

【公報種別】特許法第 17 条の 2 の規定による補正の掲載

【部門区分】第 7 部門第 3 区分

【発行日】平成 26 年 12 月 4 日 (2014.12.4)

【公表番号】特表 2013-507055 (P2013-507055A)

【公表日】平成 25 年 2 月 28 日 (2013.2.28)

【年通号数】公開・登録公報 2013-010

【出願番号】特願 2012-532241 (P2012-532241)

【国際特許分類】

H 0 4 N 19/50 (2014.01)

【F I】

H 0 4 N 7/137 Z

【誤訳訂正書】

【提出日】平成 26 年 10 月 6 日 (2014.10.6)

【誤訳訂正 1】

【訂正対象書類名】特許請求の範囲

【訂正対象項目名】全文

【訂正方法】変更

【訂正の内容】

【特許請求の範囲】

【請求項 1】

参照フレームの N 個のリストと現在フレームとに対応する N 個の動きベクトルを有する第 1 のブロックについて、 N 個の動きベクトルプレディクタを導出するステップであって、

前記第 1 のブロックに隣接し、予測のために使用される第 2 のブロックが、少なくとも 1 つの無効なステータスに設定された動きベクトルを有するときに、前記 N 個の動きベクトルプレディクタの 1 つを構成するサブステップを含み、 N が 1 よりも大きい整数であり、

前記 N 個の動きベクトルプレディクタの 1 つを構成するサブステップは、前記 N 個の動きベクトルプレディクタのうちの当該 1 つの動きベクトルプレディクタとは別の動きベクトルプレディクタ、および現在フレームから前記第 1 のブロックの前記 N 個の動きベクトル内のそれぞれの動きベクトルに関連した参照フレームまでの符号付き時間距離に基づく関数を適用することにより行われる、ステップと、

前記 N 個の動きベクトルと N 個の動きベクトルプレディクタとに基づいて、 N 個の差分動きベクトルを生成するステップと、

前記 N 個の差分動きベクトルを符号化するステップとを含む方法。

【請求項 2】

N 個の動きベクトルプレディクタを導出するステップが、1 対の動きベクトルプレディクタを導出するサブステップを含み、前記第 1 のブロックが、参照フレームの 2 つのリストに対応する 1 対の動きベクトルを有する、請求項 1 に記載の方法。

【請求項 3】

参照フレームの前記 2 つのリストの少なくとも一方について、第 2 のブロックの動きベクトルによって使用される参照フレームが、第 1 のブロックの動きベクトルによって使用される参照フレームと一致しない場合、又は前記第 2 のブロックが、参照フレームの片方のリストは使う場合、前記第 2 のブロックが、少なくとも 1 つの無効なステータスに設定された動きベクトルを有する、請求項 2 に記載の方法。

【請求項 4】

前記 1 対の動きベクトルプレディクタの一方を構成するステップが、前記 1 対のうちの

他方の動きベクトルプレディクタと、前記第 1 のブロックの前記 1 対の動きベクトルのうちの各動きベクトルに関連付けられた、前記現在フレームから前記参照フレームまでの符号付き時間距離と、に基づいた関数を適用することによって実行される、請求項 2 に記載の方法。

【請求項 5】

前記 1 対の動きベクトルプレディクタの一方を構成するステップが、参照フレームの前記 2 つのリストの少なくとも一方について、第 2 のブロックの動きベクトルによって使用される参照フレームが、第 1 のブロックの動きベクトルによって使用される参照フレームと一致しないために、前記第 2 のブロックが、少なくとも 1 つの無効なステータスに設定された動きベクトルを有するときに、一方のリストに対応する前記第 2 のブロックの有効な動きベクトルを、他方のリスト上の参照フレームへ外挿するサブステップを含む、請求項 4 に記載の方法。

【請求項 6】

参照フレームの N 個のリストと現在フレームとに対応する N 個の動きベクトルを有する第 1 のブロックについて、 N 個の動きベクトルプレディクタを導出するステップであって、

前記第 1 のブロックに隣接し、予測のために使用される第 2 のブロックが、少なくとも 1 つの無効なステータスに設定された動きベクトルを有するときに、前記 N 個の動きベクトルプレディクタの 1 つを構成するサブステップを含み、 N が 1 よりも大きい整数であり、

N 個の動きベクトルプレディクタを導出するステップが、1 対の動きベクトルプレディクタを導出するサブステップを含み、

前記第 1 のブロックが、参照フレームの前記 2 つのリストに対応する 1 対の動きベクトルを有し、

前記 1 対の動きベクトルプレディクタの 1 つを構成するサブステップは、前記第 2 のブロックの前記動きベクトルにより使用された前記参照フレームと、前記第 1 のブロックの前記動きベクトルにより使用された前記参照フレームとが、前記参照フレームの前記 2 つのリストのうちの少なくとも一つにおいて一致しないことから、前記第 2 のブロックが少なくとも 1 つの無効なステータスに設定された動きベクトルを有するときに、一方のリストに対応する前記第 2 のブロックの有効な動きベクトルを、他方のリスト上の参照フレームへ外挿するサブステップを含み、

前記 1 対の動きベクトルプレディクタの 1 つを構成するサブステップは関数を適用することにより行われ、前記関数が、

$(MVP_a \times T_b \times f_T(T_a) + (1 - (N_{prec1} - 1))) \times N_{prec1}$ 、ここで、 $f_T(T_a)$ は、

【数 1】

$$f_T(x) = \frac{1 \ll N_{prec1}}{x}$$

であり、 N_{prec1} が所定の正の整数となるように、そのエントリが満たされたルックアップテーブル i

【数 2】

$$\frac{MVP_a \times T_b}{T_a}$$

i
前記 1 対のうちの他方の動きベクトルプレディクタ i

$MVP_a \times f_{scale}(T_a, T_b)$ 、ここで、 $f_{scale}(T_a, T_b)$ は、ルックアップテーブル; 及び

ルックアップテーブル;

から成る群から選択される1つであり、

ここで、 MVP_a は、前記1対のうちの他方の動きベクトルプレディクタであり、 T_a 及び T_b は、前記第1のブロックの前記1対の動きベクトルのうちの各動きベクトルに関連付けられた、前記現在フレームから前記参照フレームまでの符号付き時間距離を表す、ステップと、

前記N個の動きベクトルと前記N個の動きベクトルプレディクタとに基づいて、N個の差分動きベクトルを生成するステップと、

前記N個の差分動きベクトルを符号化するステップと
を含む方法。

【請求項7】

前記第2のブロックの前記無効なステータスに設定された動きベクトルが、前記第2のブロックの有効な動きベクトルに基づく値が所定の条件を満たす場合に限り、構成された前記一方の動きベクトルプレディクタが使用される、請求項1に記載の方法。

【請求項8】

前記N個の差分動きベクトルの1つを生成するステップが、別の差分動きベクトルを使用する関数を実行するサブステップを含む、請求項1に記載の方法。

【請求項9】

参照フレームのN個のリストと現在フレームとに対応するN個の動きベクトルを有する第1のブロックについて、N個の動きベクトルプレディクタを導出する動きベクトルプレディクタモジュールであって、

N個の動きベクトルプレディクタの導出が、前記第1のブロックに隣接し、予測のために使用される第2のブロックが、少なくとも1つの無効なステータスに設定された動きベクトルを有するときに、前記N個の動きベクトルプレディクタの1つを構成することを含み、Nが1よりも大きい整数であり、

前記N個の動きベクトルプレディクタの1つの構成は、前記N個の動きベクトルプレディクタのうちの当該1つの動きベクトルプレディクタとは別の動きベクトルプレディクタ、および現在フレームから前記第1のブロックの前記N個の動きベクトル内のそれぞれの動きベクトルに関連した参照フレームまでの符号付き時間距離に基づく関数を適用することにより行われる、動きベクトルプレディクタモジュールと、

前記N個の動きベクトルとN個の動きベクトルプレディクタとに基づいて、N個の差分動きベクトルを生成する差分動きベクトル生成器と、

前記N個の差分動きベクトルを符号化するエントロピエンコーダと
を備える符号化装置。

【請求項10】

参照フレームの前記N個のリストの少なくとも1つについて、第2のブロックの動きベクトルによって使用される参照フレームが、第1のブロックの動きベクトルによって使用される参照フレームと一致しない場合、又は前記第2のブロックが、参照フレームのN個のリストのうち一部は使う場合、前記第2のブロックが、少なくとも1つの無効なステータスに設定された動きベクトルを有する、請求項9に記載の装置。

【請求項11】

動きベクトルプレディクタモジュールが、関数を適用して、1つの動きベクトルプレディクタを構成するように動作可能であり、前記関数が、前記N個の動きベクトルプレディクタのうちの別の動きベクトルプレディクタと、前記関数に係わる動きベクトルプレディクタに対応する、前記第1のブロックの前記動きベクトルに関連付けられた、前記現在フレームから前記参照フレームまでの符号付き時間距離と、に基づく、請求項9に記載の装置。

【請求項12】

前記差分動きベクトル生成器が、別の差分動きベクトルを使用する関数を実行することによって、前記N個の差分動きベクトルの1つを生成するように動作可能である、請求項9に記載の装置。

【請求項13】

前記関数が、生成される差分動きベクトルを予測するために、前記別の差分動きベクトルをスケーリングするような、スケーリング関数である、請求項12に記載の装置。

【請求項14】

ビットストリームを受け取り、前記ビットストリームを復号して、差分動きベクトルを含む復号されたビットストリームを生成するエンロピデコーダと、

参照フレームのN個のリストと現在フレームとに対応するN個の差分動きベクトルを有する第1のブロックについて、N個の動きベクトルプレディクタを導出する動きベクトルプレディクタモジュールであって、N個の動きベクトルプレディクタの導出が、前記第1のブロックに隣接し、予測のために使用される第2のブロックが、少なくとも1つの無効なステータスに設定された動きベクトルを有するときに、前記N個の動きベクトルプレディクタの1つを構成することを含み、Nが1よりも大きい整数であり、

前記N個の動きベクトルプレディクタの1つの構成は、前記N個の動きベクトルプレディクタのうちの当該1つの動きベクトルプレディクタとは別の動きベクトルプレディクタ、および現在フレームから前記第1のブロックの前記N個の動きベクトル内のそれぞれの動きベクトルに関連した参照フレームまでの符号付き時間距離に基づく関数を適用することにより行われる、動きベクトルプレディクタモジュールと、

前記復号されたビットストリームからの前記N個の差分動きベクトルと前記N個の動きベクトルプレディクタとに基づいて、N個の動きベクトルを生成する動きベクトル生成器と

を備える復号装置。

【請求項15】

参照フレームの前記N個のリストの少なくとも1つについて、第2のブロックの動きベクトルによって使用される参照フレームが、第1のブロックの動きベクトルによって使用される参照フレームと一致しない場合、又は前記第2のブロックが、参照フレームのN個のリストのうち一部は使う場合、前記第2のブロックが、少なくとも1つの無効なステータスに設定された動きベクトルを有する、請求項14に記載の装置。

【請求項16】

プレディクタ集合改良器が、関数を適用して、動きベクトルプレディクタを改良するように動作可能であり、前記関数が、別の動きベクトルプレディクタと、前記関数に係わる2つの動きベクトルプレディクタに対応する、前記第1のブロックの前記動きベクトルに関連付けられた、前記現在フレームから前記参照フレームまでの符号付き時間距離とに基づく、請求項14に記載の装置。

【請求項17】

前記デコーダが、動きベクトルを生成する際に通常使用される差分動きベクトルに加えて、別の差分動きベクトルを使用する関数を実行することによって、前記N個の動きベクトルの1つを生成するように動作可能である、請求項14に記載の装置。

【請求項18】

システムによって実行されたときに、前記システムに、

参照フレームのN個のリストと現在フレームとに対応するN個の動きベクトルを有する第1のブロックについて、N個の動きベクトルプレディクタを導出するステップであって、前記第1のブロックに隣接し、予測のために使用される第2のブロックが、少なくとも1つの無効なステータスに設定された動きベクトルを有するときに、前記N個の動きベクトルプレディクタの1つを構成するサブステップを含み、Nが1よりも大きい整数であり、

前記N個の動きベクトルプレディクタの1つの構成は、前記N個の動きベクトルプレディクタのうちの当該1つの動きベクトルプレディクタとは別の動きベクトルプレディクタ

、および現在フレームから前記第 1 のブロックの前記 N 個の動きベクトル内のそれぞれの動きベクトルに関連した参照フレームまでの符号付き時間距離に基づく関数を適用することにより行われる、ステップと、

前記 N 個の動きベクトルと前記 N 個の動きベクトルプレディクタとに基づいて、N 個の差分動きベクトルを生成するステップと、

前記 N 個の差分動きベクトルを符号化するステップと

を含む方法を実行させる実行可能命令を記憶する非一時的なコンピュータ可読記憶媒体

。

【請求項 19】

参照フレームの前記 N 個のリストの少なくとも 1 つについて、第 2 のブロックの動きベクトルによって使用される参照フレームが、第 1 のブロックの動きベクトルによって使用される参照フレームと一致しない場合、又は前記第 2 のブロックが、参照フレームの N 個のリストのうち一部は使う場合、前記第 2 のブロックが、少なくとも 1 つの無効なステータスに設定された動きベクトルを有する、請求項 18 に記載のコンピュータ可読記憶媒体

。

【請求項 20】

前記 N 個の動きベクトルプレディクタの 1 つを構成するステップが、関数を適用することによって実行され、前記関数が、前記 N 個の動きベクトルプレディクタのうちの別の動きベクトルプレディクタと、前記関数に係わる 2 つの動きベクトルプレディクタに対応する、前記第 1 のブロックの前記動きベクトルに関連付けられた、前記現在フレームから前記参照フレームまでの符号付き時間距離とに基づく、請求項 18 に記載のコンピュータ可読記憶媒体。

【請求項 21】

前記 N 個の動きベクトルプレディクタの 1 つを構成するステップが、参照フレームの前記 N 個のリストの少なくとも 1 つについて、第 2 のブロックの動きベクトルによって使用される参照フレームが、第 1 のブロックの動きベクトルによって使用される参照フレームと一致しないために、前記第 2 のブロックが、少なくとも 1 つの無効なステータスに設定された動きベクトルを有するときに、1 つのリストに対応する前記第 2 のブロックの有効な動きベクトルを、別のリスト上の参照フレームへ外挿するサブステップを含む、請求項 20 に記載のコンピュータ可読記憶媒体。

【請求項 22】

N 個の動きベクトルと N 個の動きベクトルプレディクタとに基づいて、N 個の差分動きベクトルを生成するステップであって、前記 N 個の差分動きベクトルの 1 つを生成するステップが、別の差分動きベクトルを使用する関数を実行するサブステップを含み、N が 1 よりも大きい整数であり、

前記 N 個の動きベクトルプレディクタの 1 つは、前記 N 個の動きベクトルプレディクタのうちの当該 1 つの動きベクトルプレディクタとは別の動きベクトルプレディクタ、および現在フレームから、参照フレームの N 個のリストと現在フレームとに対応する前記 N 個の動きベクトルを有する第 1 のブロックの前記 N 個の動きベクトル内のそれぞれの動きベクトルに関連した参照フレームまでの符号付き時間距離に基づく関数を適用することにより構成される、ステップと、

前記 N 個の差分動きベクトルを符号化するステップと

を含む方法。

【請求項 23】

前記関数が、生成される差分動きベクトルを予測するために、前記別の差分動きベクトルをスケーリングするような、スケーリング関数である、請求項 22 に記載の方法。

【請求項 24】

システムによって実行されたときに、前記システムに、

N 個の動きベクトルと N 個の動きベクトルプレディクタとに基づいて、N 個の差分動きベクトルを生成するステップであって、前記 N 個の差分動きベクトルの 1 つを生成するス

テップが、別の差分動きベクトルを使用する関数を実行するサブステップを含み、Nが1よりも大きい整数であり、

前記N個の動きベクトルプレディクタの1つは、前記N個の動きベクトルプレディクタのうちの当該1つの動きベクトルプレディクタとは別の動きベクトルプレディクタ、および現在フレームから、参照フレームのN個のリストと現在フレームとに対応する前記N個の動きベクトルを有する第1のブロックの前記N個の動きベクトル内のそれぞれの動きベクトルに関連した参照フレームまでの符号付き時間距離に基づく関数を適用することにより構成される、ステップと、

前記N個の差分動きベクトルを符号化するステップと

を含む方法を実行させる実行可能命令を記憶する非一時的なコンピュータ可読記憶媒体

。

【請求項25】

前記関数が、生成される差分動きベクトルを予測するために、前記別の差分動きベクトルをスケーリングするような、スケーリング関数である、請求項24に記載のコンピュータ可読記憶媒体。

【誤訳訂正2】

【訂正対象書類名】明細書

【訂正対象項目名】0004

【訂正方法】変更

【訂正の内容】

【0004】

[0004]本明細書では、動きベクトル予測及び符号化のための方法及び装置が開示される。一実施形態では、方法は、参照フレームのN個のリストと現在フレームとに対応するN個の動きベクトルを有する第1のブロックについて、N個の動きベクトルプレディクタを導出するステップであって、第1のブロックに隣接し、予測のために使用される第2のブロックが、少なくとも1つの無効なステータスに設定された動きベクトルを有するときに、N個の動きベクトルプレディクタの1つを構成するサブステップを含み、Nが1よりも大きい整数である、ステップと、N個の動きベクトルとN個の動きベクトルプレディクタとに基づいて、N個の差分動きベクトル(differential motion vector)を生成するステップと、N個の差分動きベクトルを符号化するステップとを含む。

【誤訳訂正3】

【訂正対象書類名】明細書

【訂正対象項目名】0005

【訂正方法】変更

【訂正の内容】

【0005】

[0005]本発明は、以下で行われる詳細な説明から、また本発明の様々な実施形態についての添付の図面からより完全に理解されるが、それらは、本発明を特定の実施形態に限定するものと見なされるべきではなく、もっぱら説明及び理解のためのものである。

【図1】別のブロックの動きベクトルのためのプレディクタを構成するのに使用される因果的近傍ブロックを示す図である。

【図2】プレディクタ集合を構成するためのプロセスの一実施形態のフロー図である。

【図3】プレディクタが近傍ブロックから構成されるケースを示す図である。

【図4】候補プレディクタを改良するためのプロセスの一実施形態のフロー図である。

【図5】MVPを生成する一例を示す図である。

【図6】動きベクトルプレディクタを導出するためのプロセスの一実施形態のフロー図である。

【図7】エンコーダにおいて差分動きベクトルを計算するためのプロセスの一実施形態のフロー図である。

【図 8】デコーダにおいて動きベクトルを再構成するためのプロセスの一実施形態のフロー図である。

【図 9】一方の M V D をスケーリングして、第 2 の M V D を予測する一例を示す図である。

【図 10 A】エンコーダにおいて動きベクトルを符号化するためのプロセスの一実施形態のフロー図である。

【図 10 B】デコーダにおいて動きベクトルを復号するためのプロセスの一実施形態のフロー図である。

【図 11】動きベクトルエンコーダのブロック図である。

【図 12】動きベクトルデコーダのブロック図である。

【図 13】コンピュータシステムのブロック図である。

【誤訳訂正 4】

【訂正対象書類名】明細書

【訂正対象項目名】0006

【訂正方法】変更

【訂正の内容】

【0006】

[0006]本発明の実施形態は、プレディクタを強化し、それによって、動きベクトルを伝達するために費やされるビットの数を削減する方法を含む。特に、本発明の実施形態は、参照フレームの 2 つ以上のリストの間の相関を利用することによって、双予測 (bi - predictive) モード又はマルチ予測 (multi - predictive) モードで符号化されるブロックについて、動きベクトルの予測を強化する技法を含む。

【誤訳訂正 5】

【訂正対象書類名】明細書

【訂正対象項目名】0014

【訂正方法】変更

【訂正の内容】

【0014】

[0014]双予測ブロックは、参照フレームの 2 つのリスト、リスト 0 及びリスト 1 にそれぞれ対応して、2 つの動きベクトル、M V 0 及び M V 1 を有し、2 つの動きベクトルはともに、差分動きベクトルを使用して伝達される。

【数 3】

$$MVD_0 = MV_0 - \overline{MVP_0}$$

$$MVD_1 = MV_1 - \overline{MVP_1}$$

ここで、M V D 0 及び M V D 1 は、差分動きベクトルを表し、

【数 4】

$$\overline{MVP_0}$$

及び

【数 5】

$$\overline{MVP_1}$$

は、それぞれリスト 0 及びリスト 1 に関する動きベクトルプレディクタを表す。マルチ予測ブロックは、双予測ブロックの概念を一般化したものである。マルチ予測ブロックは、

参照フレームの N 個のリストに対応する N 個の動きベクトルを有し、ここで、 N は 1 よりも大きい整数である。以下の説明では、明瞭となるように、双予測ブロックに関連して、本発明を説明する。しかし、マルチ予測ブロックの一般的なケースにも本発明が適用可能であることに留意されたい。

【誤訳訂正 6】

【訂正対象書類名】明細書

【訂正対象項目名】0022

【訂正方法】変更

【訂正の内容】

【0022】

プレディクタ集合を改良する

[0022]一実施形態では、プレディクタ集合は、参照フレームインデックスが、参照フレームの 2 つのリストの一方について一致する場合、近傍ブロックからプレディクタを構成することによって改良される。これは、参照インデックスが一方のリストでだけ一致する場合、又は近傍ブロックが一方のリストだけを使用する場合に生じる。図 3 は、プレディクタが近傍ブロックから構成されるケースを示している。

【誤訳訂正 7】

【訂正対象書類名】明細書

【訂正対象項目名】0023

【訂正方法】変更

【訂正の内容】

【0023】

[0023]一実施形態では、プレディクタ集合は、以下で説明するように、プレディクタ集合内の 1 つ（又は複数）の候補プレディクタを改良することによって改良され、すなわち、一方のリスト（リスト a）に対応する候補プレディクタの動きベクトル MVP_a が有効であり、他方のリスト（リスト b）に対応する候補プレディクタの動きベクトル MVP_b が無効である場合、 MVP_b についての有効値は、 MVP_a を使用して計算される。 $MVP_b = f_{pred}(MVP_a, T_a, T_b)$

【誤訳訂正 8】

【訂正対象書類名】明細書

【訂正対象項目名】0025

【訂正方法】変更

【訂正の内容】

【0025】

[0025]一実施形態では、候補プレディクタ改良フラグが「オン」である場合に、候補プレディクタが改良される。図 4 は、候補プレディクタ改良フラグ (candidate predictor refinement flag) のステータスに基づいて、候補プレディクタを改良するためのプロセスの一実施形態のフロー図である。プロセスは、ハードウェア、ソフトウェア、又は両方の組み合わせを含むことができる処理ロジックによって実行される。

【誤訳訂正 9】

【訂正対象書類名】明細書

【訂正対象項目名】0026

【訂正方法】変更

【訂正の内容】

【0026】

[0026]図 4 を参照すると、プロセスは、処理ロジックが、候補プレディクタ改良フラグがオンであるかどうかを判定する（処理ブロック 401）ことによって開始する。オンでない場合、プロセスは終了する。オンである場合、プロセスは、処理ブロック 402 に推移し、処理ロジックは、 MVP_0 と MVP_1 の両方が有効であるかどうかを判定する。と

もに有効である場合、プロセスは終了する。どちらかが有効ではない場合、プロセスは、処理ブロック 403 に推移し、処理ロジックは、MVP0 又は MVP1 のどちらかが有効であるかどうかをテストする。どちらも有効でない場合、プロセスは、処理ブロック 404 に推移し、処理ロジックは、候補プレディクタをプレディクタ集合から削除し、その後、プロセスは終了する。MVP0 又は MVP1 のどちらかが有効である場合、プロセスは、処理ブロック 405 に推移し、処理ロジックは、MVP0 が有効であるかどうかを判定する。MVP0 が有効である場合、処理ロジックは、a を 0 に等しく、b を 1 に等しく設定し（処理ブロック 406）、MVPb を $f_{pred}(MVPa, Ta, Tb)$ に等しく設定し（処理ブロック 407）、その後、プロセスは終了する。処理ロジックが MVP0 は有効でないと判定した場合、処理ロジックは、a を 1 に等しく、b を 0 に等しく設定し（処理ブロック 408）、処理ブロック 407 において、MVPb を $f_{pred}(MVPa, Ta, Tb)$ に等しく設定する。その後、プロセスは終了する。

【誤訳訂正 10】

【訂正対象書類名】明細書

【訂正対象項目名】0027

【訂正方法】変更

【訂正の内容】

【0027】

[0027]一実施形態では、候補プレディクタ改良フラグは、すべての候補プレディクタについて「オン」である。

【誤訳訂正 11】

【訂正対象書類名】明細書

【訂正対象項目名】0028

【訂正方法】変更

【訂正の内容】

【0028】

[0028]一実施形態では、候補プレディクタ改良フラグは、1 つ（又は複数）の制約が満たされた場合に限って「オン」になる。代替実施形態では、実施される制約は、以下のうちの 1 つ（又は複数）とすることができる。

1 .

【数 6】

$$\frac{MVP_a^T \times \overline{\overline{MVP_b}}}{\|MVP_a\|_2 \times \|\overline{\overline{MVP_b}}\|_2} > \overline{\overline{\delta}}$$

ここで、

【数 7】

$$\overline{\overline{\delta}}$$

は、所定のパラメータである。一実施形態では、

【数 8】

$$\overline{\overline{\delta}}$$

は、ゼロに等しい。

2 . MVPa × Ta の各要素の符号 (sign) が、

【数 9】

$$\overline{\overline{MVP_b}} \times \overline{T_b}$$

の対応する要素の符号と一致する。

3 . MVP a × T a の 1 つの要素の符号が、

【数 1 0】

$$\overline{\overline{MVP_b}} \times \overline{T_b}$$

の対応する要素の符号と一致する。

4 . MVP a の各要素の符号が、

【数 1 1】

$$\overline{\overline{MVP_b}}$$

の対応する要素の符号と一致する。

5 . MVP a の 1 つの要素の符号が、

【数 1 2】

$$\overline{\overline{MVP_b}}$$

の対応する要素の符号と一致する。

【誤訳訂正 1 2】

【訂正対象書類名】明細書

【訂正対象項目名】0 0 4 7

【訂正方法】変更

【訂正の内容】

【0 0 4 7】

[0047]図 1 0 A を参照すると、エンコーダにおいて動きベクトルを符号化するプロセスは、処理ロジックが、プレディクタ集合を構成する（処理ブロック 1 0 0 1）ことによって開始する。次に、処理ロジックは、プレディクタ集合を改良し（処理ブロック 1 0 0 2）、動きベクトルプレディクタを導出する（処理ブロック 1 0 0 3）。動きベクトルプレディクタを導出した後、処理ロジックは、差分動きベクトルを計算し（処理ブロック 1 0 0 4）、差分動きベクトルを符号化する（処理ブロック 1 0 0 5）。その後、プロセスは終了する。

【誤訳訂正 1 3】

【訂正対象書類名】明細書

【訂正対象項目名】0 0 4 8

【訂正方法】変更

【訂正の内容】

【0 0 4 8】

[0048]図 1 0 B を参照すると、デコーダにおいて動きベクトルを復号するプロセスは、処理ロジックが、プレディクタ集合を構成し（処理ブロック 1 0 1 1）、プレディクタ集合を改良し（処理ブロック 1 0 1 2）、動きベクトルプレディクタを導出する（処理ブロック 1 0 1 3）ことによって開始する。処理ロジックは、差分動きベクトルを復号し（処理ブロック 1 0 1 4）、導出された動きベクトルプレディクタを使用して、動きベクトルを再構成する（処理ブロック 1 0 1 5）。その後、プロセスは終了する。

【誤訳訂正 1 4】

【訂正対象書類名】明細書

【訂正対象項目名】 0 0 5 0

【訂正方法】 変更

【訂正の内容】

【 0 0 5 0 】

[0050] 図 1 1 を参照すると、動き情報メモリ 1 1 0 1 は、先に符号化されたブロックの属性を記憶している。目標ブロックと、動き情報メモリ 1 1 0 1 からの動き情報との受け取りに応答して、プレディクタ集合構成器 1 1 0 2 が、プレディクタ集合を構成する。一実施形態では、これは、上で図 2 において説明されたように構成される。プレディクタ集合を構成した後、プレディクタ集合改良器 1 1 0 3 が、プレディクタ集合を改良する。一実施形態では、プレディクタ集合の改良は、1 つ（又は複数）の候補プレディクタの改良を含む。一実施形態では、候補プレディクタの改良は、上で図 4 において説明されたように実行される。プレディクタ集合改良器 1 1 0 3 がプレディクタ集合を改良した後、プレディクタ生成器 1 1 0 4 が、動きベクトルプレディクタを導出する。一実施形態では、プレディクタ生成器 1 1 0 4 は、上で図 6 において説明されたように、動きベクトルプレディクタを導出する。プレディクタ生成器 1 1 0 4 は、導出された動きベクトルプレディクタを、差分動きベクトル生成器 1 1 0 5 に送り、差分動きベクトル生成器 1 1 0 5 は、動きベクトルプレディクタと、目標ブロックの動きベクトルとに응答して、差分動きベクトルを計算する。差分動きベクトル生成器 1 1 0 5 は、計算された差分動きベクトルを、エントロピエンコーダ (entropy encoder) 1 1 0 6 に送り、エントロピエンコーダ 1 1 0 6 は、差分動きベクトルを符号化し、それらをビデオビットストリームの一部に含める。その後、目標ブロックの属性は、記憶のために動き情報メモリ 1 1 0 1 に送られる（ブロック図の Z - 1 は、目標ブロックの動きベクトルが符号化されるまで、これが起こらないことを示す）。

【誤訳訂正 1 5】

【訂正対象書類名】 明細書

【訂正対象項目名】 0 0 5 1

【訂正方法】 変更

【訂正の内容】

【 0 0 5 1 】

[0051] 図 1 2 は、ビデオデコーダの一部である例示的な動きベクトルデコーダのブロック図である。図 1 2 を参照すると、動き情報メモリ 1 2 0 1 は、先に復号されたブロックの属性を記憶している。プレディクタ集合構成器 1 2 0 2 が、動き情報メモリ 1 2 0 1 からの動き情報と、目標ブロック情報とを受け取る。これらの入力に응答して、プレディクタ集合構成器 1 2 0 2 が、プレディクタ集合を構成する。一実施形態では、プレディクタ集合の構成は、上で図 2 において説明されたように実行される。プレディクタ集合構成器 1 2 0 2 は、プレディクタ集合を、プレディクタ集合改良器 1 2 0 3 に送り、プレディクタ集合改良器 1 2 0 3 は、プレディクタ集合を改良する。一実施形態では、プレディクタ集合の改良は、1 つ（又は複数）の候補プレディクタの改良を含む。一実施形態では、候補プレディクタの改良は、上で図 4 において説明されたように実行される。プレディクタ集合改良器 1 2 0 3 は、改良されたプレディクタ集合を、プレディクタ生成器 1 2 0 4 に送る。改良されたプレディクタ集合に응答して、プレディクタ生成器 1 2 0 4 は、動きベクトルプレディクタを導出する。一実施形態では、プレディクタ生成器 1 2 0 4 は、上で図 6 において説明されたように、動きベクトルプレディクタを導出する。プレディクタ生成器 1 2 0 4 は、導出された動きベクトルプレディクタを、動きベクトル生成器 1 2 0 6 に送る。エントロピデコーダ 1 2 0 5 が、ビデオビットストリームを受け取り、ビデオビットストリームに対してエントロピ復号を実行する。これが、復号された差分動きベクトルを含む、復号されたビットストリームを生成する。エントロピデコーダ 1 2 0 5 は、復号された差分動きベクトルを、動きベクトル生成器 1 2 0 6 に送る。導出された動きベクトルプレディクタと、復号された差分動きベクトルとに응答して、動きベクトル生成器 1 2 0 6 は、目標ブロックのための動きベクトルを再構成する。その後、動きベクトル生成

器 1 2 0 6 は、目標ブロックの属性に含まれるように再構成された動きベクトルを送る。最後に、目標ブロックの属性は、記憶のために動き情報メモリ 1 2 0 1 に送られる（ブロック図の Z - 1 は、目標ブロックの動きベクトルが再構成されるまで、これが起こらないことを示す）。

【誤訳訂正 1 6】

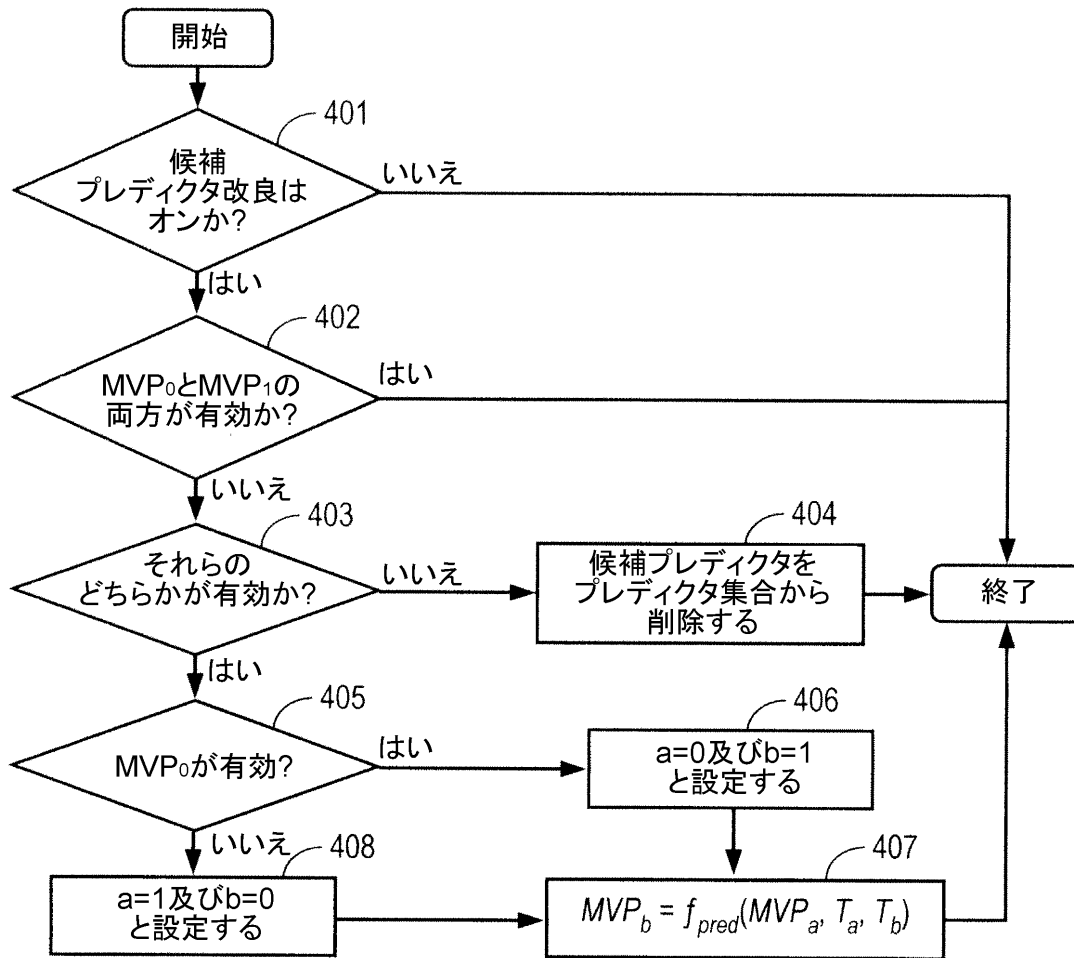
【訂正対象書類名】図面

【訂正対象項目名】図 4

【訂正方法】変更

【訂正の内容】

【 図 4 】



【 誤 訳 訂 正 1 7 】

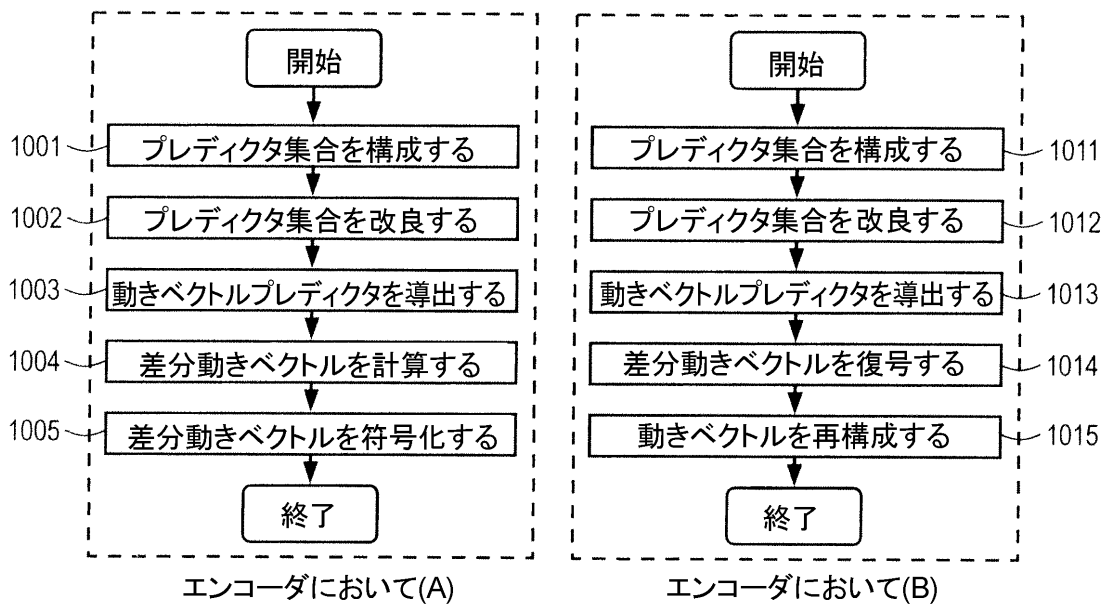
【 訂 正 対 象 書 類 名 】 図 面

【 訂 正 対 象 項 目 名 】 図 1 0

【 訂 正 方 法 】 変 更

【訂正の内容】

【図 1 0】



【誤訳訂正 1 8】

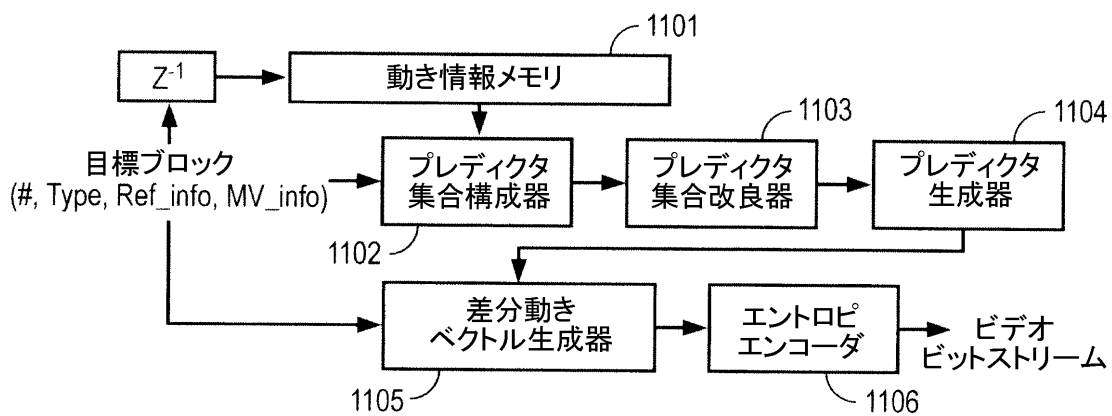
【訂正対象書類名】図面

【訂正対象項目名】図 1 1

【訂正方法】変更

【訂正の内容】

【図 1 1】



【誤訳訂正 1 9】

【訂正対象書類名】図面

【訂正対象項目名】図 1 2

【訂正方法】変更

【訂正の内容】

【図 1 2】

