格納されている。

【0194】

予測演算部123は、対象ピクチャB2, B3, P7のそれぞれに含まれる同一位置の対象ブロックBL0, BL1, BL5の予測動きベクトルを算出するために、それらの対象ブロックと同一位置にある参照ピクチャP4のブロックBL2の動きベクトルMV2を記憶部122から読み出す。

【0195】

そして、予測演算部123は、その動きベクトルMV2をスケーリングすることにより、ピクチャB2に含まれる対象ブロックBL0の、参照ピクチャP1に対する前方向の予測動きベクトルMV0F*と、参照ピクチャP4に対する後方向の予測動きベクトルMV0B*とを算出する。さらに、予測演算部123は、動きベクトルMV2をスケーリングすることにより、ピクチャB3に含まれる対象ブロックBL1の、参照ピクチャP1に対する前方向の予測動きベクトルMV1F*と、参照ピクチャP4に対する後方向の予測動きベクトルMV1B*とを算出する。さらに、予測演算部123は、動きベクトルMV2をスケーリングすることにより、ピクチャP7に含まれる対象ブロックBL5の参照ピクチャP4に対する予測動きベクトルMV5*を算出する。

【0196】

探索範囲制御部124は、予測演算部123によって算出された予測動きベクトルごとに、その予測動きベクトルにより示される位置を中心とする探索範囲(S画素×Sライン)を特定して動きベクトル検出部121に設定する。

【0197】

動きベクトル検出部121は、探索範囲制御部124によって設定された探索範囲ごとに、その探索範囲から動きベクトルを検出する。即ち、動きベクトル検出部121は、参照ピクチャP1の探索範囲から、ピクチャB2の対象ブロックBL0の前方動きベクトルを検出し、参照ピクチャP4の探索範囲から、ピクチャB2の対象ブロックBL0の後方動きベクトルを検出する。その後、動きベクトル検出部121は、参照ピクチャP1の探索範囲から、ピクチャB3の対象ブロックBL1の前方動きベクトルを検出し、参照ピクチャP4の探索範囲から、ピクチャB3の対象ブロックBL1の後方動きベクトルを検出する。さらにその後、動きベクトル検出部121は、参照ピクチャP4の探索範囲から、ピクチャP7の対象ブロックBL5の動きベクトルを検出する。そして、動きベクトル検出部121は、このように検出した動きベクトルを記憶部122に順次格納する。

【0198】

動きベクトル検出装置100aは、対象ピクチャB2, B3, P7のそれぞれに含まれる同一位置の対象ブロックをずらしながら、このような動きベクトルの検出を各対象ブロックに対して行う。そして、動画像符号化装置100は、このように検出された動きベクトルを用いてピクチャB2, B3, P7を符号化するとともに、ピクチャP7を復号化して参照ピクチャとしてピクチャメモリ120に格納する。したがって、このような符号化および復号化が行われた後には、記憶部122に、参照ピクチャP7の各ブロックの動きベクトルが格納されている。

【0199】

動きベクトル検出装置100aは、対象ピクチャB2, B3, P7を対象ピクチャB5, B6, P10に換えて、上述のような動きベクトルの検出を、その対象ピクチャB5,

10

20

30

40

50

B 6 , P 1 0 に対しても繰り返し行なう。

【 0 2 0 0 】

(変形例 1 0)

ここで、本実施の形態における動きベクトルの検出方法についての第 1 0 の変形例を説明する。

【 0 2 0 1 】

本変形例では、動画像符号化装置 1 0 0 は、動きベクトルの検出と、予測動きベクトルの算出および探索範囲の特定と、符号化および復号化処理とをパイプライン処理により行なう。

【 0 2 0 2 】

図 1 9 は、変形例 1 0 におけるパイプライン処理を示す図である。

まず、動きベクトル検出部 1 2 1 は、ピクチャ B 2 の (N - 1) 番目のブロック B L 0 の動きベクトル M V 0 F を検出する (ステップ S 5 0 0) 。

【 0 2 0 3 】

その後、予測演算部 1 2 3 は、その動きベクトル M V 0 F を用いて、ピクチャ B 3 , P 4 のそれぞれの (N - 1) 番目のブロック B L 1 , B L 2 の予測動きベクトル M V 1 F * , M V 2 * を算出する。さらに、探索範囲制御部 1 2 4 は、その算出された予測動きベクトル M V 1 F * , M V 2 * によって示される探索範囲を特定する (ステップ S 5 0 2) 。動きベクトル検出部 1 2 1 は、このような予測動きベクトルの算出および探索範囲の特定が行なわれている間に、ピクチャ B 3 の (N - 2) 番目のブロック B L 1 に対して既に特定されている探索範囲から、そのブロック B L 1 の動きベクトル M V 1 F を検出する (ステップ S 5 0 4) 。

【 0 2 0 4 】

その後、動きベクトル検出部 1 2 1 は、上述と同様に、ピクチャ P 4 の (N - 2) 番目のブロック B L 2 に対して既に特定されている探索範囲から、そのブロック B L 2 の動きベクトル M V 2 を検出する (ステップ S 5 0 6) 。

【 0 2 0 5 】

そして、動画像符号化装置 1 0 0 は、上述のように検出された動きベクトル M V 2 を用いて、ピクチャ P 4 の (N - 2) 番目のブロック B L 2 の予測残差画像を符号化するとともに、その符号化された予測残差画像を復号化して再構築画像を生成し、ピクチャメモリ 1 2 0 に格納する (ステップ S 5 1 0) 。動きベクトル検出部 1 2 1 は、このような符号化および復号化処理が行なわれている間に、ピクチャ B 2 の N 番目のブロック B L 0 の動きベクトル M V 0 F を検出する (ステップ S 5 0 8) 。

【 0 2 0 6 】

その後、上述と同様、予測演算部 1 2 3 は、その動きベクトル M V 0 F を用いて、ピクチャ B 3 , P 4 のそれぞれの N 番目のブロック B L 1 , B L 2 の予測動きベクトル M V 1 F * , M V 2 * を算出する。さらに、探索範囲制御部 1 2 4 は、その算出された予測動きベクトル M V 1 F * , M V 2 * によって示される探索範囲を特定する (ステップ S 5 1 2) 。動きベクトル検出部 1 2 1 は、このような予測動きベクトルの算出および探索範囲の特定が行なわれている間に、ピクチャ B 3 の (N - 1) 番目のブロック B L 1 に対して既に特定されている探索範囲から、そのブロック B L 1 の動きベクトル M V 1 F を検出する (ステップ S 5 1 4) 。

【 0 2 0 7 】

その後、動きベクトル検出部 1 2 1 は、上述と同様に、ピクチャ P 4 の (N - 1) 番目のブロック B L 2 に対して既に特定されている探索範囲から、そのブロック B L 2 の動きベクトル M V 2 を検出する (ステップ S 5 1 6) 。

【 0 2 0 8 】

図 2 0 は、本変形例において動きベクトルの検出対象となるブロックの順序を示す図である。

【 0 2 0 9 】

10

20

30

40

50

なお、図 20 中の各ピクチャのブロック内に示される N や $(N - 1)$ は、そのピクチャにおけるブロックの位置を示す。

【0210】

本変形例では、パイプライン処理を行なうことにより、ピクチャ B 2 の N 番目のブロックの前方動きベクトルが検出された後、ピクチャ B 3 の $(N - 1)$ 番目のブロックの前方動きベクトルが検出され、その後、ピクチャ P 4 の $(N - 1)$ 番目のブロックの前方動きベクトルが検出される。

【0211】

ここで、本変形例に係る動画像符号化装置 100 は、変形例 9 における予測動きベクトルの算出方法を用いて動きベクトルを検出する場合であっても、その動きベクトルの検出と、符号化および復号化処理とをパイプライン処理により行なう。

10

【0212】

図 21 は、本変形例の他のパイプライン処理を示す図である。

例えば、動きベクトル検出装置 100 a は、参照ピクチャ P 1 と参照ピクチャ P 4 とが既に符号化および復号化されている状態において、まず、参照ピクチャ P 4 の $(N - 1)$ 番目のブロック B L 2 の動きベクトル $MV 2$ をスケーリングすることにより、ピクチャ B 2 の $(N - 1)$ 番目のブロック B L 0 の予測動きベクトル $MV 0 F^*$ 、 $MV 0 B^*$ と、ピクチャ B 3 の $(N - 1)$ 番目のブロック B L 1 の予測動きベクトル $MV 1 F^*$ 、 $MV 1 B^*$ と、ピクチャ P 7 の $(N - 1)$ 番目のブロック B L 5 の予測動きベクトル $MV 5^*$ とを算出する。そして、動きベクトル検出装置 100 a は、算出した予測動きベクトルのそれぞれから探索範囲を特定する (ステップ S 600)。

20

【0213】

次に、動きベクトル検出装置 100 a は、ピクチャ B 2 の $(N - 1)$ 番目のブロック B L 0 の動きベクトル $MV 0 F$ 、 $MV 0 B$ を検出する (ステップ S 602)。

【0214】

その後、動画像符号化装置 100 は、その検出された動きベクトル $MV 0 F$ 、 $MV 0 B$ を用いて、ピクチャ B 2 の $(N - 1)$ 番目のブロック B L 0 の符号化を行なう (ステップ S 604)。このような符号化が行なわれている間に、動きベクトル検出装置 100 a は、ピクチャ B 3 の $(N - 1)$ 番目のブロック B L 1 の動きベクトル $MV 1 F$ 、 $MV 1 B$ を検出する (ステップ S 606)。

30

【0215】

その後、動画像符号化装置 100 は、その検出された動きベクトル $MV 1 F$ 、 $MV 1 B$ を用いて、ピクチャ B 3 の $(N - 1)$ 番目のブロック B L 1 の符号化を行なう (ステップ S 610)。このような符号化が行なわれている間に、動きベクトル検出装置 100 a は、ピクチャ P 7 の $(N - 1)$ 番目のブロック B L 5 の動きベクトル $MV 5$ を検出する (ステップ S 612)。さらに、このとき、動きベクトル検出装置 100 a は、参照ピクチャ P 4 の N 番目のブロック B L 2 の動きベクトル $MV 2$ をスケーリングすることにより、ピクチャ B 2 の N 番目のブロック B L 0 の予測動きベクトル $MV 0 F^*$ 、 $MV 0 B^*$ と、ピクチャ B 3 の N 番目のブロック B L 1 の予測動きベクトル $MV 1 F^*$ 、 $MV 1 B^*$ と、ピクチャ P 7 の N 番目のブロック B L 5 の予測動きベクトル $MV 5^*$ とを算出する。そして、動きベクトル検出装置 100 a は、算出した予測動きベクトルのそれぞれから探索範囲を特定する (ステップ S 608)。

40

【0216】

次に、動きベクトル検出装置 100 a は、ピクチャ B 2 の N 番目のブロック B L 0 の動きベクトル $MV 0 F$ 、 $MV 0 B$ を検出する (ステップ S 614)。さらに、このとき、動画像符号化装置 100 は、ステップ S 612 で検出された動きベクトル $MV 5$ を用いて、ピクチャ P 7 の $(N - 1)$ 番目のブロック B L 5 の予測残差画像を符号化するとともに、その符号化された予測残差画像を復号化して再構築画像を生成し、ピクチャメモリ 120 に格納する (ステップ S 616)。

【0217】

50

その後、動画像符号化装置100は、ステップS614で検出された動きベクトルMV0F, MV0Bを用いて、ピクチャB2のN番目のブロックBL0の符号化を行なう(ステップS618)。このような符号化が行なわれている間に、動きベクトル検出装置100aは、ピクチャB3のN番目のブロックBL1の動きベクトルMV1F, MV1Bを検出する(ステップS620)。

【0218】

その後、動画像符号化装置100は、その検出された動きベクトルMV1F, MV1Bを用いて、ピクチャB3のN番目のブロックBL1の符号化を行なう(ステップS624)。このような符号化が行なわれている間に、動きベクトル検出装置100aは、ピクチャP7のN番目のブロックBL5の動きベクトルMV5を検出する(ステップS626)。さらに、このとき、動きベクトル検出装置100aは、参照ピクチャP4の(N+1)番目のブロックBL2の動きベクトルMV2をスケールリングすることにより、ピクチャB2の(N+1)番目のブロックBL0の予測動きベクトルMV0F*, MV0B*と、ピクチャB3の(N+1)番目のブロックBL1の予測動きベクトルMV1F*, MV1B*と、ピクチャP7の(N+1)番目のブロックBL5の予測動きベクトルMV5*とを算出する。そして、動きベクトル検出装置100aは、算出した予測動きベクトルのそれぞれから探索範囲を特定する(ステップS622)。

【0219】

このように、本変形例では、動きベクトルの検出処理と他の処理とをパイプライン処理により行うため、全体的な処理時間を短縮することができる。

【0220】

以上、本発明について実施の形態およびその変形例を用いて説明したが、本発明はこれらに限定されるものではない。

【0221】

例えば、変形例10では、動きベクトルの検出処理と他の処理とをパイプライン処理により行ったが、複数の動きベクトルの検出処理を並列に実行してもよい。この場合には、動きベクトル検出装置100aは、動きベクトル検出部121を複数個備える。そして、これらの動きベクトル検出部121は、例えば、図19に示すピクチャB3の(N-2)番目のブロックの前方動きベクトルMV1Fの検出と、ピクチャP4の(N-2)番目のブロックの動きベクトルMV2の検出とを同時に実行する。

【0222】

また、本発明では、ブロック図(図1など)の各機能ブロックを集積回路であるLSIとして実現してもよい。これらは個別に1チップ化されても良いし、一部又は全てを含むように1チップ化されても良い。(例えばメモリ以外の機能ブロックが1チップ化されていても良い。)ここでは、LSIとしたが、集積度の違いにより、IC、システムLSI、スーパーLSI、ウルトラLSIと称されることもある。

【0223】

また、集積回路化の手法はLSIに限るものではなく、専用回路又は汎用プロセサで実現してもよい。LSI製造後に、プログラムすることが可能なFPGA(Field Programmable GateArray)や、LSI内部の回路セルの接続や設定を再構成可能なりコンフィギュラブル・プロセッサを利用しても良い。

【0224】

さらには、半導体技術の進歩又は派生する別技術によりLSIに置き換わる集積回路化の技術が登場すれば、当然、その技術を用いて機能ブロックの集積化を行ってもよい。バイオ技術の適応等が可能性としてありえる。

【0225】

また、各機能ブロックのうち、符号化または復号化の対象となるデータを格納する手段だけ1チップ化せずに別構成としても良い。

【産業上の利用可能性】**【0226】**

10

20

30

40

50

本発明の動きベクトル検出装置は、動きベクトルの検出のための演算量を低減して回路規模の縮小化を図ることができるという効果を奏し、例えば、携帯電話、ビデオカメラ、DVD装置、およびパーソナルコンピュータ等において、動画像を構成する各ピクチャを符号化して符号列を生成したり、生成された符号列を復号化したりするための装置として有用である。

【図面の簡単な説明】

【0227】

【図1】本発明の実施の形態における動きベクトル検出装置を備えた動画像符号化装置のブロック図である。

【図2】時間ダイレクトモードにおける動きベクトルの予測生成方法を示す模式図である。

10

【図3】本発明の実施の形態における予測演算部によって算出される前方向の予測動きベクトルを説明するための説明図である。

【図4】同上の動きベクトルの検出の順序を示す図である。

【図5】同上の予測演算部によって算出される後方向の予測動きベクトルを説明するための説明図である。

【図6】同上の動きベクトル検出装置が動きベクトルを検出する動作を示すフローチャートである。

【図7】同上のモード選択部が非時間ダイレクトモードの評価値を算出する動作を示すフローチャートである。

20

【図8】同上のモード選択部が時間ダイレクトモードの評価値を算出する動作を示すフローチャートである。

【図9】同上の変形例1における予測動きベクトルの算出方法を示す図である。

【図10】同上の変形例2における予測動きベクトルの算出方法を示す図である。

【図11】同上の変形例3における予測動きベクトルの算出方法を示す図である。

【図12】同上の変形例4における予測動きベクトルの算出方法を示す図である。

【図13】同上の変形例5における予測動きベクトルの算出方法を示す図である。

【図14】同上の変形例6における予測動きベクトルの算出方法を示す図である。

【図15】同上の変形例7における予測動きベクトルの算出方法を示す図である。

【図16】同上の変形例8における探索範囲を示す図である。

30

【図17】同上の変形例9において予測動きベクトルを算出する前の各ピクチャの状態を示す図である。

【図18】同上の変形例9における予測動きベクトルの算出方法を示す図である。

【図19】同上の変形例10におけるパイプライン処理を示す図である。

【図20】同上の変形例10において動きベクトルの検出対象となるブロックの順序を示す図である。

【図21】同上の変形例10の他のパイプライン処理を示す図である。

【図22】動画像符号化方式における各ピクチャの予測関係を示す模式図である。

【図23】ピクチャの表示順序および符号化順序を示す図である。

【図24】H.264における参照関係を示す図である。

40

【図25】従来の動きベクトルの検出における探索範囲を示す図である。

【図26】テレスコピックサーチを示す図である。

【図27】一枚の参照ピクチャにおける探索範囲の変化を示す図である。

【符号の説明】

【0228】

100 動画像符号化装置

100a 動きベクトル検出装置

111 ブロック選択部

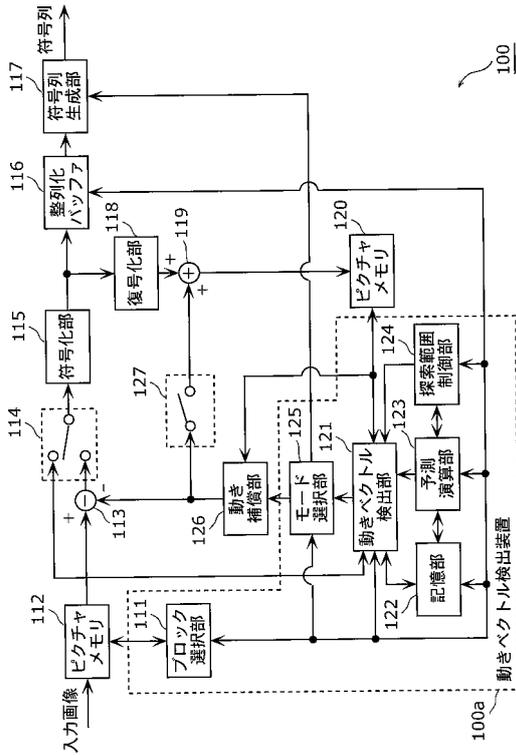
112 ピクチャメモリ

113 差分演算部

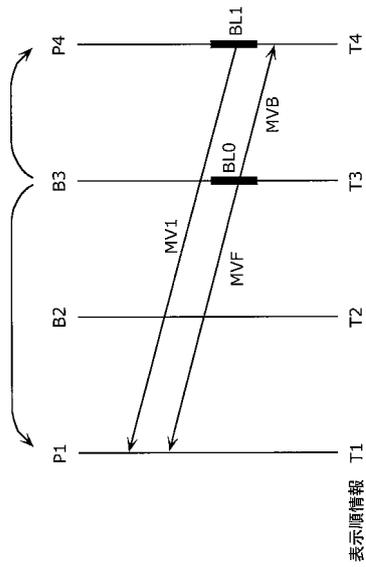
50

- 1 1 4 スイッチ
- 1 1 5 符号化部
- 1 1 6 整列化バッファ
- 1 1 7 符号列生成部
- 1 1 8 復号化部
- 1 1 9 加算演算部
- 1 2 1 動きベクトル検出部
- 1 2 2 記憶部
- 1 2 3 予測演算部
- 1 2 4 探索範囲制御部
- 1 2 5 モード選択部
- 1 2 6 動き補償部
- 1 2 7 スイッチ

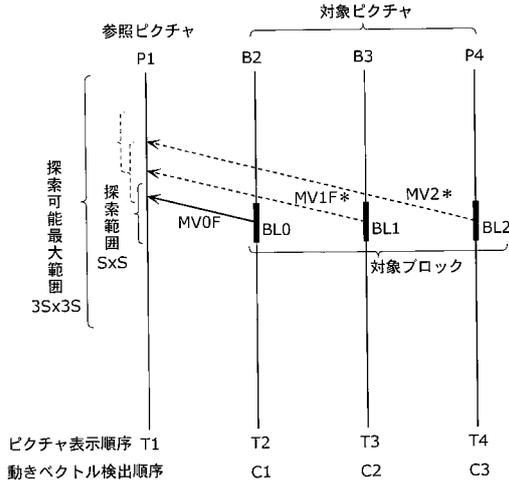
【図1】



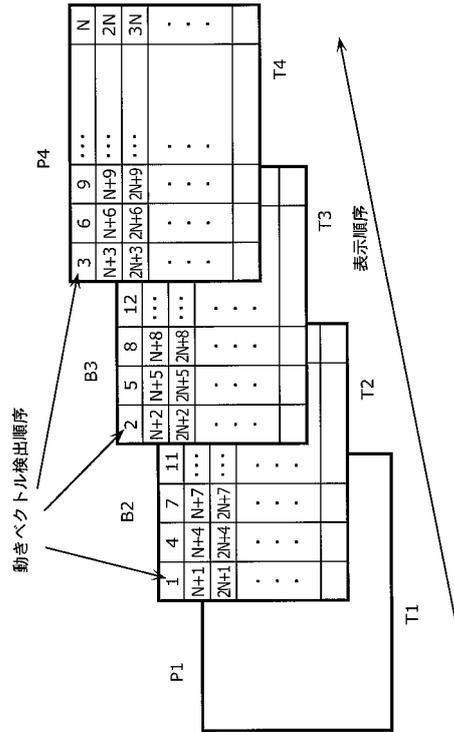
【図2】



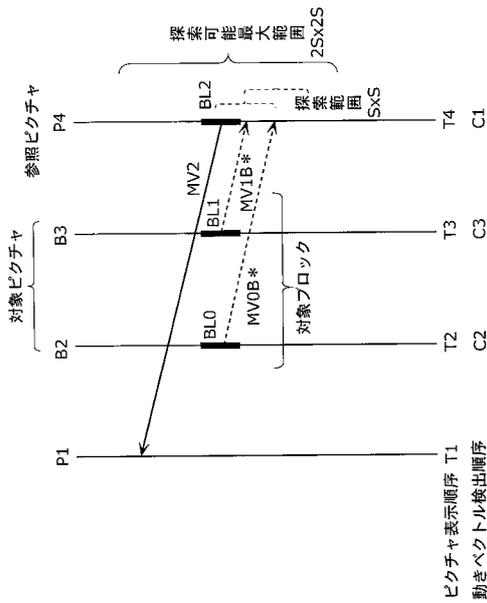
【図3】



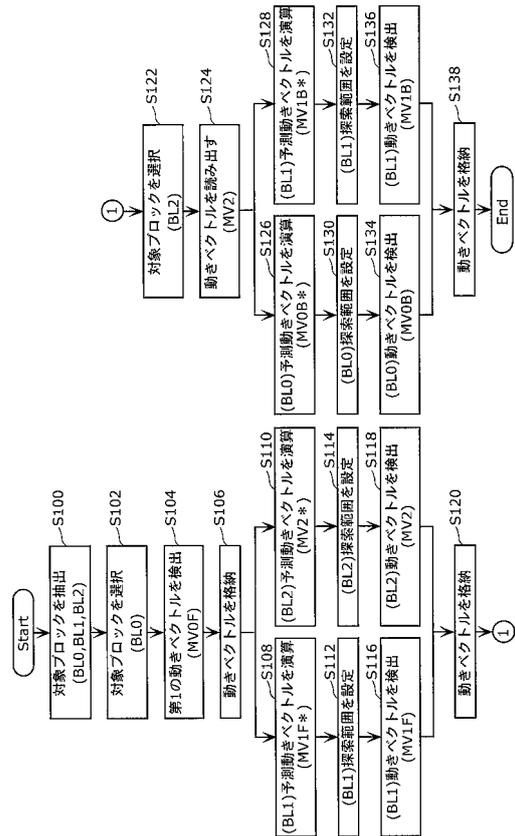
【図4】



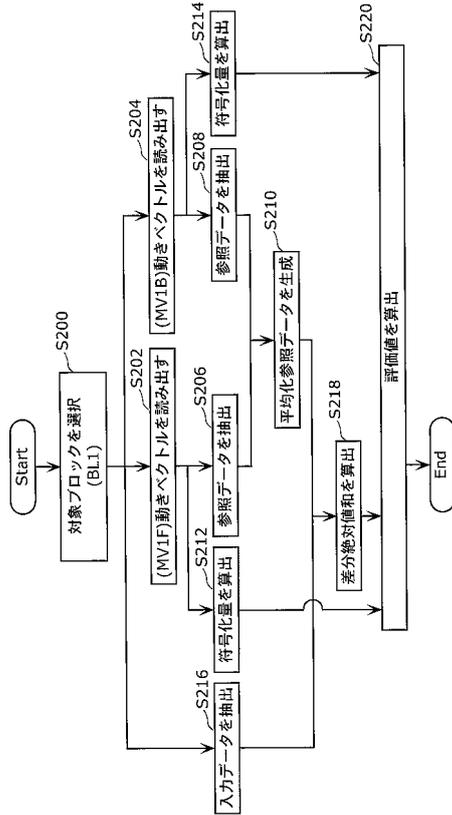
【図5】



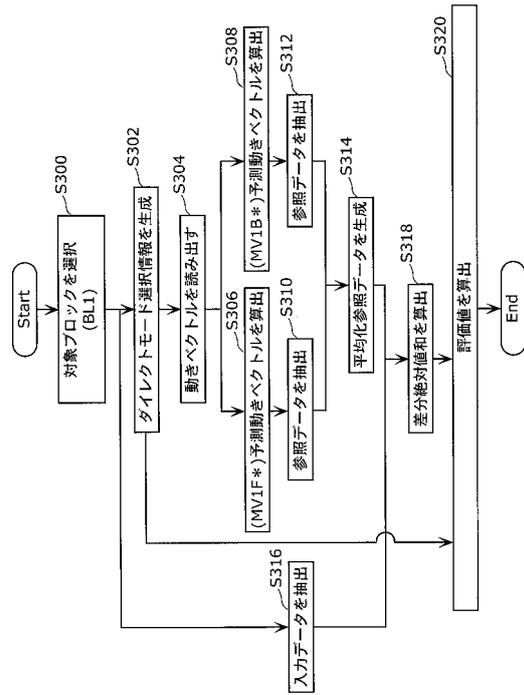
【図6】



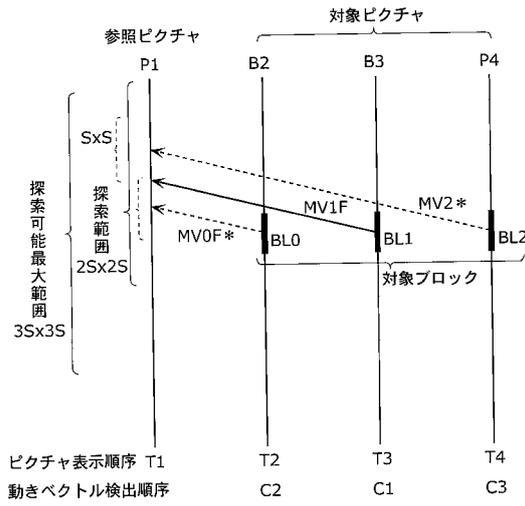
【図7】



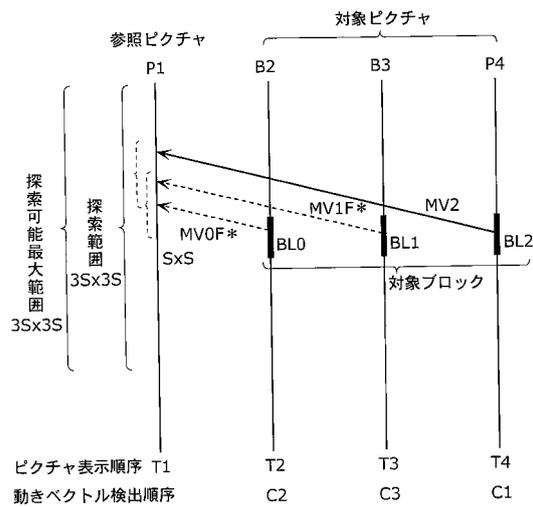
【図8】



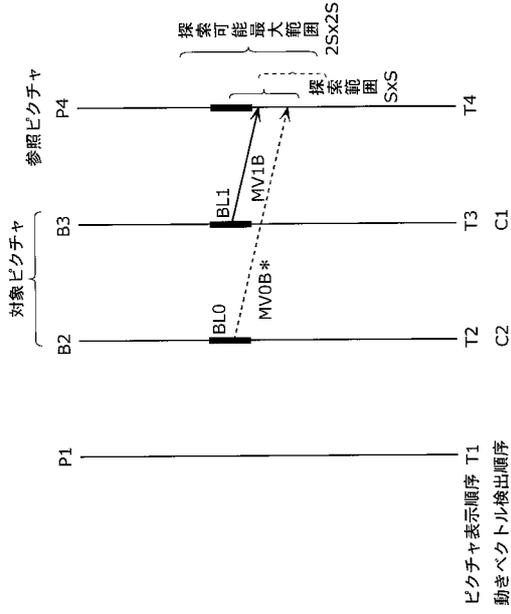
【図9】



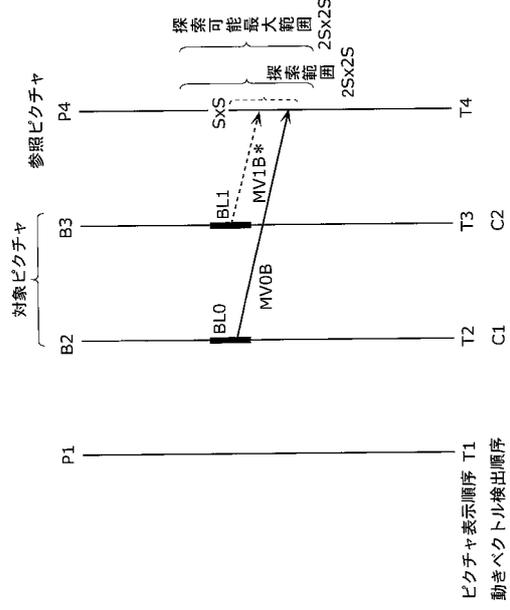
【図10】



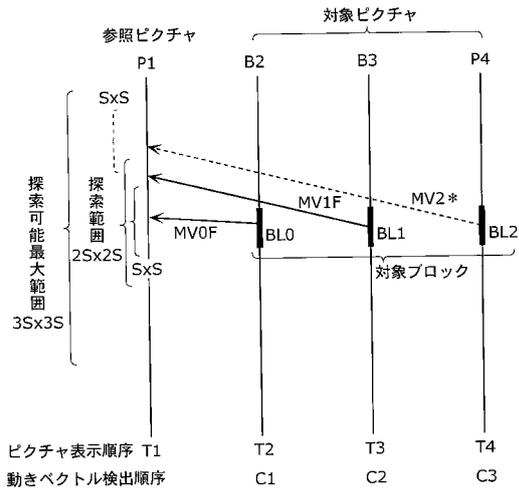
【図11】



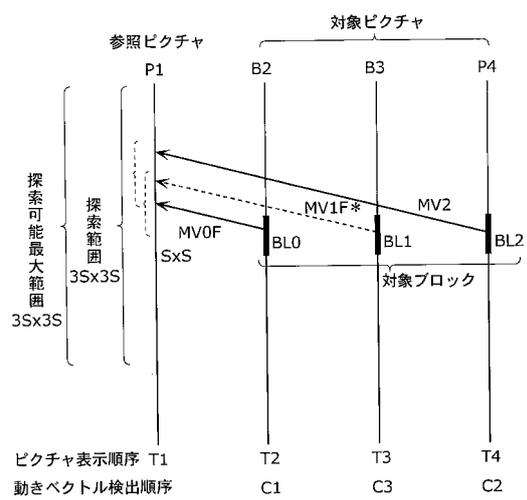
【図12】



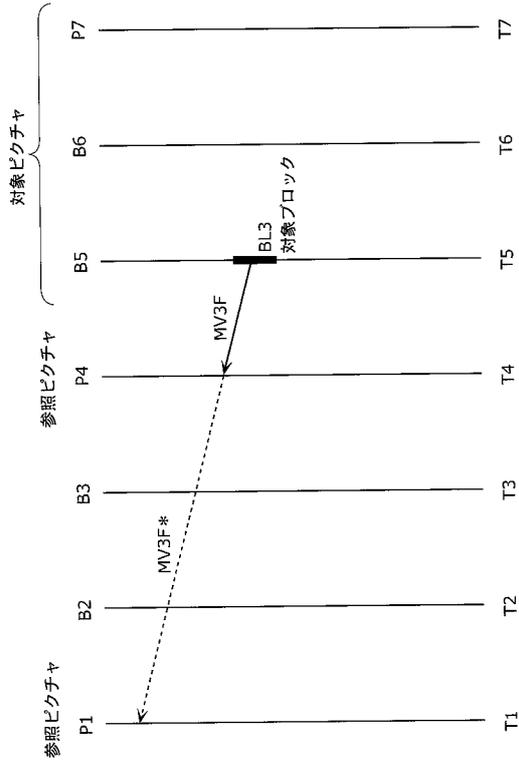
【図13】



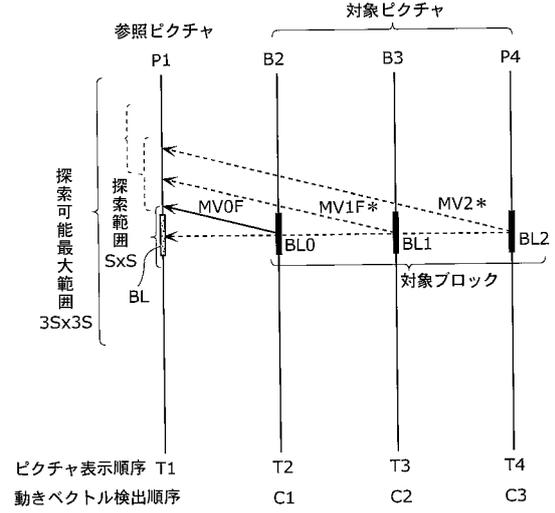
【図14】



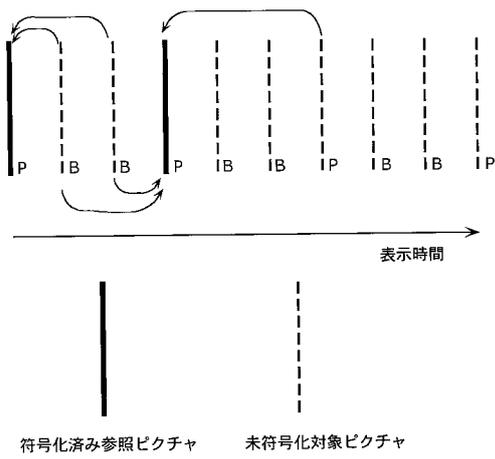
【図15】



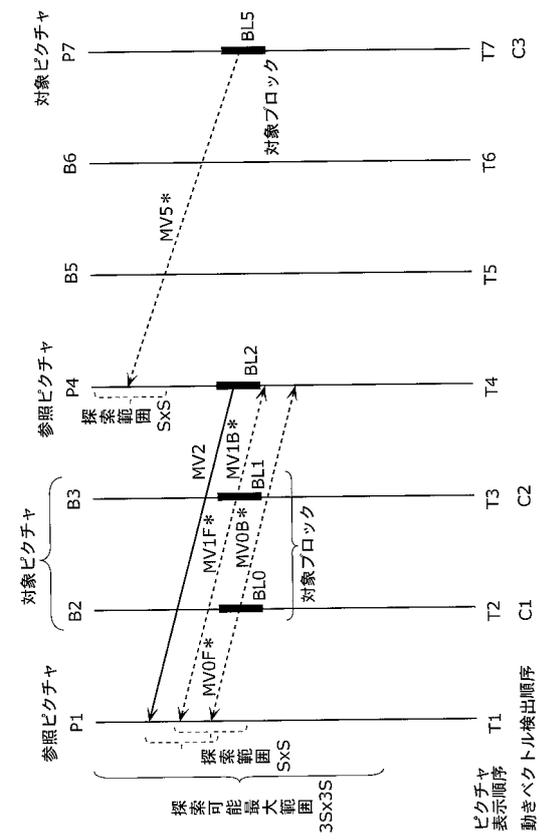
【図16】



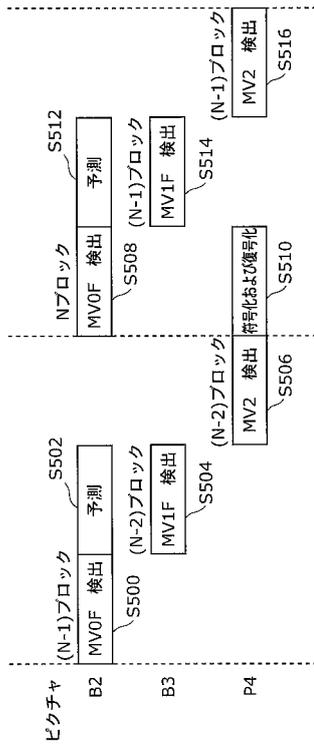
【図17】



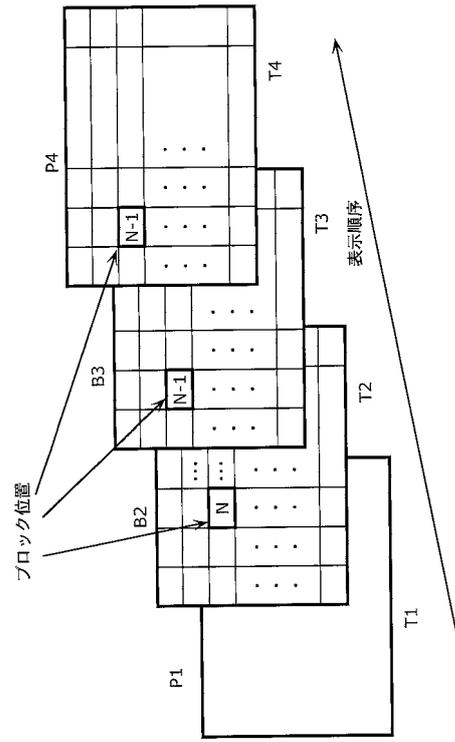
【図18】



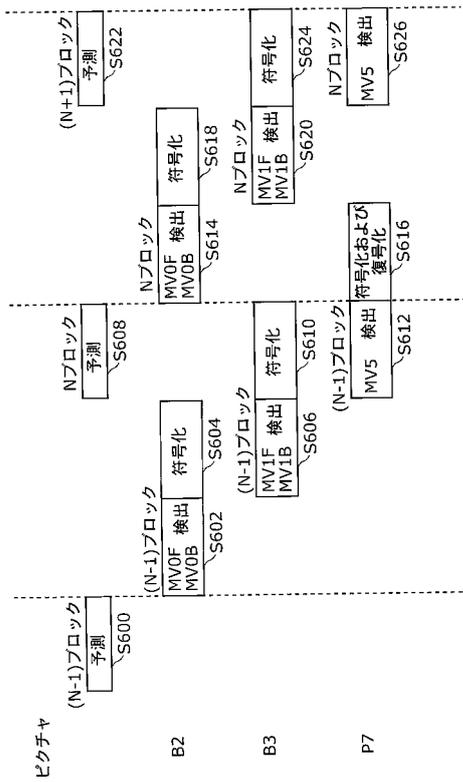
【 図 19 】



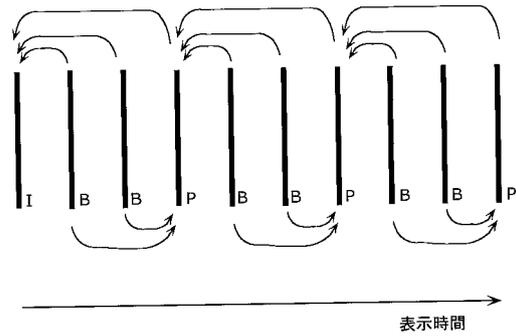
【 図 20 】



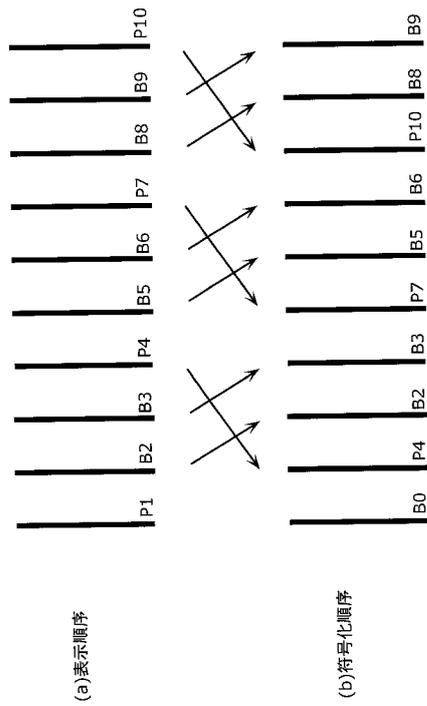
【 図 21 】



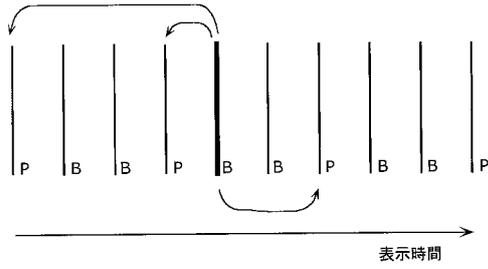
【 図 22 】



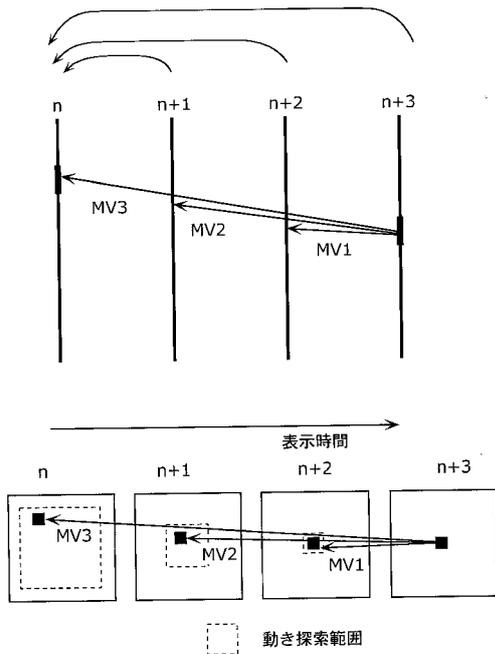
【図23】



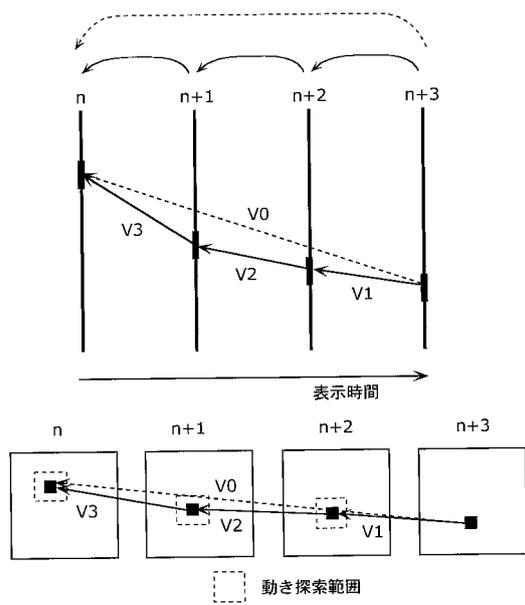
【図24】



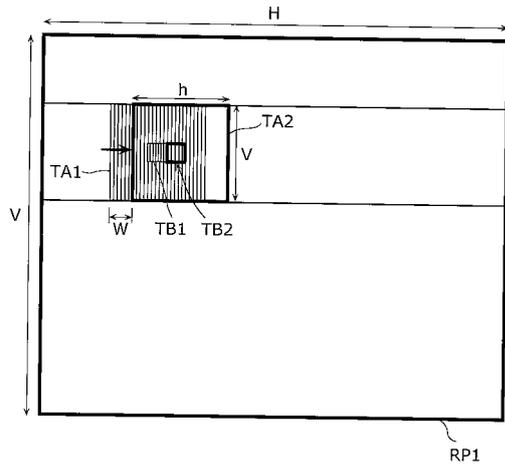
【図25】



【図26】



【 図 27 】



フロントページの続き

- (72)発明者 有村 耕治
大阪府門真市大字門真1006番地 松下電器産業株式会社内
- (72)発明者 荒川 博
大阪府門真市大字門真1006番地 松下電器産業株式会社内
- (72)発明者 田坂 啓
大阪府門真市大字門真1006番地 松下電器産業株式会社内
- (72)発明者 丸山 悠樹
大阪府門真市大字門真1006番地 松下電器産業株式会社内

審査官 坂東 大五郎

- (56)参考文献 特開2004-173009(JP,A)
特開平09-168153(JP,A)
特開平04-150284(JP,A)

- (58)調査した分野(Int.Cl., DB名)
H04N 7/24 - 7/68