

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4358436号
(P4358436)

(45) 発行日 平成21年11月4日 (2009. 11. 4)

(24) 登録日 平成21年8月14日 (2009. 8. 14)

(51) Int. Cl. F I
H O 4 N 7/30 (2006. 01) H O 4 N 7/133 Z
H O 4 N 7/32 (2006. 01) H O 4 N 7/137 Z

請求項の数 14 (全 43 頁)

(21) 出願番号 特願2000-542930 (P2000-542930)
(86) (22) 出願日 平成11年4月5日 (1999. 4. 5)
(65) 公表番号 特表2004-503951 (P2004-503951A)
(43) 公表日 平成16年2月5日 (2004. 2. 5)
(86) 国際出願番号 PCT/US1999/007474
(87) 国際公開番号 W01999/052297
(87) 国際公開日 平成11年10月14日 (1999. 10. 14)
審査請求日 平成18年3月23日 (2006. 3. 23)
(31) 優先権主張番号 60/080, 536
(32) 優先日 平成10年4月3日 (1998. 4. 3)
(33) 優先権主張国 米国 (US)
(31) 優先権主張番号 09/105, 730
(32) 優先日 平成10年6月26日 (1998. 6. 26)
(33) 優先権主張国 米国 (US)

(73) 特許権者 505131555
メディアテック インコーポレイション
台湾 300, シンチュ, サイエンス
ベースド インダストリアル パーク, イ
ノベーション ロード 1, ナンバー 1
- 2, 5階
(74) 代理人 100094318
弁理士 山田 行一
(74) 代理人 100123995
弁理士 野田 雅一
(72) 発明者 セチュラマン, スリラム
アメリカ合衆国, ニュー ジャージー州
, ハイットタウン, ケンジントン アー
ムズ アパートメンツ ナンバー 28-B
- 1

最終頁に続く

(54) 【発明の名称】 ビデオ情報を符号化する方法および装置

(57) 【特許請求の範囲】

【請求項 1】

複数の情報フレームを受信して処理し、これによって複数の処理済み情報フレームを発生するプリプロセッサであり、前記プリプロセッサは、前記複数の情報フレーム内でインターフレーム情報の不連続点を識別して当該インターフレーム情報の不連続点を示す印を発生し、前記複数の情報フレームにおける冗長な情報又は繰り返しの情報の存在を示す示唆を発生するプリプロセッサと、

前記複数の処理済み情報フレームの各々を符号化しフレームグループ (G O F) を生成するエンコーダと、

前記インターフレーム情報の不連続点を示す印に対応して、前記 G O F において、前記情報の不連続点の後の 1 つ目のアンカーフレームを I フレームとし、前記情報の不連続点の前の 1 つ目のアンカーフレームを P フレームとするとともに、前記プリプロセッサによって前記複数の情報フレームにおける冗長な情報又は繰り返しの情報の存在を示す示唆に対応して、当該冗長な情報又は繰り返しの情報を除去するコントローラと、

を備える符号化システム。

【請求項 2】

前記プリプロセッサが各情報フレームを空間的に分解して、個別の詳細ピラミッドを形成し、

前記エンコーダが、前記詳細ピラミッドを用いて、前記複数の処理済み情報フレームの非アンカーフレームの情報領域の動作を推定する動作推定モジュールを備える、請求項 1

10

20

に記載の符号化システム。

【請求項 3】

前記プリプロセッサが前記詳細ピラミッドの各々をそれぞれの圧縮されたデータ詳細ピラミッド表示物にパッキングし、

前記動作推定モジュールが、前記圧縮済みデータ詳細ピラミッド表示物を利用して動作を推定する、請求項 2 に記載の符号化システム。

【請求項 4】

前記エンコーダがさらに、

前記エンコーダが発生した各アンカーフレームを復号化するアンカーフレームデコーダと、

復号化された各アンカーフレームをそれぞれの圧縮済みデータ表示物にパッキングするデータバッカーと、

各圧縮済みデータアンカーフレーム表示物を記憶するメモリーと、

を備える、請求項 3 に記載の符号化システム。

【請求項 5】

前記コントローラが、二次レート歪みモデルに従って、前記 G O F に関するビット割り当てレベルを修正する、請求項 1 に記載の符号化システム。

【請求項 6】

前記エンコーダが複数の処理エレメントを備え、各処理エレメントが、処理済みの情報フレームのそれぞれの部分を符号化し、前記それぞれの部分が、符号化されている情報フレームのスライス、マクロブロック又はブロックの内の 1 つを含む、請求項 1 に記載の符号化システム。

【請求項 7】

各処理エレメントが、

符号化されている前記処理済み情報フレームの前記部分を基準情報フレームの対応する部分に関連付ける半画素動作ベクトルを発生する動作推定モジュールと、

符号化されている前記処理済み情報フレームの前記部分の複数のコーディングモードのうち 1 つを選択するモード選択モジュールであり、前記モード選択モジュールが前記半画素動作ベクトルを利用して前記コーディングモードを選択するモード選択モジュールと

を備える、請求項 6 に記載の符号化システム。

【請求項 8】

各処理エレメントがさらに、

符号化されている前記処理済み情報フレームの前記部分をコーディングする、離散コサイン変換モジュール、量子化モジュール及び可変長コーディングモジュールの直列カップリング物と、

前記量子化モジュールが発生した量子化済み情報ストリームに反応して前記基準情報フレームの部分を発生する、逆量子化モジュールと逆離散コサイン変換モジュールの直列カップリング物と、

前記処理済み情報フレームの前記部分を、前記動作推定モジュールからの前記半画素動作ベクトルと前記基準情報フレームとに従って予測する動作補償モジュールと

を備える、請求項 7 に記載の符号化システム。

【請求項 9】

複数の情報フレームを受信して処理し、これによって複数の処理済み情報フレームを発生するプリプロセッサであり、前記プリプロセッサは、前記複数の情報フレーム内でインターフレーム情報の不連続点を識別して当該インターフレーム情報の不連続点を示す印を発生し、前記複数の情報フレームにおける冗長な情報又は繰り返しの情報の存在を示す示唆を発生するプリプロセッサと、

複数の処理エレメントを含むエンコーダであって、前記処理エレメントの各々が前記複数の処理済み情報フレームの各々を符号化することで、符号化された前記複数の処理済み情報フレームによりフレームグループ (G O F) を形成する、当該エンコーダと、

10

20

30

40

50

前記インターフレーム情報の不連続点を示す印に対応して、前記 G O F において、前記情報の不連続点の後の 1 つ目のアンカーフレームを I フレームとし、前記情報の不連続点の前の 1 つ目のアンカーフレームを P フレームとするとともに、前記プリプロセッサによって前記複数の情報フレームにおける冗長な情報又は繰り返しの情報の存在を示す示唆に対応して、当該冗長な情報又は繰り返しの情報を除去するコントローラと、

を備える符号化システム。

【請求項 10】

前記処理エレメントの各々が、キャッシュメモリーに記憶された半画素動作ベクトルを利用して、前記複数の処理済み情報フレームの各々に対する符号化モードを選択する、請求項 9 に記載の符号化システム。

10

【請求項 11】

前記プリプロセッサが、符号化されている各それぞれの処理済み情報フレームと関連する複数の詳細ピラミッドの各々のパッキング済みデータ表示物を発生するプリプロセッサ用データバッカーを含み、

前記エンコーダに含まれた前記処理エレメントの各々が、符号化されたアンカーフレームからの基準情報フレームに関連する複数の詳細ピラミッドの各々についてのパッキング済みデータ表示物を発生するデータバッカーを含む、請求項 10 に記載の符号化システム。

【請求項 12】

映像フレームシーケンスを符号化する方法であって、

20

前記映像フレームシーケンスのある部分を処理することで、処理済み情報フレームシーケンスと、前記処理済み情報フレームシーケンス内のインターフレーム情報の不連続点を示す印と、前記複数の情報フレームにおける冗長な情報又は繰り返しの情報の存在を示す示唆と、を発生する処理工程と、

前記処理済み情報フレームシーケンスを、あるビット割当て量を持つフレームグループ (G O F) として符号化し、前記 G O F により定まるタイプのアンカーフレームを含んだ処理済み情報フレームシーケンスを生成する工程と、

前記インターフレーム情報の不連続点を示す印に対応して、前記 G O F において、前記情報の不連続点の後の 1 つ目のアンカーフレームを I フレームとし、前記情報の不連続点の前の 1 つ目のアンカーフレームを P フレームとする工程と、

30

前記複数の情報フレームにおける冗長な情報又は繰り返しの情報の存在を示す示唆に対応して、当該冗長な情報又は繰り返しの情報を除去する工程と

を含む、当該方法。

【請求項 13】

前記除去された冗長な情報又は繰り返しの情報に関連するビット割り当てレベルに従って、前記 G O F の前記ビット割当て量を増加させる工程、

をさらに含む、請求項 12 に記載の方法。

【請求項 14】

前記処理済み情報フレームシーケンスを形成する前記情報フレームの各々をそれぞれの詳細ピラミッド中に分解する工程をさらに含む、

40

前記処理済み情報フレームシーケンスを符号化する前記工程が、

各非アンカーフレームを形成する複数の情報フレーム領域の各々に対して、前記それぞれの詳細ピラミッドを用いてそれぞれのインターフレーム動作パラメータを推定する工程と、

各非アンカーフレームを形成する前記複数の情報フレーム領域の各々に対して、それぞれのインターフレーム動作パラメータからそれぞれのコーディングモードを決定する工程と、

前記それぞれのコーディングモードに従って、前記複数の情報フレーム領域の各々を符号化する工程と、

を含む、

50

各非アンカーフレームを形成する前記情報フレーム領域の内の少なくとも2つが、パイプライン式に動作するそれぞれの処理エレメントを用いて同時に処理される、請求項12に記載の方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本出願は、1998年4月3日に提出され、ここに参照して組み込まれる米国暫定出願第60/080,536号（代理人整理番号第12414P号）の恩典を請求するものである。

【0002】

この出願は、全て参照してここに組み込まれる、1998年6月26日に提出された米国特許出願第09/105,730号（代理人整理番号第12411号）、1998年9月11日に提出された第09/151,425号（代理人整理番号第12414号）及び1998年11月19日に提出された第09/196,072号（代理人整理番号第12626A号）の一部継続出願である。

【0003】

本発明は一般的には情報圧縮システムに関し、より特定的には、ビデオ供給源フォーマッティング及びコンテンツのパラメータ並びに符号化されたビデオフォーマッティングパラメータに従ってビデオ情報符号化システムを適応する方法と装置に関する。

【0004】

【従来の技術】

送信される予定のデータを、利用可能帯域幅がより効果的に用いられるように圧縮されるような通信システムがいくつか存在する。例えば、動画エクスパートグループ（MPEG）はデジタルデータ送出システムに関するいくつかの基準を発行してきた。その第1の基準は、MPEG-1として知られているがISO/IEC基準11172のことであり、参照してここに組み込まれる。MPEG-2として知られる第2の基準はISO/IEC基準13818のことであり、参照してここに組み込まれる。圧縮デジタルビデオシステムが、次世代テレビシステム委員会（ATSC）デジタルテレビ基準文書A/53に記載されており、ここに参照して組み込まれる。

【0005】

上に参照した基準は、デジタル通信システム内で固定長又は可変長のデータ構造を用いてビデオ情報、オーディオ情報及び他の情報を圧縮したり送出したりすることにすべてが適するデータの処理と操作を記載している。特に、上に参照した基準及び他の「MPEG様の」基準及び技法では、図示するように、フレーム内コーディング技法（例えば、ランレングスコーディング、ハフマンコーディングなど）とフレーム間コーディング技法（例えば、前方・後方予測コーディング、動作補償など）を用いてビデオ情報を圧縮する。具体的には、ビデオ処理システムでは、MPEG及びMPEG様のビデオ処理システムでは、ビデオフレームをフレーム内及び/又はフレーム間動作補償符号化あるなしにかかわらずビデオフレームを予測ベースで圧縮符号化することを特徴としている。

【0006】

ビデオ情報を符号化する際に、符号化プロセスをビデオ供給源フォーマッティングパラメータや符号化済みビデオフォーマッティングパラメータに適応するのが望ましいことが分かっている。不運にも、ビデオ符号化プロセスをこのように適応すると、処理パワーとメモリーの要件という点で極端に高価なものとなりやすい。このため、ビデオエンコーダの設計者は一般に、様々なパラメータを近似させて、計算上及び記憶上のオーバーヘッドを特定の応用分野（例えば、専門的な用途や消費者的な用途）で定められる「まともな」レベルにまで軽減する。

【0007】

したがって、ビデオ供給源フォーマッティングパラメータと符号化済みビデオフォーマッティングパラメータに対してビデオ符号化プロセスを適用し及び/又は強化するための口

10

20

30

40

50

バストで計算的に効果的な方法と装置を提供することによって上記の問題を処理するのが望ましいことが分かっている。具体的には、広範囲の符号化ビットレートにわたって洗練されたロバストなビデオ品質と共に比較的低い計算的複雑性とメモリー帯域幅の要件を提供する、MPEG-2ビデオ符号化方法と装置などのビデオ符号化方法と装置を提供することが望ましいことが分かっている。

【0008】

【課題を解決するための手段】

本発明は、図示するように、ビデオ情報ストリームを符号化して、フレームグループ(GOF)情報構造に従った符号化済み情報ストリームを発生するための方法と装置を含む。本発明の様々な実施形態では、ビデオ情報ストリームにおける、(例えば、3:2ブルダウン処理に起因する)情報の不連続性と冗長情報の存在にそれぞれ反応したGOF構造とビット予算に対する修正がなされる。本発明のさらなる実施形態ではピラミッド処理技法、パックデータ表示及び他の方法を用いて、符号化環境下でメモリーリソース及び/又は処理リソースを高度に効果的に利用するようにしている。

【0009】

具体的には、本発明による符号化システムは：実質的にフレームグループ(GOF)情報構造に従った複数の予備処理済み情報フレームの各々を符号化するエンコーダと、前記複数の情報フレーム内のフレーム間情報不連続性に反応してGOF情報構造を適応するコントローラと、複数の情報フレームを受信して処理して、複数の予備処理された情報フレームを発生するプリプロセッサであり、前記プリプロセッサが前記複数の情報フレーム内のフレーム間情報不連続性の印を識別して発生するプリプロセッサと、を含んでいる。

【0010】

【発明の実施の形態】

本発明の教示は添付図面と共に以下の詳細な説明を読めば容易に理解することが可能である。分かりやすいように、図面全体にわたって同一の部品には可能な限り同一の参照番号を用いている。

【0011】

以下の説明を検討すれば本発明の教示内容が、容易に利用できる情報符号化システム、特にビデオ情報符号化システムであり得ることが当業者に明瞭に理解されよう。

【0012】

MPEG様符号化システムの高レベルブロック図を図1に示す。図1のMPEG様符号化システム100は、複数の未圧縮情報フレームからなる入力情報ストリームIN、具体的にはビデオ情報ストリームを形成する一連の画像を受信して、フレームグループ(GOF)又は画像グループ(GOP)データ構造に公称的に従う符号化された出力情報ストリームOUTを生成する。各GOF/GOPデータ構造は、極大サイズのM個のフレームを持つ複数のサブGOF又はサブGOPデータ構造として配置されるN個のフレームから成る。各サブGOF/GOPはそれぞれのアンカーフレーム(例えば、Iフレーム又はPフレーム)、及び選択的に1つ以上の非アンカーフレーム(例えば、Bフレーム)を含む。

【0013】

具体的には図1のMPEG様符号化システム100は、第1プリプロセッサ101、第2プリプロセッサ102、コントローラ140、符号化モジュール104、及び出力バッファ160から成る。

【0014】

第1プリプロセッサ101は入力情報ストリームINを受けそれを処理して、予備処理をされた入力ビデオ情報ストリームINを生成する。第1プリプロセッサ101は以下の機能的構成部品を含む：3:2ブルダウンプロセッサ181、情景変化検出器182、雑音減少プレフィルタ183、及びサブGOPバッファ184である。第1プリプロセッサ101の作動方法を図2を参照して以下により詳細に説明する。

【0015】

3:2ブルダウンプロセッサ181は入力ビデオ情報ストリームINをフレーム毎に処理

10

20

30

40

50

し、それに応じて、入力ビデオ情報ストリームIN内における3:2プルダウン処理されたビデオ情報の存在(例えば、固有の24fps)又は不在(例えば、固有の30fps)を示す制御信号PDを本発明の1つの実施形態において生成する。3:2プルダウン指示制御信号PDはコントローラ140に結合する。3:2プルダウンプロセッサ181は、入力ビデオ情報ストリームIN内の「余分な」ビデオ情報の識別及び除去を支援することによって、符号化される情報の量を減少させるために用いられる。すなわちプルダウンが復元されて、その結果生じた情報フレームは、進行的な(すなわち非インターレースの)方法で符号化モジュール104によりコーディングされる。3:2プルダウンプロセッサ181の出力は情景変化検出器182に結合する。

【0016】

本発明の1つの実施形態では、3:2プルダウンプロセッサ181は「フィールド識別」方法を利用し、入力情報ストリームIN内の連続するフィールドを比べて、そのフィールドが内在するビデオ情報の3:2プルダウン処理を示すような方法で繰り返されているかどうかを確認する。この実施形態ではプルダウンプロセッサ181は制御信号R__TOP及び制御信号R__BOTを生成する。制御信号R__TOPは、ビデオフレームの頂部フィールド内のビデオ情報が直前のビデオフレームの頂部フィールド内のビデオ情報と事実上同じであるかどうかを示すために用いられる。同様に制御信号R__BOTは、ビデオフレームの底部フィールド内のビデオ情報が直前のビデオフレームの底部フィールド内のビデオ情報と事実上同じであるかどうかを示すために用いられる。

【0017】

例示の3:2プルダウンプロセッサ及び制御機能は、1998年9月11日出願の米国特許出願第09/151,425号(代理人整理番号第12414号)において開示されており、その全てを参照してここに組込む。フィールド同士の比較を実行するのに要する計算量は、絶対差合計(SAD)法を用いて、連続する情報フレームの(頂部又は底部)フィールドの部分的フィールド比較だけを行うことによって選択的に減少できる。

【0018】

情景変化検出器182は、ビデオ情報ストリームIN(又は3:2プルダウン処理された入力ビデオ情報ストリームIN)内の情景変化又は情景切り取りのような、情報の不連続性を識別するために用いられる。情報の不連続性の検出に応答して、情景変化検出器182は不連続性を示す第1制御信号SCN及び第2制御信号SCPを生成する。具体的には、第1及び第2の情景変化指示制御信号SCN及びSCPはコントローラ140に結合されて、次の(すなわち未だ符号化されていない)サブGOP(SCN)における情景変化、及び現在の(すなわち現在符号化されつつある)サブGOP(SCP)における情景変化をそれぞれ示すために用いられる。すなわち情景変化検出器の作動は、2つのフレーム間の相違が内在ビデオ内で発生した情景変化による相違かどうかを確認するために2つの情報フレームが比較されるということに基づいたルックアヘッドである。

【0019】

雑音減少プレフィルタ183は、適応可能な非線型雑音減少フィルタから成り、それはガウスの雑音、散弾雑音及びテープ雑音等の異なるタイプの雑音を減少させる一方で、ビデオ情報の場合には高い忠実度を持つエッジ及び他のハイコントラスト詳細等の高周波情報を保持する。雑音減少プレフィルタ183は、フレーム毎に、又はサブGOPバッファ184内に記憶されている全サブGOP上で動作する。当業者は1つ以上の周知の非線型雑音減少フィルタを適切に本発明に適用することが容易にできよう。例えば有限インパルス応答中央ハイブリッド(FMH)フィルタを用いて図1の雑音減少プレフィルタ183を実現することができる。

【0020】

サブGOPバッファ184は前記のように、サブGOPを形成するいかなるアンカーフレーム及びいかなる関連の非アンカーフレームをも保存できる。第1プリプロセッサの様々な工程(すなわち、3:2プルダウン処理、情景変化検出、及びフィルタリング)の完了後、処理されたサブGOPは予備処理を施された入力ビデオ情報ストリームINとして

10

20

30

40

50

、第2プリプロセッサ102及び符号化モジュール104に結合される。

【0021】

コントローラ140は、入力情報ストリームIN、3:2プルダウン指示制御信号PD、R__TOP及びR__BOT、及び情景変化指示制御信号SCN及びSCPを受信し処理して、符号化モジュール104を制御するために用いられる様々な制御信号を生成する。具体的にはコントローラ140は以下の制御信号を生成する：画像タイプ制御信号PTは、符号化されているサブGOPのアンカーフレームGOPの所望のフレームタイプ（すなわち、P-フレーム又はI-フレーム）を指示するために用いられる。GOP構造制御信号GSは、GOP構造（例えば、図2を参照して以下に説明されるような変数nPu、nBu、nBo等）へのいかなる変化をも指示するために用いられる。進行的フレーム制御信号PFは、情報フレームの進行的符号化を指示するために用いられる。「頂部フィールド第1/繰り返し第1フィールド」制御信号RBFは、符号化されるフレームが繰り返される素材を含み従って繰り返し素材無しに符号化できることを指示するために用いられる。フレームレート制御信号FRは、符号化されるビデオ情報ストリームの所望のフレームレート（すなわち、秒当たりのフレーム数）を指示するために用いられる。さらに、量子化モジュール制御信号CQ及び離散コサイン変換モジュール制御信号CDがコントローラ140によって生成される。

10

【0022】

（1つ以上の制御信号SCN及びSCPによって指示されるように）入力情報ストリームIN内のフレーム間の情報不連続性に応答して、情報の不連続性に引き続く第1のアンカーフレームはIフレームから成り、一方情報の不連続性に先行する第1のアンカーフレームはPフレームから成るように、コントローラ140は符号化された出力情報ストリームOUTを適応させる。各GOF/GOP構造に割当てられるビット予算内のビットは、新たなGOF/GOPの開始直前のフレームの内部を符号化することによって「浪費」されない。

20

【0023】

（1つ以上の制御信号PD、R__TOP、R__BOTによって示されるように）入力情報ストリームIN内に3:2プルダウン処理されたビデオ素材が存在するとそれに応答して、コントローラ140は入力情報ストリームINから「過剰の」又は重複するビデオ情報を取り除き、それによって情報ストリームINの残余の部分を符号化するために用いられるビット数を増加させる。

30

【0024】

このようにしてコントローラ140は3:2プルダウンプロセッサ181及び情景変化検出器182と協力することによって、符号化モジュール104は（3:2プルダウン処理された素材を伴った又は伴わない）情景変化にわたって円滑かつ高性能な符号化を行うことができるようになる。この様な方法によって、符号化されたビデオストリームOUT内の情景変化後の視覚的アーチファクトが、比較的低いビットレートにおいても著しく減少する。さらにプルダウンされた素材が、低い実フレームレートを用いることによってより高性能でコーディングされる。

【0025】

符号化モジュール104は予備処理された入力情報ストリームINを受信し符号化して、フレームグループ（GOF）又は画像グループ（GOP）データ構造に実質的に適応する符号化された情報ストリームS104を生成する。符号化された情報ストリームS104は出力バッファ160に結合する。コントローラ140によって生成された制御信号C1及び出力バッファ160によって生成されたレート制御信号RCに応答して、符号化モジュール104は、GOF又はGOPデータ構造、ビット割当て、量子化スケール及び符号化処理の他のパラメータに適応する。符号化モジュール104は、処理されるマクロブロックのために選択される符号化モード（すなわち、フレーム内符号化されるか又はフレーム間符号化されるか）を指示するモード判断信号MDを生成する。さらに符号化モジュール104内の動作推定モジュール150は、第2プリプロセッサ102によって生成さ

40

50

れた幾つかの制御信号に応答する。

【 0 0 2 6 】

符号化モジュール 1 0 4 は、減算器 1 5 5、モード判断モジュール 1 0 5、離散コサイン変換 (D C T) モジュール 1 1 0、量子化 Q モジュール 1 1 5、可変長コーディング (V L C) モジュール 1 2 0、逆量子化器 (Q^{-1}) 1 2 5、逆離散コサイン変換 (DCT^{-1}) モジュール 1 3 0、選択的データパッカー 1 3 2、減算器 1 5 6、動作補償モジュール 1 4 5、動作推定モジュール 1 5 0、及びアンカーフレーム記憶モジュール 1 7 0 から成る。レート制御機能はレートコントローラ 1 4 0 によって実行される。

【 0 0 2 7 】

減算器 1 5 5 は、信号経路 I N 上の入力マクロブロックから信号経路 P F 上の予測されるマクロブロックを減じることによって、残余信号 (また当技術において単に残余又は残余マクロブロックとも呼ばれる) を発生する。

【 0 0 2 8 】

モード判断モジュール 1 0 5 は、減算器 1 5 5 から残余マクロブロック (すなわち、予測されるマクロブロック) を、又信号経路 I N から入力マクロブロックを受取る。予測マクロブロックが入力マクロブロックと実質的に類似している (すなわち、残余は相対的に小さく、非常に少ないビットを用いて容易にコーディングできる) 場合は、モード判断モジュール 1 0 5 はフレーム間符号化のために減算器 1 5 5 から残余信号を選択する。すなわち、マクロブロックは補償されたマクロブロックすなわち動作ベクトル及び関連する残余として符号化される。しかしながら予測マクロブロックと入力マクロブロックとの間の差がかなりある場合には、その残余をコーディングするのは高くつく。従ってシステムは、動作補償された残余マクロブロックを符号化するよりも入力マクロブロックを直接符号化することによってより効率的に動作する。

【 0 0 2 9 】

上記の選択処理はコーディングモードの選択として周知である。入力マクロブロックの符号化はフレーム内コーディングと呼ばれ、残余のコーディングはフレーム間コーディングと呼ばれる。これら 2 つのモード間の選択はフレーム内 - フレーム間 - 決定 (I I D) として周知である。I I D は一般に、先ず残余マクロブロック (V a r R) の変化、及び入力マクロブロック (V a r I) の変化を計算することによって計算される。コーディングの決定はこれらの値に基づく。この決定を行うために用いることができる幾つかの関数がある。例えば、最も簡単な関数を用いると、V a r R が V a r I よりも小さければ I I D はフレーム間モードを選択する。逆に、V a r I が V a r R よりも小さければ I I D はフレーム内モードを選択する。

【 0 0 3 0 】

代りに、図 1 の M P E G 様エンコーダ 1 0 0 内での使用に適したモード判断方法が、1 9 9 5 年 1 0 月 2 6 日出願の米国特許出願第 0 8 / 5 4 7 , 7 4 1 号 (代理人整理番号第 1 1 8 1 9 号) に開示されており、それを参照してここに組込む。簡単に言えば、第 0 8 / 5 4 7 , 7 4 1 号出願には、各マクロブロックをコーディングするために要する全ビット数は 2 つの部分からなり、1 つは動作ベクトルをコーディングするのに必要なビットでありもう 1 つは予測残余をコーディングするのに必要なビットである。動作ベクトルのコーディングのためのビット数は一般にルックアップテーブルから得られる。予測残余をコーディングするためのビット数は、その変化の値に正比例し、量子化ステップ (量子化のスケール) の値に逆比例するという想定によって得られる。この想定を用いれば、マクロブロックをコーディングするのに必要な全ビット数が各コーディングモードに対して計算されて比較される。最小ビット数でコーディングモードを選択することにより、本発明は実際に実用化するための複雑さの度合いの低い最適に近い解法を得る。

【 0 0 3 1 】

選択されたブロック (すなわち、予備処理された入力マクロブロック又は残余マクロブロック) はその後 D C T モジュール 1 1 0 に結合される。D C T モジュール 1 1 0 はレートコントローラ 1 4 0 によって生成された制御信号 C D に応答して、選択されたブロックを

処理し、例えばDCT係数の例えば 8×8 ブロックの1セットを生成する。DCTモジュール110によって生成されるDCT係数は量子化モジュール115に結合される。

【0032】

量子化モジュール115はレートコントローラ140によって生成された制御信号CQに
応答して、受信されたDCT係数を量子化し、量子化された出力ブロックを生成する。量子化のプロセスによって、整数値を作るための適当な丸めを伴う量子化値の1セットによりDCT係数を割ることによってDCT係数を表す際の精度が落ちる。量子化されたDCT係数は可変長コーディング(VLC)モジュール120及び逆DCTモジュール125に結合される。

【0033】

それからVLCモジュール120は周知の可変長コーディング及びランレングスコーディング方式を用いて、受信され量子化されたDCT係数(例えば、 8×8 ブロックの量子化されたDCT係数)及びマクロブロックについての副次情報を符号化する。

【0034】

動作の予測及び補償を行うために、符号化モジュール104は参照フレームとして用いるための符号化されたアンカーフレームを再び生成する。この機能は逆量子化モジュール125、逆DCTモジュール130及び選択的にデータパッカー132によって実行される。

【0035】

具体的には、逆量子化(Q^{-1})モジュール125は量子化されたDCT係数(例えば、 8×8 ブロックの量子化されたDCT係数)を処理して、それぞれのDCT係数を生成する。逆DCTモジュール130はそれぞれのDCT係数を処理して、復号化された誤差信号を生成する。DCT $^{-1}$ モジュール130によって生成された誤差信号は加算器156の入力に結合される。DCT $^{-1}$ モジュール130はまた、引き続く符号化のための参照フレームとして用いるためにデータを符号化することによって、入力映像シーケンスのIフレーム及びPフレームを再生成するために機能する。この参照情報は情報ストリームA INとしてアンカーフレームメモリー170に結合される。データパッカー132は、その情報をアンカーフレームメモリー170内に記憶するのに先立って、参照情報ストリームA INをパックするために選択的に用いられる。

【0036】

動作推定モジュール150は予備処理された入力情報ストリームI N 及び記憶されたアンカーフレーム情報ストリームA O U Tを受取る。記憶されたアンカーフレーム情報ストリームA O U Tはアンカーフレーム記憶モジュール170によって生成され、それは逆DCTモジュール130によって生成された入力アンカーフレーム情報ストリームA INを記憶する。簡単に言えば、記憶されたアンカーフレーム情報ストリームA O U Tは、符号化モジュール104によって現在符号化されているG O F又はG O Pのフレーム間コーディングされたアンカーフレームの復号化されたバージョンを表す。

【0037】

動作推定モジュール150は、予備処理された入力情報ストリームI N 及び記憶されたアンカーフレーム情報ストリームA O U Tを用いて、動作ベクトルを推定する。動作ベクトルは2次元のベクトルであり、現在の画像内のブロックの座標位置からのオフセットを行って参照フレーム内に適応するための動作補償に用いられる。参照フレームは、前方予測されコーディングされたフレーム(Pフレーム)又は双方向(すなわち、前方及び後方)予測されたフレーム(Bフレーム)であり得る。動作ベクトルは動作補償モジュール145及びVLCモジュール120に結合される。

【0038】

動作推定モジュール150は、全画素動作推定器151及び半画素動作推定器152を含むように選択的に実現される。全画素動作推定器151は全画素動作ベクトルF P M V及び全画素歪み信号(すなわち、2つの画素ブロック間の相違の程度)F P Dを応答的に生成する。半画素動作推定器152は、全画素動作ベクトルF P M V及び全画素歪み信号F

10

20

30

40

50

P Dを用いて、半画素動作ベクトルH P M V及び半画素歪み信号H P Dを応答的に生成する。選択的な半画素動作ベクトルH P M V及び半画素歪み信号H P Dはモード判断モジュール1 0 5に結合されるものとして表され、そこでそれらはモード判断の精度を増大させるために効果的に利用される。

【 0 0 3 9 】

動作補償モジュール1 4 5は動作ベクトルを受取ってサンプル値の予測効率を高める。動作の補償は動作ベクトルを用いた予測を伴い、予測の誤差を形成するために用いられる先行して復号化されたサンプル値を含む過去及び／又は未来の参照フレーム内にオフセットを施す。具体的には、動作オフセットモジュール1 5 0は先行して復号化されたアンカーフレーム及び動作ベクトルを用いて、現在のフレームの推定を組立てる。さらに、動作推定モジュール及び動作補償モジュールによって実行される機能は、復号モジュール例えば単一ブロックの動作補償器内で実現可能であることが当業者に理解されよう。

10

【 0 0 4 0 】

所与のマクロブロックに対して動作補償予測を実行するのに先立って、コーディングモードが選択されなければならない。コーディングモード判断の分野においては、M P E G及びM P E G様の標準が複数の異なるマクロブロックのコーディングモードを提供する。具体的にはM P E G - 2は、フレーム内モード、無動作補償モード(N o M C)、前方／後方／平均フレーム間モード、及びフィールド／フレームD C Tモードを含むマクロブロックコーディングモードを提供する。

20

【 0 0 4 1 】

ひとたびコーディングモードが選択されれば、動作補償モジュール1 4 5は、過去及び／又は未来参照画像に基づくブロックのコンテンツの経路P F上に、動作補償された予測フレーム(例えば、予測された映像)を生成する。経路P F上のこの動作補償された予測フレームは、減算器1 5 5を通じて、現在のマクロブロック内の入力情報フレームI N (例えば、ビデオ映像)から減算されて、誤差信号又は予測残余信号を形成する。予測残余信号を形成することによって入力ビデオ映像内の冗長情報が効果的に取除かれる。前記のように、予測残余信号はさらなる処理のためにモード判断モジュール1 0 5に結合される。

【 0 0 4 2 】

V L Cエンコーダ1 2 0によって生成されたV L Cデータストリームは、バッファ1 6 0、具体的にはG O Pデータ構造に従って少なくとも1つ以上の符号化サブG O Pを保持することができる「先入先出」(F I F O)バッファ内で受け取られる。出力バッファ1 6 0はその利用レベルを示すレート制御信号R Cを生成する。

30

【 0 0 4 3 】

異なる画像タイプの使用及び可変長コーディングの結果、出力バッファ1 6 0内への全ビットレートが変更可能となる。すなわち、各フレームのコーディングに使われるビット数が異なり得る。出力情報ストリームO U Tを例えば記憶媒体又は通信チャネルに結合するための固定レートチャネルを含む応用例においては、出力バッファ1 6 0は、エンコーダ出力をビットレート平滑化のためのチャネルに一致させるのに用いられる。従って出力バッファ1 6 0の出力信号O U Tは、予備処理済入力情報ストリームI N の圧縮された表現である。

40

【 0 0 4 4 】

図1に示されるレート制御モジュール1 4 0は、マイクロプロセッサ1 4 0 - 4とともに、同時符号化 - 適応フレーム切替方式2 0 0及び／又は「オンデマンド」符号化 - 適用フレーム切替方式4 0 0を記憶するためのメモリー1 4 0 - 8を含む。マイクロプロセッサ1 4 0 - 4は、電源、クロック回路、キャッシュメモリー等の従来型サポート回路1 4 0 - 6とともに、ソフトウェア体系を支援する回路と協同する。従って、ソフトウェア処理等のここで議論された処理工程のいくつかは、例えばマイクロプロセッサ1 4 0 - 4と協同して様々な工程を実行する回路のようなハードウェアの中で実行できると考えられる。コントローラ1 4 0はさらに、様々な符号化モジュール(1 0 4及び1 0 2)と選択器(

50

104) 間のインタフェースを形成する入出力回路140-2を含む。コントローラ140は本発明による様々な制御機能を実行するようにプログラムされた汎用コンピュータとして示されているが、本発明はまた特定用途向け集積回路(ASIC)としてのハードウェア内でもまた実現可能である。従ってここで(例えば、図2Aから2Cに関連して)説明された全てのプロセス又は方法の工程は、ソフトウェア、ハードウェア、又はそれらの組合せによって同等に実行されるものと広義に解釈されることを意図したものである。

【0045】

レート制御モジュール140の主要なタスクは出力バッファ160の十分さ又は利用レベルを管理することであり、その出力バッファから一定の出力ビットレートが送信チャネルへ与えられる。符号化レートが、各映像及び映像の連続のコンテンツに依存して著しく変化したとしても、一定のビットレートは維持されなければならない。出力バッファ160は、その利用レベルを示すレート制御信号RCを生成する。

10

【0046】

レート制御モジュール140はレート制御信号RCを用いて、(制御信号CQ経由の)DCT係数の量子化に用いられる量子化スケール(工程のサイズ)及び/又はシステムによりコーディングされた制御信号CD経由のDCT係数の数のような、符号化処理の様々なパラメータに適應する。すなわち、画像の全シーケンスにわたる均等な視覚的特性を維持しつつ画像に対する目標ビットレートが達成されるように、各フレームについて量子化器のスケールが選択される。この様にして、レート制御モジュール140は、VLC120によって生成される出力情報ストリームのビットレートを制御し、それによって出力バッファ160の適切な利用レベルを維持する。

20

【0047】

本発明の1つの実施形態では、レート制御モジュール140は二次レート-歪みモデルを用いたフレームレベル目標決定方式に従って動作し、それは1996年10月23日出願の米国特許出願第08/738,228号(代理人整理番号第11835号)の中で説明されており、参照してここに組込まれる。簡単に言えば、開示されている二次レート歪みモデルはQにおける二次モデルを提供して、使用されるビット数をモデル化するが、標準のTM-5レート制御モデルは一次モデルを利用する。すなわち、先行して符号化された情報フレーム部分(例えば、フレーム、スライス又はマクロブロック)に割当てられたビット数を大まかに調べて先行符号化部分の複雑さのレベルを決定し、それに応じて符号化される現在の又は引き続く部分の複雑さレベルを決定するレートコントローラによって、レート制御が改善される。従ってより高精度のビット利用モデルによって、ビット予算をより高精度に管理することができる。

30

【0048】

レート制御モジュール140の別の重要なタスクは、エンコーダによって生成されるビットストリームが、出力情報ストリームOUTから成る送信を受取る(例えば、図示されない受信機又は目標記憶デバイス内の)デコーダ内の入力バッファにおけるオーバーフロー又はアンダーフローが無いことを保証することである。オーバーフロー及びアンダーフローの制御は、エンコーダ内の仮想的バッファを維持しかつ監視することによって行われる。仮想的バッファはビデオバッファリング検証器(VBV)として知られる。正しいデコーダ入力バッファビットの制御を保証するために、エンコーダのレート制御プロセスによって、各画像さらには各画像を構成する画素の各マクロブロックについて、ビット割当量(ここではビット予算とも呼ばれている)が確立される。それぞれのビット予算内にあるそれぞれのビット数を用いてブロック及び全体の画像をコーディングすることにより、VBVはオーバーフロー又はアンダーフローしない。VBVはデコーダの入力バッファを反映するので、VBVがアンダーフロー又はオーバーフローしなければ、デコーダの入力バッファはアンダーフロー又はオーバーフローしない。

40

【0049】

図2Aと図2Bはともに図1のMPEG様符号化システムに用いるのに適應したレート制御方法200の工程系統図を表す。図2Aと図2Bは、図2Aに概略的に表される方法で

50

配置されて方法200の全体が見られるように意図されている。以下の議論では一般に次のことが前提となっている、すなわちフレームレートに関わりなくMPEG様符号化システム100は動作して、所定の画像グループ(GOP)情報構造に従って受信済ビデオ情報ストリームINを符号化するが、その構造においては各GOPには所定のビット数(R)が割り当てられて、一定のビットレート出力ストリームOUTが出力バッファ160により生成される。すなわち、2つのIフレーム間の距離Nと2つの連続した参照フレーム間の距離Mは、符号化のプロセスにわたって同一に保たれている。さらに、1つのモード変化の極大点がいかなる2つの連続する参照フレーム間にも存在することが前提されている。

【0050】

10

図2Aと2Bのレート制御方法200は、受信済ビデオ情報ストリームIN内のモード変化(すなわち、24fps→30fps)の検出にตอบสนองして、TM-5レート制御アルゴリズムによって用いられるレート制御パラメータ(R)を変化させて、GOPビット割当て内に残っているビット数を指示する。この様にして、TM-5レート制御アルゴリズムは、そのようなモード変化に応じて適切に機能するように作られる。

【0051】

例えば図1のMPEG様符号化システム100が符号化のためにビデオ情報ストリームINを受取ったときに、方法200はステップ205に入れられる。方法200はそれからステップ210に進み、GOPシーケンスの第1GOPの処理に先立って、R_{SLACK}がゼロに初期化される。変数R_{SLACK}は先行して符号化されたGOPのビットのオーバー/アンダーランを指示するために使われる。すなわちR_{SLACK}は、先行GOPが割当てられたよりも少ないビット(アンダーラン)又は多いビット(オーバーラン)で符号化されことを指示する。従って現在処理されているGOPは、先行するオーバー/アンダーランを(必ずしもオフセットによってではないが)オフセットする傾向があるようなやり方で符号化される。それから方法200はステップ212に進む。

20

【0052】

ステップ212では、3つの変数(n20、n30及びGOP_{BITS})がゼロに初期化される。第1の変数n20は、24fps動作モードで符号化される処理中のGOP内のフレーム数を示すために用いられる。第2の変数n30は、30fps動作モードで符号化される処理中のGOP内のフレーム数を示すために用いられる。第3の変数GOP_{BITS}は、処理中のGOP内のビデオフレームを符号化するためにこれまで使われた全ビット数を表示する。この情報は、符号化中のビデオストリームのフォーマットの変化によって生じる動作モード間の遷移中にGOPの持続時間を動的に更新するために利用される。それから方法200はステップ215に進む。

30

【0053】

ステップ215では、受信済ビデオ情報ストリームINのモードについての照会が行われる。すなわち入力ビデオストリームINは現在、固有の30fpsビデオストリーム(例えば、「標準の」NTSCストリーム)から成るか、又は固有の24fpsビデオストリーム(例えば、30fpsストリームを形成するための3:2プルダウン処理に従って処理される24fpsの動画)から成るかについての照会が行われる。受信済情報ストリームのモードを識別するための例示の方法を図3、4、及び5を参照して以下に説明する。

40

【0054】

ステップ215での照会で、受信済ビデオ情報ストリームINが30fpsビデオ素材から成ることが示された場合、方法200はステップ225に進み、変数FRAME RATEが30に初期化される。方法200はそれからステップ230に進む。ステップ215の照会で、受信済ビデオ情報ストリームINが24fpsビデオ素材から成ることが示された場合は、方法200はステップ220に進み、そこで変数FRAME RATEが24に初期化される。それから方法200はステップ230に進む。

【0055】

ステップ230では、処理すべき画像グループ(GOP)内に残っているビット数(R)

50

が(次の)等式1に従って計算され、TM-5反応パラメータ(r)が(次の)等式2に従って計算される。ここで、

“R”は、現在のGOPに対するビット割当て内に残るビット数であり、

“R_{SLACK}”は、先行して符号化されたGOPのビットのオーバー/アンダーランであり、

“N_P”は、現在のGOP内に残るPフレームの数であり、

“N_B”は、現在のGOP内に残るBフレームの数であり、

“BITRATE”は、符号化された結果のビットストリームのビットレートであり、

“FRAMERATE”は、ステップ215から225について24又は30であり、及び

“r”は、TM-5レート制御アルゴリズムで用いられる反応パラメータである。

【0056】

【式1】

$$R = R_{SLACK} + \{(1 + N_P + N_B) \text{ BITRATE} \} / \text{FRAMERATE} \quad (\text{式1})$$

【0057】

【式2】

$$r = 2 \times \text{BITRATE} / \text{FRAMERATE} \quad (\text{式2})$$

【0058】

現在のGOP(R)に対するビット割当て内に残るビット数と反応パラメータ(r)を計算した後、変数GOP_{START}が計算された値Rに等しく設定されて、方法200はステップ235に進み、そこでモードの変化が発生したかどうか(すなわち、24fps 30fps又は30fps 24fps)についての照会がなされる。ステップ235における照会で、24fpsモードから30fpsモードへのモード変化が示された場合は、方法200はステップ236へ進む。ステップ235における照会で、30fpsモードから24fpsモードへのモード変化が示された場合は、方法200はステップ237へ進む。ステップ235における照会でモード変化が起こらなかったことが示された場合は、方法200はステップ238へ進む。

【0059】

ステップ238において変数Rは、GOP_{START} - GOP_{BITS}に等しくなるように設定される。すなわちGOPビット割当て(R)内に残るビット数は、初期ビット割当て(GOP_{START})マイナス使用されたビット数(GOP_{BITS})に等しくなるように設定される。それから方法200はステップ260へ進む。

【0060】

ステップ260では、現在処理されているGOP内の現在処理されているビデオフレームを形成する各マクロブロックについて、量子化パラメータ(s)が計算される。それから方法200はステップ262へ進む。

【0061】

ステップ262では、例えばTM-5レート制御アルゴリズムに従ってフレームが符号化される。符号化されるフレームタイプを決定するのに用いられる画像グループ(GOP)構造が、図2Cの方法に従って選択的に変更される。以下にそれをより詳細に説明する。

【0062】

方法200はそれからステップ265に進み、そこで変数N_P(現在のGOP内に残るPフレームの数)及びN_B(現在のBフレーム内に残るBフレームの数)の値が必要に応じて調整される。方法200はそれからステップ266へ進む。

【0063】

10

20

30

40

50

ステップ 266 では、現在符号化されているモードについての、すなわちステップ 262 で符号化されたフレームは 24 fps 又は 30 fps 固有モードフレームのどちらであるかについての、照会が行われる。ステップ 266 での照会の結果、ステップ 262 で符号化されたフレームが 30 fps フレームだった場合は、方法 200 はステップ 267 へ進み、そこで変数 n_{30} が 1 つ増分される。それから方法 200 はステップ 270 に進む。ステップ 266 での照会の結果、ステップ 262 で符号化されたフレームが 24 fps だった場合は、方法 200 はステップ 268 に進み、そこで変数 n_{20} が 1 つ増分される。方法 200 はそれからステップ 270 に進む。

【0064】

ステップ 270 で、変数 GOP_{BITS} が調整される。変数 GOP_{BITS} は、処理されている GOP 内のビデオフレームを符号化するためにこれまで使われた全ビット数を表示する。従って変数 GOP_{BITS} は、処理されている GOP 内の直近のフレームを処理するのに使われる（ヘッダ情報及び他のビット消費情報を含む）ビットの量によって増大させられる。方法 200 はステップ 275 に進む。

【0065】

ステップ 275 では、直近に符号化されたフレームが処理中の GOP の最後のフレーム（すなわち、GOP フレームの終点）か否かについて照会が行われる。ステップ 275 での照会の結果が否定的であれば、方法 200 はステップ 215 に進む。

【0066】

ステップ 275 での照会の結果が肯定的であれば、方法 200 はステップ 280 に進み、そこで変数 R_{SLACK} が R に等しくなるように設定される。 R は処理されている GOP に対するビット割当て内に残るビット数を表示するので、さらに GOP は現在処理されているので、 R に対するゼロ以外の全ての値は、割当てられたビット予算の利用不足（ $R > 0$ ）又は利用過剰（ $R < 0$ ）を表す。ビット割当てのこのオーバー/アンダーフローは先行して符号化された GOP から残っている全てのオーバー/アンダーフローに加えられて、符号化プロセスのあいだ利用可能な帯域幅が最大限に利用される。例えば利用不足のビット予算は、次の GOP の I フレームを増加したビット数で符号化することによって、及び符号化処理を質的に強化するための他の周知の方法によって、用いることができる。それから方法 200 はステップ 212 へ進む。

【0067】

ステップ 236 及び 237 の双方において、3 つの変数の値が計算される。具体的には、第 1 の変数 n_{Bu} は、新しい動作モード（すなわち、モード変化の検出後に導入されたモード）でコーディングされるために残留する B フレームの数を示す値に等しく、第 2 の変数 n_{Pu} は、新しい動作モードでコーディングするために残留する P フレームの数を示す値に等しく、第 3 の変数 n_{Bo} は、古い動作モード（すなわち、モード変化検出以前のモード）でコーディングされるために残留する B フレームの数を示す値に等しい。ステップ 236 の実行後、方法 200 はステップ 240 に進む。ステップ 237 の実行後、方法 200 はステップ 250 に進む。

【0068】

ステップ 240 では、現在処理中の GOP 内に残っているビット数（ R ）が（次の）等式 3 によって計算され、一方ステップ 250 では、現在処理中の GOP 内に残っているビット数（ R ）が（次の）等式 4 によって計算される。ここで、“ n_{20} ” は、24 fps 動作モード中に符号化された現在の GOP 内のフレーム数であり、

“ n_{30} ” は、30 fps 動作モード中に符号化された現在の GOP 内のフレーム数であり、

“ n_{Bu} ” は、（モード変化検出後の）新しいモードでコーディングされるために残留する B フレームの数であり、

“ n_{Pu} ” は、新しいモードでコーディングされるために残留する P フレームの数であり、および

“ n_{Bo} ” は、古いモードでコーディングされるために残留する B フレームの数である。

【 0 0 6 9 】

【 式 3 】

$$R = \text{B I T R A T E} \times (\text{n30} + \text{nPu} + \text{nBu}) / 30 + \text{n20} / 20 + \text{nBo} / 24 \\ + R_{\text{SLACK}} - \text{G O P}_{\text{BITS}} \quad (\text{式 3})$$

【 0 0 7 0 】

【 式 4 】

$$R = \text{B I T R A T E} \times (\text{n30} + \text{NBo}) / 30 + \text{n20} / 20 + \text{nPu} + \text{nBu} / 24 \\ + R_{\text{SLACK}} - \text{G O P}_{\text{BITS}} \quad (\text{式 4})$$

10

【 0 0 7 1 】

ステップ 2 4 0 で現在処理中の G O P 内に残っているビット数 (R) を計算した後、方法 2 0 0 はステップ 2 4 5 に進み、そこで (次の) 等式 5 に従って反応パラメータ (r) が計算される。方法は次にステップ 2 6 0 に進む。

【 0 0 7 2 】

【 式 5 】

$$r = 2 \times \text{B I T R A T E} / 30 \quad (\text{式 5})$$

20

【 0 0 7 3 】

ステップ 2 5 0 で現在処理中の G O P 内に残っているビット数 (R) を計算した後、方法 2 0 0 はステップ 2 5 5 に進み、そこで (次の) 等式 6 に従って反応パラメータ (r) が計算される。方法はそれからステップ 2 6 0 に進む。

【 0 0 7 4 】

【 式 6 】

$$r = 2 \times \text{B I T R A T E} / 24 \quad (\text{式 6})$$

30

【 0 0 7 5 】

上記のビット割り当て方法は、式 1 から 6 に関連して上述したように、閉じた形態で解くことができる二次レートモデルに基づいている。利点として、図 2 のビット割り当て方法は、図 2 の T M 5 の線形モデルを用いて達成されるより正確なフレームレベル目標ビット割り当てを提供する。上記の方法は、I フレーム、P フレーム及び B フレーム上にビットを適切に分布させることによって時間的なバッファ変動を制限し均一な品質を提供する。このようにして、低ビットレートにおける市販のエンコーダに共通の I フレーム、P フレーム及び B フレーム上で品質が変動することに起因する従来の「パルス化」アーチファクトが避けられる。そのうえ、テレシネ変換 (すなわち、3 : 2 プルアップ処理) を利用することによって、24 f p s ビット割り当て、情景遷移での G O P 構造の適応及びビット割り当て、符号化 1 0 4 に対するレートコントローラ 1 4 0 が出力する制御信号によって、結果得られる符号化済みビデオストリームは、平滑な情景間遷移とフレーム内ビット割り当てを提供しながらも利用可能ビット予算を固化的に利用することを保証する。

40

【 0 0 7 6 】

図 2 C に、情報不連続性 (情景変化など) の存在 / 不在に反応して所定の画像グループ (G O P) 情報構造を適応的に修正する方法 2 6 2 のフローチャートを示す。具体的には、図 2 C の方法 2 6 2 は、情景変化に反応して発生するような、緊密近似の I フレームの符号化を避け易いように事前定義済み G O P 構造を適応する。すなわち、ある情景の最終サブ G O P 内のアンカーフレームが、I フレームではなく P フレームとして符号化され、その一方では、次の情景の第 1 のアンカーフレームが I フレームとして符号化され、これに

50

よって、GOPに割り当てられたビット予算をより効果的に利用する。

【0077】

GOP方法262は、例えば図2のレート制御方法200がステップ262（すなわち、フレームが符号化される）を実行すると、ステップ262-2で入力される。方法262は次にステップ262-4に進み、ここで、情景変化制御信号SCPとSCNの状態が検査される。次に方法262はステップ262-6に進む。

【0078】

ステップ262-6で制御信号SCPとSCNがそれぞれ0と1に等しいかどうか照会される。ステップ262-6における照会が肯定的に回答された（情景変化が現在のサブGOP中で発生した）場合、方法262はステップ262-8に進む。ステップ262-6での照会に対する回答が否定的である場合、方法262はステップ262-10に進む。

【0079】

ステップ262-8で、現在のサブGOPのアンカーフレームがPフレームとして符号化される。次に方法262はステップ262-20に進み、ここから脱出する。

【0080】

ステップ262-10では、制御信号SCPが1に等しいかどうか照会される。すなわち、前に符号化されたサブGOP中で情景変化が発生したことを制御信号SCPが示しているかどうか照会される。ステップ262-10での照会に対する回答が肯定的である場合、方法262はステップ262-12に進み、ここで、現在のサブGOPのアンカーフレームがIフレームとして符号化される。次に方法262はステップ262-20に進み、ここから脱出する。ステップ262-10での照会に対する回答が否定的である場合、方法262はステップ262-16に進む。

【0081】

ステップ262-16では、現在のサブGOPのアンカーフレームはGOP構造による。符号化されるフレームがアンカーフレームでない場合、このフレームはGOP構造ベースで符号化される。次に、方法262はステップ262-20に進み、ここから脱出する。

【0082】

図2Cの方法では、情報のサブGOPを、例えばサブGOPバッファ184によってバッファリングすることが必要である。本発明の1つの実施形態では、このようなサブGOPバッファは用いられていない。この実施形態では、参照してここにその全体が組み込まれる1998年6月26日に提出された米国特許出願第09/105,730号（代理人整理番号第12411号）の教示を利用している。具体的には、サブGOPバッファが欠落している本発明の1つの実施形態では、各アンカーフレームをIフレームとPフレーム双方として符号化される。情景変化の際には、Pフレームをアンカーフレームとして用い、次のアンカーフレームをIフレームとして符号化する。入力サブGOPバッファだけが欠落している本発明の別の実施形態では、出力サブGOPバッファを用いて符号化済みサブGOPを記憶するようになっている。情景変化の際には、出力サブGOPバッファに記憶されているアンカーフレームが必要に応じて再符号化されて、ある情景の最後のサブGOP内のPフレームと新しい情景の最初のサブGOP内のIフレームを用意する。

【0083】

図3に、図1のMPEG様のエンコーダ100で使用するのに適した方法のフローチャートを示す。具体的には、図3の方法300を用いて、サブGOPを形成する複数の受信された情報フレームの符号化を制御する。方法300はステップ310で入力されステップ315に進む。

【0084】

ステップ315で、情報フレーム、例えばビデオ情報フレームが入力情報ストリームINから受信されてサブGOP記憶用バッファ184に記憶される。次に、方法300はステップ320に進む。

【0085】

ステップ320で、ステップ315で受信された情報フレームが前に受信された情報フレ

10

20

30

40

50

ームとフィールド単位で比較される。すなわち、前の情報フレーム（すなわち、受信された最後のビデオフレームに次ぐフレーム）の頂部フィールドが受信された情報フレーム（すなわち、受信された最後のビデオフレーム）の頂部フィールドと比較される。同様に、前の情報フレームの底部フィールドが受信された情報フレームの底部フィールドと比較される。この比較の結果を用いて、図1を参照して上述したように、プリプロセッサ101の3:2プルダウンプロセッサ181が発生した制御信号SCPとSCNの状態を変更する。次に、方法300はステップ325に進む。

【0086】

ステップ325では、情景変化の有無がコントローラに示される。すなわち、プリプロセッサ101の情景変化検出器182を図1を参照して上述したように利用して、情景変化指示制御信号SCPとSCNを発生する。次に、方法300はステップ330に進む。

【0087】

ステップ330では、入力簿でイストリーム中のいかなるフィールドの繰り返しもコントローラに示される。すなわち、プリプロセッサ101の3:2プルダウンプロセッサ181は、フィールドベース比較ステップ320で誘導された情報を利用して、なんらかのフィールドが繰り返されたかどうかをフレーム毎に判断する。頂部フィールドが繰り返されていたらそれは制御信号R__TOPによって示され、一方、底部フィールドの繰り返しは制御信号R__BOTによって示される。次に、方法300はステップ335に進む。

【0088】

ステップ335では、フレームのサブGOPに対する全てのフレームレベル判断が完了したかどうか照会される。すなわち、なんらかの情景変化又は3:2プルダウン情報がサブGOP全体の各フレームに対して決定されるように、サブGOPバッファ184中に記憶されている入力フレームが処理されたかどうかについて判断される。ステップ335での照会に対する回答が否定的である（すなわち、記憶されているサブGOPが完全には処理されていない）場合、方法300はステップ315に進み、ここで、サブGOPの次の入力フレームが受信されてサブGOPバッファ中に記憶される。ステップ335での照会に対する回答が肯定的である場合、方法300は次にステップ340に進む。

【0089】

ステップ340では、サブGOPバッファ184に記憶されている情報フレームのサブGOPはエンコーダ104にカップリングされて、ここで符号化される。このカップリングされたサブGOPの符号化は、この開示のどこかに記載されている方法に従って実行される。次に、方法300はステップ315に進み、ここで、次のサブGOPの最初の情報フレームが受信されてサブGOPバッファ184に記憶される。

【0090】

既述したように、図1のMP EG様の符号化システムでの使用に適したフィールド間輝度又はクロミナンス区別法を利用した3:2プルダウン検出方法が、米国特許出願第09/151,425号に詳述されている。フィールド区別計算を加速し、また、メモリー帯域幅を減少させるために、ここに開示する区別技法（ルマ繰り返しフィールド検出器）がオプションとして、次の方法に従って2つのフィールドを比較する：第1に、例えば、各マクロブロック（例えば、1マクロブロック=8×行×16画素）中の1つの行に沿った同じパリティの連続輝度フィールド内での絶対差（SAD）の和が計算される。第2に、この絶対差（SAD）の和の計算値が上限しきい値T__MB__HIGHより大きい場合、第2のフィールドは第1のフィールドの繰り返しではなく、この方法は終了する。第3に、SADの計算値が上限しきい値レベルT__MB__HIGH未満である場合、そのフィールド中の次のマクロブロックに進み、ステップ1から3を繰り返す。第4に、SADの計算値が上限しきい値レベルT__MB__HIGH以下であり下限しきい値レベルT__MB__LOW以上である場合、マクロブロック全体にわたってSADを計算する。第5に、SADがマクロブロックT__MB上でしきい値より大きい場合、第2のフィールドは第1のフィールドの繰り返しではない。繰り返しである場合、次のマクロブロックに進み、上記のステップを繰り返す。最後に、全てのマクロブロックが処理された結果、SADの合計値がフ

10

20

30

40

50

レーン T_{FR} 上でしきい値を越えない場合、第 2 のフィールドは第 1 のフィールドの繰り返しである。

【 0 0 9 1 】

このような多段しきい値付けによって、誤警告の可能性が軽減される。また、これらのしきい値によって、フィールドが正確な繰り返しとなることを妨げかねないトランスコーディング誤差を明瞭化することが可能となる。これらのしきい値は、3 : 2 プルダウン素材の大集合に基づいて実験的に到達させることができる。

【 0 0 9 2 】

受信されたサブ GOP を予備処理したら、予備処理された入力情報ストリーム IN 内の情報フレームは符号化モジュール 1 0 4 と第 2 のプリプロセッサ 1 0 2 にカップリングされる。

10

【 0 0 9 3 】

図 1 に戻ると、第 2 のプリプロセッサ 1 0 2 は予備処理された入力情報ストリーム IN ' を受信して処理し、これによってピラミッド情報ストリーム PG と、オプションとして、ブロック分類ストリーム BC を発生する。第 2 のプリプロセッサ 1 0 2 は次の機能的構成部品を含んでいる：ピラミッド発生器 1 0 2 - 2、オプションのデータパッカー 1 0 2 - 3 及びオプションのブロック分類器 1 0 2 - 4 である。

【 0 0 9 4 】

ピラミッド発生器 1 0 2 - 2 は、予備処理された入力情報ストリーム IN ' 内のフレームを空間的に分解して、フレーム、奇数フィールド及び偶数フィールドの低域通過ピラミッドを誘導する。次に、アップサンプリングされた下位レベルを減算することによって、詳細なピラミッドが作成される。詳細画素が 2 つのレベルに量子化されて、1 画素当たり 1 ビットを用いて表される。画素値を表示するために使用されるビットの数がこのようにしてかなり減少するため、例えば、符号化モジュール 1 0 4 内での動作推定プロセスにおける計算上オーバーヘッドが減少する。例えば、動作推定プロセスで実行されるブロック整合動作を加速させることができるが、その理由は、画素値が取ることができる値がほとんどなく、従って全体的なブロック整合プロセスが簡略化されるからである。

20

【 0 0 9 5 】

オプションのデータパッカー 1 0 2 - 3 を用いて、図 9 から 1 2 を参照して以下に説明するバック戦略を用いて、結果としてのピラミッドを表示するデータをバックする。オプションのデータパッカー 1 0 2 - 3 を用いて、例えば、符号化モジュール 1 0 4 内での動作推定プロセスで利用されるメモリーリソースの量を減少させる。簡単に言えば、平均データ及び / 又は M 進法データが「バック」されて定義済みデータワードとなり、これによって計算上の要件とメモリー帯域幅上の要件とを達成する。すなわち、M 進法ピラミッドを表すデータを、以降の記憶、検索及び処理などの動作が効率的に実行されるような非常に特定の仕方で配置する。例えば、1 つの 3 2 ビットワードを用いて、あるピラミッドの画素ブロック内の 1 行全体の画素を表すようにする。このようにして、動作推定に用いられるピラミッドの全体サイズが (4 から 8 桁だけ) 減少する。

30

【 0 0 9 6 】

オプションのデータパッカー 1 0 2 - 3 を用いると、バックされた表示データは図 1 の M P E G 様の符号化システム 1 0 0 の他の部分で使用しなければならない。具体的には、符号化システム 1 0 4 は、ピラミッド情報ストリーム PG を表すバック済みデータを処理することが可能でなければならない。

40

【 0 0 9 7 】

オプションのブロック分類器 1 0 2 - 4 を用いて、ブロックとマクロブロックを、例えば動作推定プロセスでより効率的にこれら両者を比較できるように分類する。例えば、ブロックを「アクティビティレベル」で分類されるが、この場合、アクティビティのレベルは非ゼロ画素の数、又は画素もしくはなんらかの他のパラメータの範囲又は時計的分布に関連する。

【 0 0 9 8 】

50

図7に、平均ピラミッドが複数のレベル710、720及び730を含む平均ピラミッド700を発生するために用いられる量子化プロセスを表すグラフである。最下位レベル710は、「x」で表される複数の画素711を有する映像シーケンスから取ったオリジナルの映像である。一般的に、これらの画素は、画素値を表すために割り当てられるビットの数によって制限される範囲を有する画素値によって表される。例えば、8個のビットが割り当てられた場合、画素値は256の考えられる値の内の1つの値をとる。

【0099】

平均ピラミッドにおいては、次に高いレベルが双方向での低域通過フィルタリングと2桁ダウンサンプリングによって発生され、これによって、より下位のレベルでの4つの画素値(子)からより高位のレベルの単一の画素値(親)を発生する。この様子を図7に示すが、ここでは、4つの画素712aから712dの各集合を用いて、レベル720で1つの画素値721を発生する。次に、4画素値722aの集合を用いて、レベル730の1つの画素値731を発生する、という具合に次々に画素値を発生する。本発明は3つのレベルを有する平均ピラミッドに制限されないことが理解されよう。レベルの数は一般的に、映像のサイズと、次に低い解像度を持つ映像を発生させるために選択されたダウンサンプリング係数と、によって制限される。したがって、平均ピラミッドのレベルの数は特定の用途に応じて選択することができる。

【0100】

平均ピラミッドにおいては、親画素値は、その4つの子画素値の平均値を取ることで誘導されるため、この平均ピラミッドという用語が付いている。しかしながら、他の尺度や計測法を用いて、他のタイプのピラミッドを発生させても良いが、例えば4つの子画素値の中央値に基づく尺度を用いてもよい。代替例として、子画素の周辺のより大きい領域を加重平均値に対して用いて一般的な低域通過ピラミッドを得てもよい。

【0101】

M進法ピラミッドにおいては、画素値は、量子化された各画素が「M」個の可能な画素値しか取り柄内容に量子化される。例えば、Mが2に等しい場合、量子化された各画素は0又は1しか取ることができない、すなわちその結果「二進法ピラミッド」となる。

【0102】

図8に、Mが3に等しい場合の三進法ピラミッドを発生するために用いられる量子化プロセスを表すグラフを示す。具体的には、8ビットの画素255(810a)を、子画素と親がその差に基づいて2ビット画素値10(820a)に量子化する。すなわち、親830aとその子810aから810dの各々との間の差を計算するが、この場合、4つの差の各々が次に、3つの可能な値10、00及び01に量子化される。このように、画素値128(810bと810c)が画素値01(820bと820c)に量子化され、画素値0(810d)が画素値01(820d)に量子化される。これらの表示レベルは、動作推定で用いられるビット毎XORベースの費用関数に適している。これらのレベルはまた、特徴検出とブロック分類に有用である。M進法ピラミッドは画素値の精度を減少させ、これによって、映像内の「特徴」を迅速に検出することが可能となる。

【0103】

特徴とは項アクティビティ又は強度の領域、例えば物体のエッジであると定義される。レベル810と830は平均ピラミッドのレベルであり、一方、レベル820はM進法ピラミッド(ここでM=3)のレベルであることに注意すべきである。これらのピラミッドは双方共が、図8に示すように追加のレベルを有するが、M進法ピラミッドは常に平均ピラミッドのレベルより1つレベルが低い。すなわち、1つのM進法ピラミッドレベル820を発生するためには2つの平均ピラミッドレベル810と830を必要とする。

【0104】

Mは任意の正の整数値でよいが、二進法ピラミッドの分解は雑音に対して敏感であることが分かっている。すなわち、量子化された画素値は可能な2つの値しか取り得ないので、雑音は誤差を導入しかねず、このような場合、画素値は0と1を互いに誤って有するものと誤解釈されかねない。このような過剰感度のため、特徴のあるなしが誤解釈されかね

10

20

30

40

50

い。したがって、M進法ピラミッドの分解はMが3以上である場合に用いるのがベストであることが分かっている。

【0105】

代替例として、M進法平均ピラミッドの分解を式という形態で表現できる。(i, j)で映像フレーム上での画素位置を表し、I(i, j)で位置(i, j)での強度を表すものとする。さらに、lがピラミッド内のレベルを表し、 $0 = l = L$ とするが、ここでLはピラミッド中での最高位レベルであるとする。すると、平均ピラミッド $X^l(i, j)$ 、 $1 = l = L$ が次のように構成される：

【0106】

【式7】

$$X^l(i, j) = 1/4 \sum_{m=0}^{l-1} \sum_{n=0}^{l-1} X^{l-1}(2i+m, 2j+n) \quad (\text{式7})$$

【0107】

ここで、 $X^0(i, j) = I(i, j)$ である。

【0108】

これらの平均ピラミッドから、図6を参照して以下に既述するようにブロック内の特徴を抽出することができる。この好ましい実施形態では、ブロックはマクロブロックの 8×8 サブブロックであるが、本発明はこのブロックサイズに限られるわけではないことを理解すべきである。特に、エッジなどの特徴はブロック内の強度の変動から抽出することができる。この変動は、レベルl、 $0 = l = L - 1$ での平均値とレベル $l + 1$ での平均値間の差を計算することによって表される。しかしながら、ロバストな特徴を得るため、また、高速動作推定を容易化するために、この差をMこのレベルを用いて量子化して、量子化された値を $\log_2 M$ ビットを用いて表示する。これによって、エッジやゼロ交差などの映像特徴を識別するために用いられる映像に関するパターンを作成する。このパターン値を $Y^l(i, j)$ で表すと次式のようになる：

【0109】

【式8】

$$Y^l(i, j) = \text{Quant} [X^l(i, j) - X^{l+1}(\text{INT}(i/2), \text{INT}(j/2))] , 0 \leq l \leq L-1$$

(式8)

【0110】

Quant[?]の引数を で表す。例えば、しきい値Tを持つ三進法ピラミッドの場合を考えると、 $Y^l(i, j)$ は次のようになる：

【0111】

【式9】

$$Y^l(i, j) = \begin{cases} 00 & |\lambda| < T \\ 01 & \lambda > T \\ 10 & \lambda < -T \end{cases} \quad (\text{式9})$$

【0112】

この定義は、量子化しきい値T（例えば、この実施形態ではTは5に選択される）を特定の応用分野に対して適切に選べば雑音ロバスト性という利点がある。すなわち、雑音による画素値のささいな変動を効果的に除去できる「デッドゾーン」、例えば $|\lambda| < T$ を定義することができる。したがって、ゼロ近辺のデッドゾーンを有するいかなるM進法ピラミッド($M > 2$)も二進法ピラミッドで見られるような雑音感度問題を最小化する。

【0113】

比較的平坦な領域（低アクティビティの領域）では、 $Y^l(i, j)$ は多くのゼロ(0)

10

20

30

40

50

を含み、一方エッジを含む領域では、 $Y^1(i, j)$ は複数個の 1 を含んでいる。入力映像が M 進法ピラミッドに分解されると、この入力映像中のブロックは M 進法ピラミッド $Y^1(i, j)$ を用いて特徴を抽出するという目的のために分類することができる。すなわち、M 進法ピラミッドを用いて、高い計算オーバーヘッドを招くことなく入力映像中の特徴を迅速に検出できる。この検出された特徴を用いて、以下に説明するような動作推定プロセス又は他の映像処理ステップ、例えばセグメント化モジュール 151 を用いての、例えば映像内の領域（物体など）のセグメント化を強化することができる。セグメント化は重要な映像処理ステップであり、この場合、映像中の重要な領域を識別して特殊な処理を受けるようにすることができる。例えば、ビデオ会議に応用した場合に人の顔は、コーディングビットをより多く割り当てるなどの特殊な映像上の処理を必要とすることがある。加えて、セグメント化を用いて大型の物体を識別できるが、この場合、グローバルな動作推定がこれらの大型の物体に対して実行できる。

10

【0114】

先行する説明では零として三進法ピラミッドを用いて、量子化しきい値又はレベルを特徴の識別と分類のために割り当てられる 1 つの可能な方法を示していることを理解すべきである。一般に、 $M > 2$ である M 進法ピラミッドを、映像シーケンスの特殊な応用及び / 又はコンテンツの要件に依存する量子化しきい値を特定の割り当てて用いることができる。

【0115】

図 6 に、ブロックベースの動作推定のために動作ベクトルを決定する際の計算上の複雑性を減少させるための方法のフローチャートを示す。すなわち、方法 600 は、整合が発生しそうな初期探索領域を迅速に定義することによってブロックベースの動作推定方法を強化するものである。

20

【0116】

具体的には、方法 600 はステップ 605 で始まりステップ 610 に進み、ここで、M 進法ピラミッド（又は M 進法平均ピラミッド）を、予備処理された入力情報ストリーム IN' 内の映像シーケンス中の各映像フレームに対して発生する。ステップ 610 での重要な側面は映像シーケンス中の入力映像の各々に対して M 進法ピラミッドが発生されることであることを理解すべきである。したがって、この好ましい実施形態が M 進法平均ピラミッドを発生するとはいえ、他のタイプの M 進法ピラミッド、例えば M 進法平均ピラミッド、M 進法低域通過ピラミッドなどを本発明で用いてもよい。次に、方法 600 はオプションステップ 615 に進む。

30

【0117】

オプションステップ 615 では、ステップ 610 で発生された M 進法ピラミッドは、後で例えば符号化モジュール 104 の動作推定プロセスで使用するためにパッキングされて記憶される。この M 進法ピラミッドのデータのバック動作と記憶動作は図 9 から 12 を参照して以下に詳述する。加えて、適当な方法が、その全体を参照してここに組み込む 1998 年 11 月 19 日発行の米国特許出願第 09 / 196,072 号（代理人整理番号第 12626A）に開示されている。次に、方法 600 はオプションステップ 620 に進む。

【0118】

オプションステップ 620 では、フレーム中のブロックが、M 進法ピラミッドを見て低アクティビティであるか高アクティビティであるかという点で分類される。この好ましい実施形態では、「分類ブロックサイズ」は、128 ビットで表される 64 個の M 進法画素値を有する 8×8 ブロックである。25 以上の画素値が非ゼロであれば 8×8 ブロックが高アクティビティブロックと分類される場合、「アクティビティしきい値」は 25 であると設定される。そうでない場合、 8×8 ブロックは低アクティビティブロックと分類される。追加のより高いブロックの分類、例えば、マクロブロックを高アクティビティ又は低アクティビティをマクロブロックとして分類することが可能である。この好ましい実施形態では、高アクティビティとして分類される少なくとも 1 つのサブブロックを含むようなマクロブロックは、高アクティビティとしても分類されることになる。「分類ブロックサイ

40

50

ズ」と「アクティビティしきい値」を特定の応用分野に対して調整することが可能であり、したがって、この好ましい実施形態で選択されたこれらの値に制限されることはないことを理解すべきである。次に、方法 600 はオプションステップ 630 に進む。

【0119】

ステップ 630 では、ブロック分類を用いて、符号化モジュール 104 の動作推定プロセスを強化する。一般に、重要な映像特徴を持った領域での動作推定は、アパーチャ問題によるほとんど無変化の比較的「平坦な領域」（例えば、隣り合ったブロック動詞の映像のコンテンツが非常に類似している一様な領域）での動作推定より信頼性が高い。したがって、上記の分類方法を用いて、一般的な動作推定の信頼性を高める。すなわち、本発明を用いて、様々なタイプ又は異なったアーキテクチャの動作推定方法の性能を高めることができることを理解すべきである。

10

【0120】

具体的には、動作推定は一般的に、ラスタースキャン順序でブロック毎に実行される。計算上のオーバーヘッドすなわち費用は一般的に、動作推定プロセス中で全てのブロックにわたって均一に分配される。本発明では、エッジブロック（高アクティビティブロック）における動作推定は最初に、 $Y^1(i, j)$ 及び / 又は $X^1(i, j)$ に依存する費用関数を用いて実行することができる。この方式によって映像中の特徴を強調することができ、センサー雑音、量子化雑音及び証明変化の存在下でもロバストで信頼性の高い動作推定値が提供される。費用関数のある例では、ある種のアーキテクチャに対する高速方法として実現することが可能な、ピラミッド中の M 信号レベルに対するビット毎の XOR 演算を必要とすることがある。この費用関数を用いて、「最良の正号」を決定することができる。時点 t （現行フレーム）における M 進法値ブロック $Y^1(i, j)$ と時点 $t-1$ （前のフレーム）における別の M 進法値ブロック $Y^1(m, n, t-1)$ を考える。すると、費用関数は次のようになる：

20

【0121】

【式 10】

$$\sum \{Y^1(i,j,t)(X) Y^1(m,n,t-1)\} \text{ における 1 の数} \quad (\text{式 10})$$

【0122】

ここで、シグマはブロック内の画素を表し、 (X) はビット毎の XOR 演算を表す。

30

【0123】

この費用関数は、オリジナルの 8 ビット画素強度値に対して用いられる標準の「絶対差」費用関数と比較してかなりの計算の節約をもたらす。この手順を M 進法ピラミッド上で階層的に実行する。言い換えれば、この動作推定方法は高アクティビティブロックから開始される。

【0124】

複数のブロックに分割され、その内の 2 つが高アクティビティブロックと分類された入力映像フレームの場合を考える。したがって、動作推定を最初にこの 2 つのブロックに対して実行する。事実、計算費用はこの 2 つのブロックの場合は高くなるかもしれないが、その理由は、これらの高アクティビティブロック（高信頼性エッジブロック）は非常に高い精度の動作ベクトルを提供しやすいからである。したがって、より集約的な動作推定を入力映像フレーム中の他のブロックよりこの 2 つのブロックに対して実行する。例えば、この 2 つの高アクティビティブロックを分割して、より正確な動作ベクトルを得ることができ、「半画素」動作推定をこの 2 つのブロック中で実行することが可能であったりより細密な探索戦略を用いてもよい。

40

【0125】

次に、この高アクティビティブロックに対する動作推定が完了したら、次に動作推定はその映像中の低アクティビティブロック（「低信頼性」ブロック）に伝搬する。しかしながら、この伝搬は、分類から得られる領域又は物体のセグメント化に依存してインテリジェ

50

ントに実行される。この伝搬は、隣同士のブロックの動作のための初期化としてのエッジブロックの動作と、この初期化を洗練させるための比較的小さい探索範囲を用いて実行される。すなわち、動作推定プロセスは、高アクティビティブロックに近接したブロックに（例えば、螺旋順序で）伝搬する。次に、この伝搬戦略は、エッジブロックに隣接した又は近接した平坦ブロックにまで拡大される。

【0126】

このようにして、動作推定探索範囲が迅速にしかも比較的低い計算上の複雑さで洗練される。さらに、結果得られる動作推定値はより平滑でより符号化しやすく、これによって、動作情報がビットストリームのかかなりの部分を占める非常に低いビットレート（V L B R）応用分野では大きな利点となっている。さらに、これらのより平滑な動作推定値は、一時的内挿をする応用分野ではより性能がよいものと期待される。最後に、この分類方法は又、半画素を洗練させて動作推定値の精度を増すと計算回数の節約をもたらすが、その理由は、半画素の洗練動作はエッジブロックに対してだけ実行され、映像の比較的平坦な領域には実行されないからである。

10

【0127】

図1のMPEG様エンコーダのメモリーリソース要件をさらに減少させるために、本発明のある実施形態では、主要な4つの要素を提供しているがそれは：1）平均ピラミッド及び／又はM進法ピラミッドの作成；2）作成された平均ピラミッドとM進法ピラミッドを表すデータのパッキングとメモリー中への記憶；3）このパッキングされたデータをレジスタに移動させてアーキテクチャ計算に用いる；4）このデータを用いて整合誤差計算に用いる。これら4つの主要な用度を以下に説明する。

20

【0128】

この4つの主要な要素の内の最初の要素は、平均ピラミッド及び／又はM進法ピラミッドの作成動作を含んでいる。この開示の前の部分では平均ピラミッド及び／又はM進法ピラミッドの作成に適した方法と装置を述べているので、このような作成はこれ以上詳細には説明しない。しかしながら、本発明による計算上の複雑性の減少を理解するためには、平均ピラミッド又はM進法ピラミッドを作成するために必要とされる計算上の負荷を理解することが重要である。図示の例をここで提示する。

【0129】

ビデオストリームを $J + 1$ 子のレベルに分解し、レベル0が最も細密であり J が最も粗であると仮定する。MPEG-2の主要プロフィール@主要レベルビデオストリームの場合、 J の一般的な値は3である。Pをレベル0での画素の数とすると、レベル J における画素の数は $P / 4^J$ となる。

30

【0130】

平均ピラミッドの場合、 $J = 0$ 以外のいかなる特定のレベル J においても、特定のレベルに対する値を発生するために必要な計算は、画素1つ当たり3つの加算と1つのシフトを含む。 $J = 0$ の場合、平均ピラミッドはオリジナルの映像自身であり、計算は不要ない。したがって、それを発生するための演算の総数は $4 / 3 * P$ であり、 J が大きい場合の限界内である。

【0131】

M進法ピラミッドの場合、 $J = 0$ 以外のいかなる特定のレベル J においても、特定のレベルに対する値を発生するために必要な計算は、画素1つ当たり3つの加算と1つのシフトを含む。すなわち、M進法ピラミッドは0から $J - 1$ までの全てのレベルでの計算が必要である。さらに、各画素に対して、 $M - 1$ 回の比較動作を実行する必要がある。したがって、計算の回数は J が大きい場合の限界内である $4 / 3 * P * (M - 1)$ である。

40

【0132】

上記の4つの主要要素の内の第2の要素は、作成された平均ピラミッドとM進法ピラミッドを表すデータをパッキングしてメモリー中に記憶する動作を含む。本発明の少なくとも1つの実施形態が利用する動作推定スキームは、例えばM進法データを定義済みのデータワード中に「パッキング」して計算上の要件とメモリー帯域幅要件を達成することに依存

50

している。すなわち、M進法ピラミッドを表すデータを、後続の記憶、検索及び処理などの動作が効率的に実行されるように非常に特定のな方法で配置する。M進法データのこの「パッキング」は、一般のマイクロプロセッサ又はコンピュータアーキテクチャで用いられるような本発明による方法を高速でソフトウェア的に実現するためには重要である。

【0133】

的確にパッキングできるかどうかは動作推定のためのブロックサイズ、すなわち各画素での状態の数Mと特定のアーキテクチャの場合のレジスタサイズとによって異なる。例えば、一般的な画素ブロックサイズ(すなわち、 $N_1 \times N_2$ ブロック、ここで N_1 は行の数であり、 N_2 は列の数を示す)は 16×16 、 16×8 又は 8×8 個の画素ブロック(すなわち、 $N_2 = 16$ 又は $N_2 = 8$)である。M個のレベルを表すために用いられるビットの数はNであり、Nは歪みを計算する正にその方法によって異なる。

10

【0134】

歪みは2つの画素ブロック同士間の非類似性の尺度である。例えば、符号化される画素ブロックと基準の画素ブロック間の歪みは、この2つのブロック間で論理的排他OR(XOR)演算を実行することによって計算される。歪みはこのXOR演算から結果とし得られる"1"の数に等しい。1がない(すなわち、歪みがゼロに等しい)場合、画素ブロックは正確に整合していることになる。多くの1、例えばしきい値レベルを越える数の1がある場合、画素ブロックは整合しない。1の数がしきい値未満であれば、画素ブロックは正確に整合していないまでも非常に類似している。歪みをこのように計算すると、Nは $M - 1$ にもなる。M = 3の場合、N = 2で一般的に十分である。

20

【0135】

図9に本発明を理解する際に有用な画素情報を表すパッキングされた又はブロッキングされたデータを示す。具体的には、図9は複数の画素ブロックを示しており、その各々の画素ブロックが複数のそれぞれのワード(図では4つのワードが示されている)によって表され、各々のワードが複数の画素(図では4つの画素が示されている)を表す情報を含んでいる。これらのブロックは垂直方向デマケーションVB1及びVB2並びに水平方向デマケーションHB1及びHB2によって示される。各「x」はM個の状態を有する1つの画素を表し、画素の各ボックス化されたグループが1つのデータワードを表す。

【0136】

図9に、レジスタサイズ $N_1 \times N_2$ がワードサイズに等しい場合を示す。すなわち、32ビットというレジスタサイズによって、三進法ピラミッド(M = 3、N = 2)の 16×16 画素ブロックのライン全体を示すことができる。

30

【0137】

しかしながら、各画素の状態の数(M)、行の数(N_1)、列の数(N_2)及び処理されている特定のピラミッドレベル(J)によっては、レジスタサイズ $N_1 \times N_2$ はワードサイズより大きくなったり小さくなったりする。レジスタサイズ $N_1 \times N_2$ がワードサイズより大きい場合、ライン中のデータを表すのに2ワード以上を使用する必要がある。レジスタサイズ $N_1 \times N_2$ がワードサイズ未満であれば、図10を参照して以下の述べるようにデータは冗長記憶内容とオーバーラップさせて記憶させてもよい。

【0138】

図10に、本発明を理解する際に有用な画素情報を表すパッキングされた又はブロッキングされたデータを図示する。具体的には、図10は複数の画素ブロックを示しているが、その各々の画素ブロックが複数のそれぞれのワード(図では4ワード)によって表され、各々のワードが複数の画素(図では4画素)を表す情報を含んでいる。これらのブロックは、垂直方向デマケーションVB1からVB4と水平方向デマケーションHB1とHB2によって示される。各「x」はMこの状態を有する1つの画素を表し、画素の各ボックス化されたグループが1つのデータワードを表す。

40

【0139】

図10は、レジスタサイズ $N_1 \times N_2$ がワードサイズ未満であり、このため、データが冗長記憶内容とオーバーラップするように記憶される場合を示している。この2つの4ビット

50

画素ブロックは垂直方向デマケーションラインの両側からの画素を含んでいることに注意されたい。したがって、各ワードはそのブロックからのデータと、水平方向に互いに隣接する2つのブロックからのデータとを、を含んでいる。より細かいレベルで小さい探索範囲を持つ階層的な動作推定スキームでは、この方法は負荷の数を非常に小さい数値に保ち、これによって、メモリーの帯域幅を減少させている。オーバーラップがない場合、基準フレーム中のブロックが水平方向ブロック境界に対応しない場合には、隣接する水平方向ブロックに対応する追加のワードをロードする必要がある。

【0140】

パッキングプロセスでは、各画素に対して2つのALU演算並びにシフト演算及び論理OR演算が必要である。したがって、非オーバーラップパッキングの場合（図9を参照）、画素毎の演算数は $8/3P$ である。オーバーラップパッキングの場合（図10を参照）、さらなるシフトとOR演算が必要である。

10

【0141】

4つの主要な要素の内の第3の要素は、パックされたデータをレジスタ中に移動させて算術計算できるようにする動作を含む。具体的には、この主要な4つの要素の内の第3の要素はメモリー帯域幅問題、すなわち、レジスタに対するデータの出し入れの速度とこのような移動の必要は発生回数を含意している。

【0142】

図5に、本発明の理解に有用な計算デバイスのある部分、特にメモリー帯域幅問題を示す。具体的には、図5は汎用、メモリーモジュール510、キャッシュメモリーモジュール520、データバス530、汎用レジスタ540及び算術/論理演算ユニット（ALU）550を備えるマイクロプロセッサ又はコンピュータアーキテクチャのある部分を示している。汎用レジスタファイルは、複数のレジスタグループを含むものと考えられる。

20

【0143】

例えばM進法ピラミッドを表すパッキングされたデータはメモリーモジュール510に記憶される。このデータの処理、例えば動作推定プロセスを実行するには、最初に、データを汎用レジスタファイル540に移動させる必要がある。これは、所望のデータワードをキャッシュメモリーモジュール520から又は、必要に応じて、メモリーモジュール510から直接に検索することによって達成される。このデータはメモリーモジュール510から信号経路S1を介してキャッシュメモリーモジュール520に、またキャッシュメモリーモジュール520からデータバス530を介して汎用レジスタファイル540にカップリングされる。次に、このデータは、反応レジスタファイル540中の複数のレジスタファイル540又はレジスタグループの内の1つに記憶される。

30

【0144】

このデータに対して算術演算又は論理演算を実行するために、このデータは第1の信号経路S2（又はS3）を介してALU550にカップリングされる。ALU550によって実行される算術演算又は論理演算が第2のオペランドを必要とする場合、この第2のオペランドが第2の信号経路S3（又はS2）を介してALU550にカップリングされる。すると、ALU550の出力は汎用レジスタファイル540内のレジスタに記憶される。次に、この記憶された出力はメモリーキャッシュ520を介してメモリー510に記憶され、これによってデータバス530と信号経路S1を利用する。

40

【0145】

画素を表すデータワードが汎用レジスタファイル540中のレジスタ又はALU550の算術/論理処理エレメントのサイズより大きい場合、データワードはインクレメント的に処理される。すなわち、データワードは、データワード全体が処理されるまで、適切なサイズを持つ一連のサブワードとして処理される。

したがって、データワードへの画素情報の表示を、汎用レジスタファイル540又はALU550の算術又は論理処理エレメントに制約することによって、データワードは一連のサブ演算によってではなく1回の演算によって処理される。

【0146】

50

主要な4つの要素の内の第4の要素に関連して説明される誤差計算を実行するためには、後で処理するためにデータを表す画素を様々なレジスタに入れる必要がある。簡単に言えば、誤差計算を用いて、画その強度レベルの整合を発見して、ある時間にわたる画像内（例えば、連続する映像フレーム内）の画素ブロックの動作を識別する支援とする。したがって、図5に示す計算装置を、例えば図1のMPEG様符号化システム100の動作推定モジュール150部として用いてもよい。

【0147】

オリジナルの映像の画素ブロック中の各画素の強度レベルを8ビットで表し、レジスタサイズが32ビットに等しいとすると、画素ブロックのほんの小部分を1時にレジスタ中にロードできるに過ぎない。したがって、画素の強度レベルをオリジナル映像（すなわち、 $J = 0$ ）に整合させる際には、多くの検索や処理や記憶の動作を実行して、2つの画素を比較しなければならない。しかしながら、M進法では、画素ブロックと関連する強度レベルデータのほとんど（又はすべての）が一時にロードされ、これによってメモリー帯域幅要件をかなり節約する。多くの場合、現行の（オリジナルの）映像は動作推定の開始時にレジスタ中にロードでき、また、その特定のブロックの探索時全体にわたってそこに常に留まる。

【0148】

本発明を表すパックされたデータを用いた整合する画素ブロックの探索を次に説明する。この探索は、現行画素ブロック内の画素の画素強度レベルを例えば先行するフレームの選択された探索領域中の画素から成る類似サイズのブロックと比較するステップを含む。したがって、処理されている基準フレームの現行画素ブロックを例えばレジスタファイル540中にロードする必要がある。ここに説明するパッキング方法を利用して、ライン内の多くの画素に対応するM進法値が1ワード中に記憶される。例えば、 $\{M = 3, N = 2, N^2 = 16\}$ の場合、16の32ビットワードを例えばレジスタファイル540のレジスタにロードする必要がある。

【0149】

説明の目的上、探索範囲は $-K_1$ から K_1 という垂直方向探索範囲と $+K_2$ から K_2 という水平方向探索範囲から成り、ここで、 K_1 は垂直方向を示し、 $2K_1 + 1$ は行の数を示し、 K_2 は水平方向を示し、 $2K_2 + 1$ は列の数を示すものと仮定する。

【0150】

最初のステップはゼロ探索ベクトルデータをレジスタ中にロードすることである。ゼロ探索ベクトルデータは、（例えば、粗ピラミッドレベルからの）前の推定値に基づいたゼロ洗練値を含むことがある。ゼロ探索ベクトルはブロック境界と整合する場合、すなわち、オーバーラップした記憶内容が境界との不整合を覆うに十分である場合、そのブロックに対応するワードをロードする必要があるだけである。そうでない場合、2集号以上のワードをロードして、このロードされたワード集合に対してシフト、マスク及び論理OR演算を実行して、データをレジスタ内で適切に整合させる必要がある。

【0151】

現行ブロックと現時点で識別された探索ブロック間の誤差が次に計算される。整合することがわかったら、現行ブロックと関連する動作ベクトルが決定される。基準フレーム中の次の画素ブロックが現行ブロックとして選択され、「新しい」現行画素ブロックに対する探索プロセスが繰り返される。

【0152】

整合しないことが認められた場合、水平方向の探索に移行する。すなわち、探索ウィンドウが、例えば1画素分だけ水平方向にシフトする。さらに、負荷の数が、水平方向基準ブロックがブロック境界に沿って存在するかどうかによって異なる。次に、現行ブロックと現時点で識別されている探索ブロック間の誤差を計算して整合するかどうかを識別するステップが繰り返される。これらのステップは、水平方向探索範囲が検査されるまで繰り返される。

【0153】

水平方向探索範囲が現行ブロックに対する整合を発見することなく検査されたら、探索ウィンドウは垂直方向に1ラインだけシフトして、水平方向探索範囲全体（-K2からK2）が（整合が発見されるまで）再度検査される。このプロセスは、探索範囲中の全ての垂直方向位置（2K1+1位置）で繰り返される。これは重複がない場合である。

【0154】

主要な4つの要素の内の第4の要素は、整合誤差の計算用のデータの使用を含んでいる。例えば、現行ブロック（ci）が前の又は基準のブロック（pi）と整合するかどうか（すなわち、同じか又は類似しているか）を判断するために、2つのブロックを排他OR（XOR）演算して結果を出す。その結果中の1の数がこの2つのワード間の不整合の数である。不整合の数がゼロ（すなわち、結果がゼロ）又はしきい値未満であれば、現行ブロック（ci）は前の又は基準のブロック（pi）と整合していると言われる。探索を実行するときに、この誤差計算は、現行ブロック（ci）と整合する前の又は基準のブロック（pi）が見つかるまで繰り返し実行される。

10

【0155】

上記のパッキング・記憶方法を利用することによって、誤差整合計算を実行するために必要とされる処理とメモリー帯域幅は大幅に減少する。そのうえ、データがパッキングされてオーバーラップするように記憶されると、検査中の前の又は基準のブロック（pi）に起因しないエラー誤差計算結果中のいかなる不整合も無視される。

【0156】

図11に、本発明の理解に役立つ画素情報を表すパッキングされた又はブロック化されたデータを図示する。具体的には、図11は複数の画素ブロックを示し、各画素ブロックはそれぞれが複数のそれぞれのワード（図では4ワード）によって表され、各ワードは、複数の画素（図では4画素）を表す情報を含む。これらのブロックは垂直方向でマージョンVB1からVB4及び水平方向デマージョンHB1とHB2によって示される。各「x」が、M個の状態を有する1つの画素を表し、各ボックス化された画素のグループが1つのデータワードを表している。

20

【0157】

図11に、画素ブロックの1つのラインを提供する、非鎖線で示された、2つのワードからの情報を含む鎖線で示された現行ブロックciを示す。したがって、各ラインに対する2ワードの各々の全体を用いる誤差計算は上記のように実行されるが、現行ブロックciの外部のデータに起因する結果としてのワードの部分は単に、整合があったかどうかの判断の際には無視される。これは、各ブロックライン中の全てのワードに対して実行し、その結果を累積する必要がある。

30

【0158】

本発明の1実施形態では、XOR結果の内の1つが、例えばサイズ256のテーブルに関してテーブルルックアップによってカウントされる。この実施形態では、XOR結果を保持しているレジスタ内の適切なビットがバイトに分割され、テーブルルックアップが実行されて、累積されたXOR結果が加算されて最終的な誤差数字となる。

【0159】

既述したように、M進法ピラミッドは平均ピラミッドから構築される。したがって、平均ピラミッドは現行のM進法ピラミッドレベルにとって利用可能である（すなわち、平均ピラミッドはM進法ピラミッドの構築中に記憶されて、必要に応じて検索され得る）。また、平均値はブロック境界内に存在する各ブロックに対して利用可能であると仮定されるが、その理由は、この情報が平均ピラミッド中の最も粗であるレベルから得ることができるからである。

40

【0160】

平均ピラミッド情報を整合判断基準の1部として用いる場合、各シフト演算にとって、シフトされた基準ブロックに対する平均を計算する必要がある。水平方向シフトの場合、これは2N1個の負荷と、N1回の減算と、N1回の加算とに等しく、さらに水平方向では、2N2個の負荷と、N2回の加算とN2回の減算を必要とする。また、平均整合は一般

50

的には、平均の誤差をXORの誤差を合成するには1回の減算と1つの加算を必要とする。

【0161】

三進法ピラミッド例との平均整合演算を実行するには、16のワードをレジスタ中のロードして平均値から減算し、16の新しいワードを次にレジスタにロードして平均値に加算する。平均ピラミッドがオーバーラップパッキングされて記憶された(図10を参照して上述した)場合、乗算又は除算(シフト演算を用いて実行されることがある)を実行するために追加の演算が必要となり得ることに注記すべきである。したがって、平均計算を1回実行するには32回の負荷演算と34回の算術又は論理演算が必要である。1つの平均値整合に必要な計算の総数は36プラス、あらゆる乗算と除算(図では4つ)の回数である。

10

【0162】

全探索ブロック整合を直接的に適応することは極めて非効率的であるので、本発明の方法を高速化する階層的スキームを考慮するのが有用である。一般的な階層的なスキームにおいては、初期レベルピラミッドは映像から構築される。より粗である各ピラミッドレベルはフィルタリングされ、(粗雑であるという点で)先行する映像ピラミッドのサブサンプリング版となる。動作は最初にピラミッドのより粗であるレベル(範囲が最も細密でオリジナルの解像度における範囲よりはるかに低いレベル)で推定され、次に、小さいウィンドウ上での探索が洗練されるより細密なレベルに伝搬する。範囲はより細密な解像度(より多くの画素があり、したがって、整合誤差計算はより多くの計算を必要とする)では全

20

【0163】

強度ベースの整合の替わりにM進法整合のレベルの1部(又はすべて)を用いることによって、特定のレベルにおける計算上の負荷が2桁だけ減少し、メモリー帯域幅要件が6から7桁だけ減少する。これによって、M進法ピラミッド計算にとって必要とされる余分の計算が相殺され、計算回数とメモリー帯域幅がかなり減少する。また、M進法ピラミッドも、映像予備処理及びブロック分類(例えば、上記のようなもの)などの他の目的にも有用であり、また、余分の計算リソースを、例えば本発明を用いるビデオエンコーダ内の別のモジュール中で利用してもよいことに注意されたい。

30

【0164】

本発明を、M進法ピラミッドから得た特徴ベクトルに基づいた動作推定スキームの文脈内で説明した。この動作推定スキームは、類似の階層的動作推定スキームより高速であり、また、メモリー帯域幅要件が少ない。本発明は特に、本発明の計算上の利点とメモリー帯域幅上の利点をよりよく図示するために例示の三進法ピラミッド(M=3)応用例の文脈内で説明した。他のM値、他のブロックサイズ及び類似物を用いても類似の利点が認識されよう。当業者及び本発明を知った者は、添付クレームの範囲内である他の順列に本発明の教示を容易に応用できるであろう。

【0165】

バックデータ構造又はワードは、少なくとも2つの画素に関連する情報を含んでいる。バックデータ構造は、連続する論理OR演算とシフト演算を用いて、データセグメント(例えば、バイト)を表す2つ以上の画素をより大きいデータワード(例えば、32ビットワード)中に挿入して、バックデータ構造を形成することによって形成される。次に、バックデータ構造は、後続の処理を容易化する様な仕方では記憶される。例えば(上述したように)、4つの8ビット画素を1つの32ビットデータ構造中に記憶させ、8個の32ビットデータ構造を用いて1つの画素ブロックを記憶する。このように、後続のブロックレベル処理を、バックデータ構造を利用して効率的に実行する。

40

【0166】

ブロックの1行当たりのワードの数(WPR)は次式で計算される：

【0167】

50

【式 1 1】

$$WPR = \frac{N(BLK_WIDTH + N_OVERLAP_L + N_OVERLAP_R) + N_UNUSED_BITS}{WORDSIZE}$$

【0 1 6 8】

ここで、 M_j はレベル j の M 進法ピラミッドであり、 $WIDTH$ は M 進法ピラミッドの幅であり、 $HEIGHT$ は M 進法ピラミッドの高さであり、 BLK_WIDTH は M 信号ピラミッド内の画素ブロックの幅であり、 N は 1 画素当たりのビット数であり、 $N_OVERLAP_L$ は、パッキングしている際のブロックの左側でオーバーラップする画その数であり、 $WORDSIZE$ はブロックがパッキングされるデータタイプのサイズ（ビット数）であり、 N_UNUSED_BITS は、 $WORDSIZE$ のサイズを有するデータタイプを表すパッキングされたデータ中の不使用ビットの数である。

10

【0 1 6 9】

$WORDSIZE$ は様々なデータをパッキングする前に選択されたり固定されたりすることに注意すべきである。加えて、パッキングされたデータは、パッキングされるワード中のあらゆる可能なビット位置を占めるわけではないことが決定される（例えば、3 つの 8 ビット画素表示物を 3 2 ビットワード中にパッキングする）。したがって、 WPR はワード中の不使用ビット数によって修正される。

20

【0 1 7 0】

形成された 1 つの重要なパックデータ構造はパッキングされたフレームを含むが、この場合、パッキングされたフレーム内の各ワードは複数行の内の 1 つ、すなわち画素ブロック内の M 進法画素の行全体又は部分的な行を含む。パッキングされたフレーム内の各ワードもまた、オーバーラップパッキングのために水平方向に隣り合った画素ブロックの対応する行からの追加の画素を含んでいる。このようなパッキングされたフレームはオプションのデータパッカー 1 3 2 によって発生されてアンカーフレームメモリー 1 7 0 に記憶される。

【0 1 7 1】

本発明の冗長記憶実施形態では、各データワードは、これまた別のデータワードに含まれる画素情報を含むことに注意することが重要である。すなわち、第 1 のデータワードは第 1 の複数の水平方向は位置画素用の画素情報を含み、第 2 のデータワードは、第 2 の複数の水平方向は位置が素用の画素情報を含み、第 1 と第 2 の複数の水平方向は位置が素は共通の（すなわち、共用の）が素情報を含んでいる。

30

【0 1 7 2】

同様に、本発明のオーバーラップ記憶実施形態では、各データワードは少なくとも 2 つの画素ブロックから成る隣り合った行からの画素情報を含む。すなわち、データワードは、第 1 の画素ブロックの行ともう 1 つの隣接した画素ブロックの隣接した行からの画素情報を含んでいる。

【0 1 7 3】

利点として、図 1 2 の方法 1 2 0 0 の場合のように、対応する行からの追加の画素は、画素ブロック内の M 進法画素の内の複数行、全体行又は部分的行と共にレジスタ中にロードされる。このように、画素ブロック内の M 進法画素の内のロードされた複数行、全体行又は部分行は、単なる XOR 演算とロード演算ではなく XOR 演算とシフト演算を用いて迅速に処理されて、水平方向探索を実行する。

40

【0 1 7 4】

図 1 2 に、本発明による画素ブロック整合のための探索方法のフローチャートを示す。具体的には、図 1 2 は、例えば、各々の予測モード符号化済みブロック、動作ベクトル及び残余又は誤差情報を準備するブロックベースのエンコーダでの使用に適す探索方法を示す。このようなエンコーダの効率にとっては、予測されている画素ブロックと同一な少なくとも類似の画素を示す動作ベクトルを提供し、これによって、残余又は誤差情報が可能な

50

限り減少するようにすることが重要である。したがって、図 12 の探索方法は、映像又は映像ピラミッド中の各画素ブロックに対して、基準の映像又は映像ピラミッド中の画素ブロックの整合（又は類似性）を識別するために用いると利点がある。

【0175】

方法 1200 は、現行の画素ブロック（例えば、予測される画素ブロック）を、例えば図 12 を参照して上述したように処理され記憶された記憶済みの映像又は映像ピラミッド内の画素情報と比較する。

【0176】

図 12 の方法 1200 は、パッキングされたドメイン基準映像又はパッキングされたドメイン基準映像ピラミッドを使用して用いられれば利点がある。例えば、低域通過フィルタリングされサブサンプリングされた現行の画素ブロックをほぼ同じ粗雑さを持つ映像ピラミッド（例えば、現行の画素ブロックと $J = 3$ という粗雑さを持つ基準映像ピラミッド）を比較することによって、各現行画素ブロックの整合（又は整合の欠如）を発見するのに必要な処理時間を軽減できる。

【0177】

図 12 の方法はステップ 1202 で入力されてステップ 1204 に進み、ここで、現行画素ブロックは例えば、予測的に符号化される映像フレーム（又は関連の映像ピラミッド）の第 1 の画素ブロックに等しいように設定される。現行画素ブロックは基準の映像又は映像ピラミッドを形成するパッキング済みドメイン画素ブロックと比較されるので、第 1 画素ブロックは、基準映像又は映像ピラミッド様に使用される特定のパッキング済みドメイン表示物に従って（必要に応じて）フォーマットされて、現行の画素ブロックとして単数（又は複数の）レジスタに記憶される。方法 1200 は次にステップ 1206 に進む。

【0178】

ステップ 1206 では、ゼロ探索ベクトルが 1 つ以上のレジスタにロードされる。すなわち、特定の探索範囲にとって十分な左右のオーバーラップするデータを含むパッキングされたデータドメイン基準ワード（又は基準ブロック）が 1 つ以上のレジスタにロードされる。オーバーラップするパッキング済みデータを基準ワード（又はブロック）のパッキング済みデータと共にロードすることによって、整合動作を、さらなるロード動作無しで限られた範囲内で実行する。複数のワードを用いて形成されているゼロ探索ベクトルの場合、ゼロ探索ベクトルが複数のワードから抽出されて、基準映像又は映像ピラミッドに対して用いられる特定のパッキング済みドメイン表示物に従って（必要に応じて）フォーマットされる。ゼロ探索ベクトルデータは、例えば、前の（すなわち、より粗である）ピラミッドを用いて与えられた先速度を含むか又は、このゼロ探索ベクトルデータは単に、探索される領域の左上部分を含む。この方法 1200 は次にステップ 1208 に進む。

【0179】

ステップ 1208 では、現行ブロックと探索ベクトルデータによって識別された探索間の誤差が計算される。例えば、現行画素ブロックは探索ベクトルデータによってその位置を識別された画素ブロックと XOR 演算されて、この 2 つの画素ブロック間の歪み（すなわち、差）が決定される。この歪みは、既述したように、2 つの画素ブロック同士の差（SAD）の和を含んでいる。論理 XOR 比較の場合、この SAD は二進法の 1 のカウント値を含むが、この場合、1 は各々が、基準の画素ブロックの対応するビット又は画素と整合しない現行画素ブロックのビット又は画素を表している。この歪みの大きさは、2 つの画素ブロック間の整合（完全な整合では歪みはゼロ）又は整合の欠如を示している。方法 1200 は次にステップ 1210 に進む。

【0180】

ステップ 1210 では、ステップ 1208 の計算の結果整合が発見されたかどうか照会される。すなわち、ステップ 1210 では、2 つの画素ブロックに対する排他委的演算すなわち論理演算によって与えられた二進法の 1 の数が整合又は近整合を示すしきい値レベル未満であるかどうか照会される。ステップ 1210 での照会に対する回答が肯定的であれ

10

20

30

40

50

ば、方法 1 2 0 0 はステップ 1 2 1 4 に進む。ステップ 1 2 1 0 での照会に対する回答が否定的であれば、方法 1 2 0 0 はステップ 1 2 1 4 に進む。

【 0 1 8 1 】

ステップ 1 2 1 4 では、動作ベクトルが決定される。すなわち、現行画素ブロックと整合する画素ブロックに対して現行ブロックを関連付ける動作ベクトルが決定される。例えばビデオエンコーダ中での後続の符号化ステップの間、現行画素ブロックは動作ベクトル（整合する基準ブロックを指示する）と残余（現行画素ブロックと整合する画素ブロック間の符号化済みの差）として符号化される。方法 1 2 0 0 は次にステップ 1 2 1 6 に進む。

【 0 1 8 2 】

ステップ 1 2 1 6 では、現行の画素ブロックが現行の映像又は映像ピラミッドの再度の画素ブロックであるかどうか照会される。ステップ 1 2 1 6 での照会に対する回答が肯定的であれば、方法 1 2 0 0 はステップ 1 2 2 0 に進み、脱出する。ステップ 1 2 1 6 での照会に対する回答が否定的であれば、方法 1 2 0 0 はステップ 1 2 1 8 に進む。

【 0 1 8 3 】

ステップ 1 2 1 8 では、現行画素ブロックと識別された画素ブロックは、予測的に符号化される映像フレーム内の次の画素ブロックに等しくなるように設定される。次の画素ブロックは次に、現行画素ブロックとして 1 つ以上のレジスタに記憶される。方法 1 2 0 0 は次にステップ 1 2 0 6 に進み、ここでプロセスが繰り返される。

【 0 1 8 4 】

ステップ 1 2 1 2 では、ステップ 1 2 0 8 で現行画素ブロックと探索ベクトル出たによって識別された基準フレームの画素ブロック間の差の計算値が記憶される。具体的には、排他的論理 OR (XOR) 演算で得られた 1 の数によって決まる誤差レベル又は歪みの場合、1 の数を示す数値（すなわち、1 の数の和）が記憶されて基準画素ブロックと関連付けられる。方法 1 2 0 0 は次にステップ 1 2 2 2 に進む。

【 0 1 8 5 】

ステップ 1 2 2 2 では、探索ウィンドウが水平方向にシフトする。すなわち探索ウィンドウは例えば 1 画素分だけ左又は右へ水平にシフトする。本発明の 1 つの実施形態では、ステップ 1 2 0 6 で初期値のゼロはベクトルデータを探索して、参照フレームの左上方においてウィンドウを記述する。この実施形態では、探索ウィンドウは参照フレームの全範囲にわたって左から右へ水平にシフトする。参照フレームの端に達すると、ウィンドウは 1 つのラインによって垂直下方にシフトし、さらに 1 画素分だけ右から左へシフトする。方法 1 2 0 0 は次にステップ 1 2 2 4 へ進む。

【 0 1 8 6 】

事前定義された範囲内での探索ウィンドウのシフトは追加データのロードすることなしに達せられる。その理由は、レジスタファイル内にロードされた参照画素ブロック（又はその部分）はパックされたドメインデータの左右のオーバーラップ部分のいくつか又は全てを含むからである。従ってシフト操作はロードされた参照画素のブロック行を現在の画素のブロック行に再整合する。例えば、± 2 画素の水平探索範囲が 1 6 ビット行の画素に用いられ、またステップ 1 2 0 6 で 2 0 ビットワードがレジスタ内にロードされれば、レジスタの再ロードを行うこと無く 5 回のシフト及び比較操作を実行することができる。これはメモリリソース利用におけるかなりの節約を意味する。事前定義されたオーバーラップ範囲を超えた場合には、レジスタ内にロードされた様々なワードを（シフト及び論理和の操作を用いて）フォーマットすることが必要となる。

【 0 1 8 7 】

ステップ 1 2 2 4 では、水平方向の範囲オーバー状態が存在するかどうかについて照会がなされる。すなわち、ステップ 1 2 2 4 における探索ウィンドウの水平方向シフトが探索されている参照フレームの境界を超えたかどうかについて照会が行われる（例えば、5 回のシフト及び）。選択的にオーバーラッピング記憶方法が用いられている場合は、ステップ 1 2 2 4 での照会によって、その範囲オーバー部分（例えば、右への範囲オーバー、又は左への範囲オーバー）がシフト動作によって超えられたかが示される。ステップ

10

20

30

40

50

1 2 2 4での照会が否定的な回答を得た場合は、方法1 2 0 0はステップ1 2 0 8に進む。ステップ1 2 2 4での照会に肯定的な回答があった場合は、方法1 2 0 0はステップ1 2 2 6へ進む。

【0 1 8 8】

ステップ1 2 2 6では、探索ウィンドウは例えば1つのラインによって垂直にシフトする。本発明の1つの実施形態では、探索ウィンドウは水平の範囲オーバー状態が存在するまで右にシフトした後、1つのラインによって下方にシフトする。その後、ステップ1 2 2 4における照会によって記されたように左の境界が横断されるまで、探索ウィンドウの水平シフトは右から左に進行する。次に方法1 2 0 0はステップ1 2 2 8に進む。ステップ1 2 2 8では、垂直方向の範囲オーバー状態が存在するかどうかについての照会がなされる。すなわち垂直シフトが、検査されている参照フレームを超えたかどうかについての参照が行われる。ステップ1 2 2 8での照会に否定的な回答があれば、方法1 2 0 0はステップ1 2 0 8に進む。ステップ1 2 2 8での照会に肯定的な回答があれば、方法1 2 0 0はステップ1 2 3 0に進む。

【0 1 8 9】

ステップ1 2 3 0では、最小記憶誤差が現在の画素ブロックに対する整合として選択される。すなわちステップ1 2 3 0で、現在の画素ブロックに対する可能な整合の全てが検査され、各検査の誤差がステップ1 2 1 2で記憶された。ステップ1 2 3 0では、現在の画素ブロックが誤差についての最小差を持つ画素ブロックと関連するように、最小誤差が選択される。標準的なビデオエンコーダの場合この最小誤差がある閾値レベルを超えると、ビデオエンコーダは現在の画素ブロックのフレーム内コーディングの実行を選択できる。同様に誤差が第2の閾値レベルより低ければ、ビデオエンコーダは現在の画素ブロックのフレーム間コーディングの実行を選択して、フレーム間コーディング済画素ブロックを運動ベクトルと残余の誤差に関連付けることができる。次に方法1 2 0 0はステップ1 2 1 4へ進む。

【0 1 9 0】

図1 2を参照して上述した方法1 2 0 0は、画素ブロックベースによる画素ブロックの歪み計算を用いることによって画素ブロックに整合する。すなわち比較される2つの画素ブロックはそれぞれレジスタファイル内にそっくりロードされる。複数の水平位置のそれぞれに対して、各レジスタファイルの排他的論理和の演算が実行される（すなわち、現在の画素ブロックは（全てのオーバーラップを含む）参照画素ブロックに関して複数の水平位置を通じてシフトする）。この方法は非常に良く機能する、とはいってもそれは必然的に画素ブロックを保持するために十分な数のレジスタを必要とする。今度はレジスタ制約下の処理環境内での使用に適した方法を図1 3を参照して説明する。

【0 1 9 1】

部分歪み測定法を用いて現在のブロックと探索ブロック間の誤差を計算する方法を図1 3に示す。具体的に言えば図1 3には、（全歪みの値ではなくて）部分歪み値を生成するための（画素別ではなくて）行毎の方法1 3 0 0が示される。図1 3の部分歪み方法1 3 0 0の教示内容は図1 2に関連して既述されたより一般的な場合にも容易に適用できる。

【0 1 9 2】

図1 3の方法1 3 0 0は、比較とシフトの機能を繰り返し実行することによって、現在の画素ブロック内のN画素の各行を参照画素フレーム内の画素の対応する行全体の複数のN画素部分と比較する。すなわち現在の画素ブロックは、参照画素フレーム内の画素の対応する行の部分と比較される。現在の画素ブロックの各行が参照画素フレーム内のそれぞれの対応する行と比較された後、現在の画素ブロックの行と参照画素ブロックの行との間の対応が垂直にシフトされて、処理が繰り返される。

【0 1 9 3】

例えば8行画素ブロックを用いる本発明の実施形態では、現在の画素ブロックの8行が参照画素フレームの対応する8行と比較される（例えば、現在の画素ブロックの行1から8が参照画素フレームの行1から8に対応する）。次に行間の対応が1行だけ垂直にシフト

10

20

30

40

50

され（すなわち、現在の画素ブロックの行 1 から 8 は今度は参照画素フレームの行 2 から 9 に対応する）て、図 13 の方法 1300 が繰り返される。全ての行が比較されるまでこのプロセスが繰り返される。

【0194】

図 13 の方法 1300 はステップ 1305 に入ってからステップ 1310 に進む。ステップ 1310 では、現在の画素ブロックの第 1 行が第 1 レジスタ内に記憶される。方法 1300 は次にステップ 1315 に進み、そこで現在選択されている参照ブロックの第 1 行と全てのオーバーラップが第 2 レジスタ内に記憶され得る。方法 1300 は次にステップ 1320 へ進む。

【0195】

ステップ 1320 では、第 1 レジスタと第 2 レジスタのコンテンツ間の歪みが計算されて部分（すなわち、画素全体に関する部分的）歪みとして記憶される。すなわち第 1 レジスタと第 2 レジスタに排他的論理和の演算が行われて結果をもたらす。その結果における「1」の数が累算されて、部分歪みと見なされる値を生成する。そしてその部分歪みは記憶される。方法 1300 は次にステップ 1325 に進む。

【0196】

ステップ 1325 では、第 2 レジスタが 1 画素値（すなわち、N ビット）だけシフトされる。次に方法 1300 はステップ 1330 に進み、そこでレジスタの境界が超えられたかどうかについての照会が行われる。すなわち、ステップ 1325 での N ビットのシフト操作によって、第 1 レジスタ内に記憶された画素の行が第 2 レジスタ内に記憶されたいかなる（又は不十分な）画素とももはや整合しないかどうかについての照会が行われる。ステップ 1330 での照会に否定的な回答があれば、方法 1300 はステップ 1335 へ進む。ステップ 1330 での照会に肯定的な回答があれば、方法 1300 はステップ 1340 へ進む。

【0197】

ステップ 1340 では、参照画素フレームからの新たなワードと全てのオーバーラップが第 2 レジスタ内にロードされる。すなわち第 2 レジスタの先行してロードされた画素に（参照フレーム内で）水平方向に近接する画像グループが第 2 レジスタ内にロードされる。方法 1300 は次にステップ 1345 に進み、そこで新しいワードが必要に応じて（すなわち、新しいワードがパックされたデータ表示に適合しなければ）フォーマットされる。方法 1300 は次にステップ 1320 へ進む。

【0198】

ステップ 1335 では、計算（ステップ 1320）とシフト（ステップ 1325）のステップが、適切な水平方向の探索位置（すなわち、参照画素フレーム内の画素の位置）のそれぞれに対して実行されたかどうかについての照会が行われる。ステップ 1335 での照会が否定的な回答を得た場合は、方法 1300 はステップ 1320 に進む。ステップ 1335 での照会が肯定的な回答を得た場合は、方法 1300 はステップ 1350 に進む。

【0199】

ステップ 1350 では、ブロック内の全ての行が検査されたかどうかの照会が行われる。すなわち、計算（ステップ 1320）とシフト（ステップ 1325）のステップを通じて、参照画素フレーム内の画素の適切な近接行が対応する現在の画素ブロック行と比較されたかどうかについて照会が行われる。ステップ 1350 での照会が否定的な回答を得れば、方法 1300 はステップ 1355 に進む。ステップ 1350 での照会が肯定的な回答を得れば、方法 1300 はステップ 1365 に進む。

【0200】

ステップ 1355 では、現在の画素ブロックの次の行が第 1 レジスタ内に記憶される。次に方法 1300 はステップ 1360 に進み、そこで参照画素ブロックの次行プラス全てのオーバーラップが第 2 レジスタ内に記憶される。次に方法 1300 はステップ 1320 へ進む。

【0201】

10

20

30

40

50

ステップ 1 3 6 5 では、記憶された部分歪みが処理されて、現在の画素ブロックと参照画素ブロック間の相関関係についての指示が与えられる。すなわち、現在の画素ブロックの行の各水平位置について計算された部分歪みの値が、画素ブロックの他の行の各対応水平位置について計算された対応する部分歪みと関連付けられる。この様にして、現在のブロックと参照画素ブロックを関連付ける全体的な又は全歪みの値が決定される。

【 0 2 0 2 】

参照画素フレーム内の垂直方向探索位置のそれぞれについて、図 1 の方法 1 3 0 0 が繰り返される。図 1 3 の方法 1 3 0 0 は、現在の画素ブロックを複数の水平に近接する画素ブロックと比較することに注意されたい。例えば 4×4 画素ブロックが 4×6 参照ブロックと比較される場合には、部分歪み測定法が各行に対して実行される。従ってその機能は現在の画素ブロック行のそれぞれを 3 つの別個の参照画素ブロックの対応する行と比較することである。

10

【 0 2 0 3 】

図 1 3 の方法 1 3 0 0 では、特定の水平探索位置に対応する部分歪みはそれらが利用可能になるにつれて蓄積され、それによって探索を実行する処理システムのレジスタの要件を低減することは注意すべき重要な点である。

【 0 2 0 4 】

本発明の 1 つの重要な実施形態は、レジスタのリソースが、現在の画素ブロック行、(全てのオフセットを含む) 対応する参照ブロック行、及び処理される部分歪み測定値の全部を記憶することができないような「レジスタ制約下の」処理システムの場合に特に有効である。そのようなシステムでは方法 1 3 0 0 は水平方向探索位置の「パッチ」を処理するように適応することができる。例えば 4×4 現在画素ブロックと対応する 4×9 参照画素ブロックの場合には、方法 1 3 0 0 を用いて各行を 2 度処理することができる。具体的には、現在行と参照行との 6 回の比較を行う代りに、方法 1 3 0 0 は 3 回の比較を 2 セット行うように適応する。この様にして、画素データを記憶し歪みの結果を蓄積するのに必要なリソースが低減される。

20

【 0 2 0 5 】

本発明の 1 つの実施形態では、データのパッキングが「リアルタイム」で実行される。すなわち、パッキング方法 1 2 0 0 において例えばそれぞれのステップ 1 2 0 6 と 1 2 0 8 の間に実行されるシフトと論理和の演算は、(例えば、ピラミッド計算のための) M 進法ピラミッドの計算中に実行される。この様にしてメモリ帯域幅の利用及びパックされていない M 進法の値の記憶が回避される。例えば図 7 の方法 7 0 0 において、ステップ 2 1 0 と 2 1 5 は結合されて、ステップ 2 1 0 は参照フレームの 1 部分の M 進法ピラミッドを構成し、一方ステップ 2 1 5 は (ステップ 2 1 0 で先に処理された) M 進法ピラミッドの先に構成された部分をパックし記憶する。

30

【 0 2 0 6 】

本発明の 1 つの実施形態では前記の動作推定機能の実施後、エンコーダ 1 0 4 はマクロブロックのパイプライン式方法を利用して残余の符号化機能を遂行する。マクロブロックのパイプライン化は、次のマクロブロックに進む前にマクロブロックを完全に処理することによってメモリ帯域幅の要件を最小化するために用いられる。すなわちパイプライン式の実施形態は、次のマクロブロックを処理するのに先立って各マクロブロックに対して多くの符号化ステップを実行する。この様にして、大きなマクロブロック (すなわち、映像フレーム) グループを処理するのに必要な「読み込み、処理、記憶」の計算の繰り返しを著しく減少させることができる。

40

【 0 2 0 7 】

パイプライン式アーキテクチャは複数の処理エレメント (P E) を用いて実現され、各 P E はそれぞれの例えばマクロブロックを処理して処理済マクロブロックを生成する。例えば各 P E は符号化モジュール 1 0 4 のために実行される機能の少なくとも一部を含むプロセッサ機能から構成されており、従って各 P E は、各マクロブロックに対する例えば動作推定、モード判断、動作補償、DCT 変換及び量子化の諸機能を遂行する。通常のメモリ

50

モジュールを全ての P E のために用いてもよいが、各 P E はメモリと関連付けられている。エンコーダ 104 の逆量子化、逆 DCT 変換及び可変長コーディングの諸機能をまた各 P E の機能の中に包含して、(その効果において) P E が単一マクロブロックの符号化専用の全符号化モジュールから成るようにすることができる。複数の P E の出力は出力バッファ 160 にカップリングされる。

【0208】

本発明の 1 つの実施形態において、符号化されるフレームが P フレーム又は B フレームの場合には、1997 年 12 月 31 日出願の米国特許出願第 08/002,258 号(代理人整理番号第 12455 号)に開示され参照してここに組込まれる方法を用いて、3 つの大きさのタイリングを持つ 2 進法ピラミッド動作推定法(BPME)が実行される。簡単に言えば、フレーム、奇数フィールド及び偶数フィールドに対して動作推定が実行される。フレームが B フレームの場合は、前方及び後方動作推定の両方が実行される。バックされた画素記憶を持つ BPME によって、必要な演算の数とともにメモリ帯域幅の要件も効果的に低減される。各プロセッサが映像(例えば、スライス、マクロブロック又はブロック)のそれぞれの部分を処理するように作動する、パイプライン又は多重プロセッサ配置内の複数のプロセッサを、利用可能なメモリリソースに応じて選択的に用いることによって、BPME の手順はさらに加速される。プロセッサ間のデータアクセス時間を最小化するために、パイプライン又は多重プロセッサ配置の処理エレメント(P E)はオーバーラッピングデータ領域を選択的に利用し、それによってプロセッサ間のデータをコピーする操作の必要性を低下させる。

【0209】

パイプライン化された処理アーキテクチャ内のマクロブロックを処理する方法の工程系統図を図 4 に示す。具体的に言えば、図 4 の方法 400 は多重処理環境内の複数の処理エレメントのそれぞれによって用いられ、そこではあるマクロブロックを次のマクロブロックに進む前に完全に処理することによって、メモリ帯域幅が最小化される。図 4 の方法 400 は多重処理環境に良く適しているが、もちろん単一の処理エレメントだけを用いて実行してもよい。方法 400 は 16×16 マクロブロックの例で議論されるが、方法 400 の教示に従って他の領域サイズも処理可能であることは当業者に容易に理解されよう。

【0210】

処理される情報フレーム内のマクロブロックが、マクロブロックに対して「最良の」動作推定に帰結する動作推定処理を受けた後で、方法 400 はステップ 405 に入る。次に方法はステップ 410 に進む。

【0211】

ステップ 410 では、現在のフレームのマクロブロックに対する 16×16 画素領域がキャッシュメモリ内にロードされる。次に方法はステップ 415 へ進む、そこではフレームモードに対する最良の動作推定を中心とする 18×18 画素領域(又はフィールドモードに対する最良の動作推定を中心とする 18×10 画素領域)が、それぞれの参照フレームからメモリ内にロードされる。

【0212】

ステップ 410 で利用された参照フレームが例えば連続するステップ 415 - 1 及び 415 - 2 に従って生成される。具体的には、ステップ 415 - 1 では双線形の補間法を用いて参照フレーム領域が補間され、ステップ 415 - 2 では 9 個の半画素位置のそれぞれに対する絶対差合計(SAD)が計算される。次にモード判断の基準としての平均 2 乗誤差を用いて、最良の半画素動作推定が各モード(フレームモード又はフィールドモード)に対して選択される。この様にして、ロードする最大画素数は例えば P フレームに対しては 1300 となり、B フレームに対しては 2700 となる。ライン方向の補間を行うことによってデータサイズが適度なレベル 1 のキャッシュサイズ内に効果的に保たれる。従ってデータの入手不可能性に起因する、キャッシュミスに対するサイクルペナルティ及び中央処理装置/デジタル信号プロセッサ(CPU/DSP)のアイドル時間が減少する。

【 0 2 1 3 】

18 × 18 画素領域をメモリ内にロードした（ステップ 4 1 5）後、方法 4 0 0 はステップ 4 2 0 に進み、そこでマクロブロックレベル量子化器のスケールが例えばレート制御モジュールによって計算される。次にマクロブロックレベル量子化器のスケールはモード判断モジュールと量子化モジュールに渡される。例示の実施形態では、量子化器のスケール計算は従来型の T M 5 の方法では行われない。むしろ量子化器のスケール計算は、この開示内の別の場所で説明される方法に従って（例えば、モデルベースのフレームレベル量子化でバッファ飽和度ベースの量子化器スケールを平均することによって）行われ、それによって映像フレーム全体にわたって平滑な映像の質を保証する。方法 4 0 0 は次にステップ 4 2 5 に進む。

10

【 0 2 1 4 】

ステップ 4 2 5 ではコーディングモードが選択される。モード判断モジュールは先ず、半画素動作補償された歪みに基づいて、頂部から頂部と頂部から底部フィールド、及び底部から頂部と底部から底部フィールドの間で、フィールドを決定するための選択を行う。次にそれはまた必要に応じて、マクロブロック間の変化及びフレーム差の変化を計算する。代わりに、フレーム差の変化は情景変化の検出中にプリプロセッサ内で計算されて、データへの再アクセスを最少化する。他のモードに対しては、半画素への細分それ自体において得られる歪みが用いられる。動作ベクトルをコーディングするのに要するビット数とオーバーヘッドビットが、各モードについて推定される。レート制御モジュール 1 4 0 はマクロブロックレベル量子化器のスケールを計算してそれをモード判断モジュール 1 0 5 へ渡す。マクロブロックに対する最良のモード判断は、その基礎を所与のマクロブロック量子化器スケールにおけるレート及び歪みの双方に置くことによって実現される。この機能を実行するための適切な装置が、1996 年 10 月 23 日に出願され、参照してその全体がここに組込まれる米国特許第 0 8 / 7 3 8 2 2 8 号の中でより詳細に説明されている。方法 4 0 0 は次にステップ 4 3 0 に進む。

20

【 0 2 1 5 】

ステップ 4 3 0 では動作補償が要求されているかどうかについての照会が行われる。ステップ 4 3 0 での照会が否定的な回答を得た場合は、方法 4 0 0 はステップ 4 4 5 へ進む。ステップ 4 3 0 での照会が肯定的な回答を得た場合は、方法 4 0 0 はステップ 4 3 5 へ進む。

30

【 0 2 1 6 】

ステップ 4 3 5 では、最良の動作推定としてキャッシュ内に現在記憶されているデータを用いて半画素の補償が繰り返される。さらにマクロブロックについてのクロミナンス成分がロードされて、動作補償の操作で用いられる。方法は次にステップ 4 4 0 へ進み、そこで動作補償された予測がマクロブロックから減算されて、残余が得られる。方法は次にステップ 4 4 5 に進む。

【 0 2 1 7 】

ステップ 4 4 5 では例えば処理されている 6 つのマクロブロックのそれぞれに対して離散コサイン変換が実行される。方法は次にステップ 4 5 0 に進み、そこで変換された値はレート制御モジュールによって計算された量子化マトリックスと量子化スケールファクタを用いて量子化され、次にステップ 4 5 5 に進みそこで、V L C モジュールは動作ベクトルと量子化済 D C T 係数を符号化する。動作ベクトルと量子化済 D C T 係数は出力バッファと逆 D C T モジュールにカップリングされる。次に方法 4 0 0 はステップ 4 6 0 に進む。

40

【 0 2 1 8 】

ステップ 4 6 0 では、逆量子化モジュールと逆 D C T モジュールが残余を、それらがデコードによって見られるように復元する。方法 4 0 0 は次にステップ 4 6 5 に進み、復元された残余が予測に加算されて、再構成されたマクロブロックを得る。キャッシュサイズがより多くのマクロブロックを組込むことを可能にするならば、次のモジュールに行く前に（1 つのマクロブロックではなくてむしろ）1 セットのマクロブロック上で各モジュールの動作を繰り返すことによって、命令帯域幅を最小化できる。命令アクセスがより通常で

50

あるので、先取りメカニズムを持つアーキテクチャは前もって命令を容易に取出すことができ、それらをどのような待ち時間も無しに利用可能にする。パラレルプロセッサ又は単一命令 (SIMD) プロセッサについては、複数のマクロブロックを単一命令ストリームで同時に処理することができる。これを図 1 4 A と 1 4 B を参照して以下により詳細に説明する。方法 4 0 0 は次にステップ 4 7 0 に進む。

【 0 2 1 9 】

ステップ 4 7 0 では、フレームの最後のマクロブロックが処理されたかどうかについての照会が実行される。ステップ 4 7 0 での照会が否定的な回答を得た場合は、方法 4 0 0 はステップ 4 7 5 へ進み、そこでフレーム内の次のマクロブロックが検索されて、ステップ 4 1 0 に進み検索されたマクロブロックが処理される。ステップ 4 7 0 での照会の回答が肯定的であれば、方法 4 0 0 はステップ 4 8 0 に進みそこでそれは終了する。

10

【 0 2 2 0 】

図 1 の M P E G のようなエンコード内で用いるのに適したパイプライン化された処理アーキテクチャを図 1 4 A に示す。具体的には、図 1 4 A は複数の処理エレメント (P E) 1 4 0 0 - 1 から 1 4 0 0 - N から成る符号化装置の一部を表す。各 P E は命令バス I 経由で命令 (例えば、符号化命令) を、データバス D 経由でデータ (例えば、マクロブロック) を受取る。各 P E はそれに応答して処理済出力データ (例えば、符号化されたマクロブロック) を生成する。

【 0 2 2 1 】

本発明の 1 つの実施形態では、図 1 4 A の複数の P E は単一命令複数データストリーム (SIMD) 処理システムとして配置され、そこで各処理エレメントはプロセス、例えばスライス、マクロブロック又はブロック等の符号化されている情報フレームのそれぞれの部分を処理するために動作する。例えば図 1 4 A は処理エレメント P E 1 - P E N を、マクロブロック 1 (M B 1) からマクロブロック N (M B N) をそれぞれ処理して処理済マクロブロック M B 1' から M B N' を生成するものとして表している。

20

【 0 2 2 2 】

特定の処理エレメントにマクロブロックを処理させるために必要な命令ストリームは同じなので、各処理エレメント 1 4 0 0 - 1 から 1 4 0 0 - N はコントローラ (図示せず) から同じ命令を受取る。しかしながら、複数命令単一データストリーム (MIMD) 処理システムは実行されて比較可能な結果を得ることができる。この様にして処理タスクは複数のプロセッサ間で分割することができ、それによって情報フレーム全体を符号化するための時間を減らすことができる。動作推定の場合には、複数命令複数データ処理システムを用いて、異なる処理エレメントを処理中情報フレームの各スライスに割り当てることができる。

30

【 0 2 2 3 】

図 1 4 A のパイプライン化された処理アーキテクチャ内での使用に適した処理エレメントの実施形態を図 1 4 B に示す。具体的には処理エレメント 1 4 0 0 はエンコードの以下の機能的構成部品の直列接続で構成される: 半画素解像度 (H P R) 動作推定機能 1 4 1 0、モード判断機能 1 4 2 0、予測機能 1 4 3 0、離散コサイン変換 (D C T) 処理機能 1 4 4 0、量子化 (Q) 機能 1 4 5 0、逆量子化 (Q^{-1}) 機能 1 4 6 0、逆 D C T 機能 1 4 7 0、及び再構成機能 1 4 8 0。

40

【 0 2 2 4 】

さらにキャッシュメモリ 1 4 9 0 が 1 つ以上の機能的構成部品に動作可能にカップリングされて、情報の一時記憶装置を作る。例えば動作推定機能 1 4 1 0 によって生成された半画素解像度歪みがキャッシュメモリ 1 4 9 0 内に記憶され、引き続いてモード推定機能 1 4 2 0 によって使用される。図 1 4 B の処理エレメント 1 4 0 0 の機能的構成部品のそれぞれは、図 1 を参照して上述された対応する機能的構成部品と実質的に同じ方法で動作するので、処理エレメントの機能的構成部品を詳細に説明することはしない。

【 0 2 2 5 】

M P E G のような符号化システム 1 0 0 は複数のモジュールから構成されるが、様々なモ

50

ジュールによって実行される諸機能を図 1 に示すように分離したモジュールに孤立化させる必要は無いことが当業者に理解されよう。例えば動作補償モジュール 145、逆量子化モジュール 125 及び逆 DCT モジュール 130 から成るモジュールのセットは一般に「埋込式デコーダ」として周知である。さらに様々な予備処理機能 (101 及び 102) 及びレート制御機能 (140) が、単一の処理エレメント (例えば、マイクロプロセッサ又はディジタル信号プロセッサ (DSP))、又は並列処理の構成で配置された又はそれぞれの機能に関連した独立モジュールとして配置された複数の処理エレメントによって実行することができる。上記のアーキテクチャは、例えば DCT 及び量子化タスクのようなある高度に複雑な計算タスクのための特殊目的ハードウェアを用いて実行することができる。

10

【0226】

本発明はこれらの処理を行うためのコンピュータ実現型の処理及び装置の形態で具体化することができる。本発明はまた、CD-ROM、ハードディスクドライブ、又は他のコンピュータ読込可能記憶媒体のような具体的な媒体の中で実現されるコンピュータプログラムコードの形態で具体化ことができ、そこではコンピュータプログラムコードがコンピュータ内にロードされてそれによって実行されるとき、そのコンピュータは本発明を実行するための装置となる。本発明はまた、例えば記憶媒体内に記憶されたものであれ、コンピュータ内にロードされ及び/又はコンピュータにより実行されるものであれ、又は例えば電線、ケーブル、光ファイバ、電磁放射線等のなんらかの送信媒体を伝って送信されたものであれ、コンピュータプログラムコードの形態で具体化ことができ、そこではコンピュータプログラムコードがコンピュータ内にロードされてそれによって実行されるとき、そのコンピュータは本発明を実行するための装置となる。汎用マイクロプロセッサ上に実現されるとき、コンピュータプログラムコードのセグメントはマイクロプロセッサを構成して特定の論理回路を形成する。

20

【0227】

本発明の教示を具体化する様々な実施形態をここで詳細に示し説明したが、当業者はこれらの教示をさらに具体化する多くの他の様々な実施形態を容易に考案することができる。

【図面の簡単な説明】

【図 1】 MPEG のような符号化システムの高レベルブロック図である。

30

【図 2】 図 1 の MPEG のような符号化システムでの使用に適した適応レート制御方法のフローチャートである。

【図 2A】 図 1 の MPEG のような符号化システムでの使用に適した適応レート制御方法のフローチャートである。

【図 2B】 図 1 の MPEG のような符号化システムでの使用に適した適応レート制御方法のフローチャートである。

【図 2C】 図 1 の MPEG のような符号化システムでの使用に適した適応レート制御方法のフローチャートである。

【図 3】 図 1 の MPEG のような符号化システムでの使用に適した方法のフローチャートである。

40

【図 4】 パイプライン式処理アーキテクチャでマクロブロックを処理する方法のフローチャートである。

【図 5】 本発明を理解する際に有用な計算デバイスの 1 部分の図である。

【図 6】 ブロックベースの動作推定用の動作ベクトルを決定する際の計算上の複雑さを軽減する方法のフローチャートである。

【図 7】 平均ピラミッドを発生するために用いられる量子化プロセスを示すグラフである。

【図 8】 M が 3 である、三進法ピラミッドを発生するために用いられる量子化プロセスを示すグラフである。

【図 9】 本発明を理解する際に有用な画素情報をパックデータ又はブロックデータで表

50

示した図である。

【図 10】 本発明を理解する際に有用な画素情報をパックデータ又はブロックデータで表示した図である。

【図 11】 本発明を理解する際に有用な画素情報をパックデータ又はブロックデータで表示した図である。

【図 12】 本発明による画素ブロック整合を探索する方法のフローチャートである。

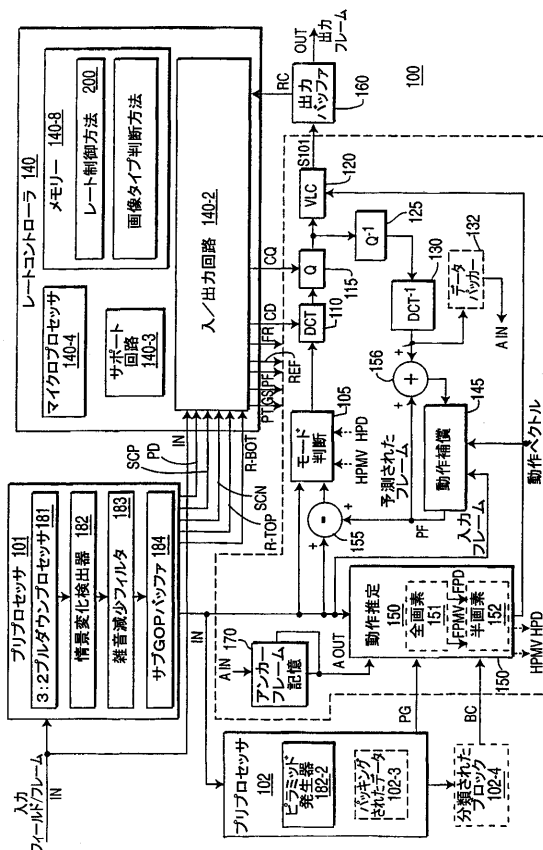
【図 13】 本発明による画素ブロック整合を探索する方法のフローチャートである。

【図１４Ａ】 図１のＭＰＥＧのようなエンコーダでの使用に適したパイプライン式処理アーキテクチャの図である。

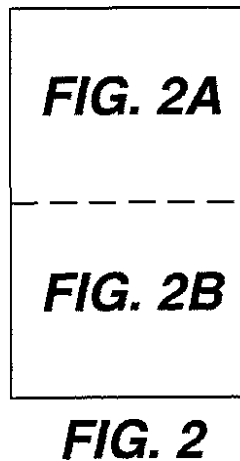
【図 1 4 B】 図 1 4 A のパイプライン式処理アーキテクチャでの使用に適したエレメント処理の実施形態である。

10

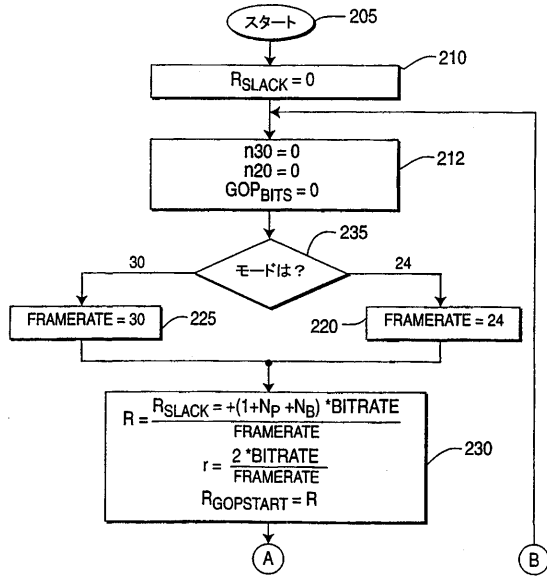
【 図 1 】



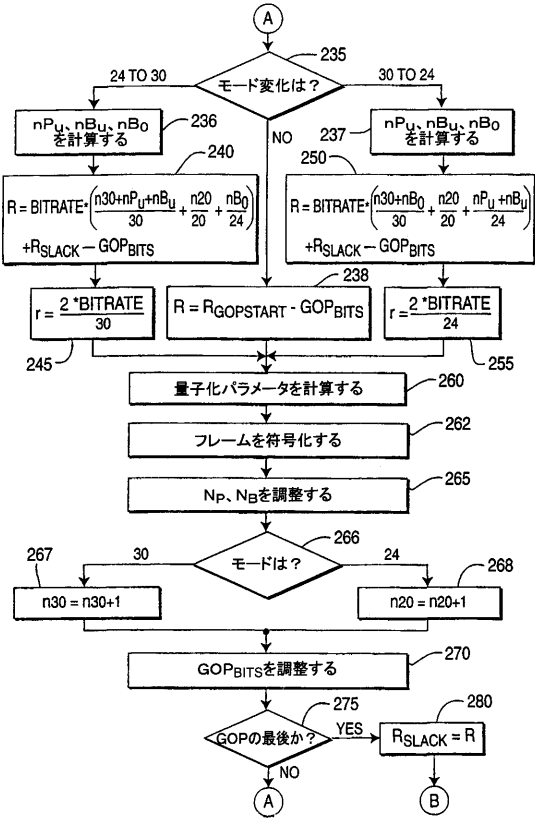
【 図 2 】



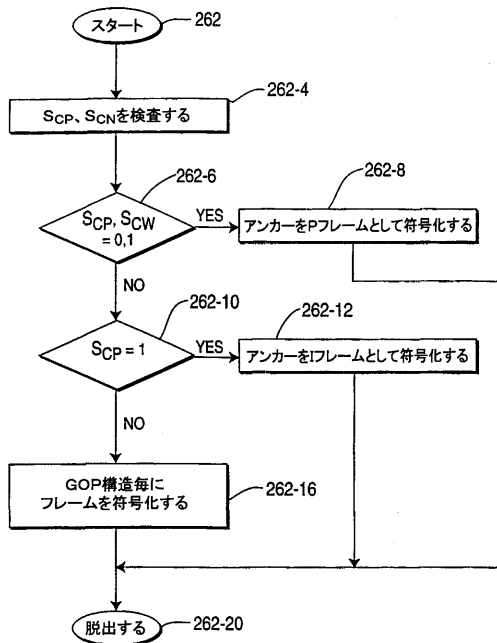
【図 2 A】



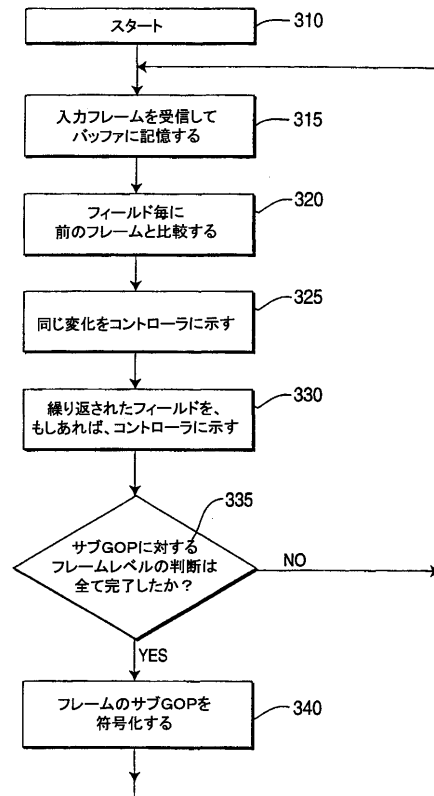
【図 2 B】



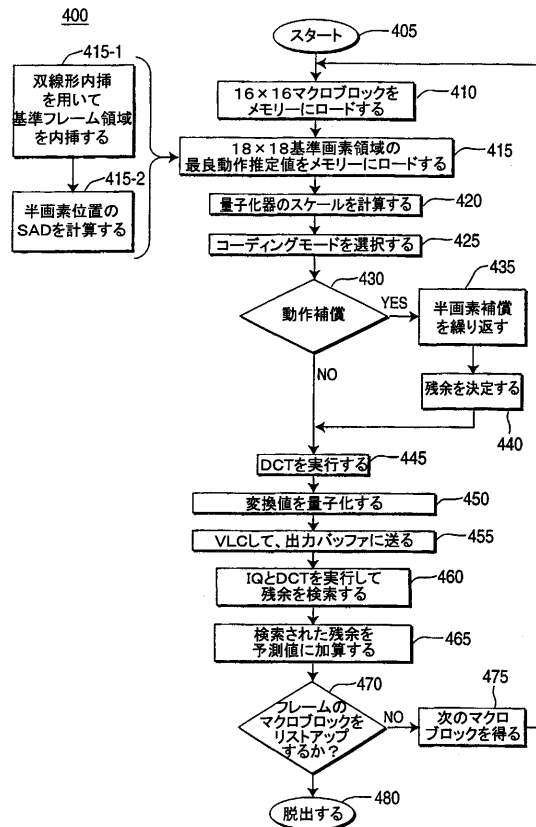
【図 2 C】



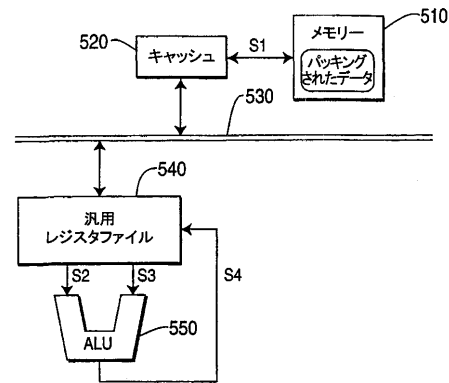
【図 3】



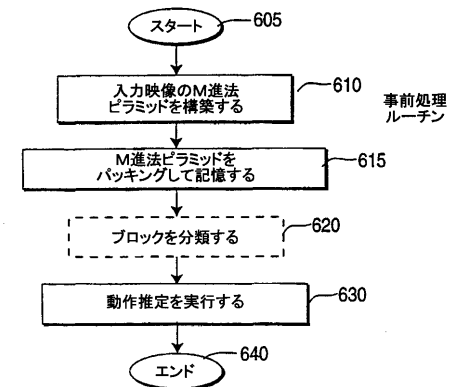
【図 4】



【図 5】



【図 6】



【図 7】

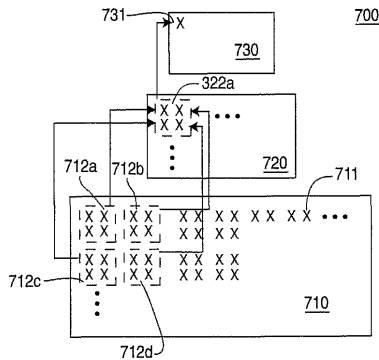


FIG. 7

【図 8】

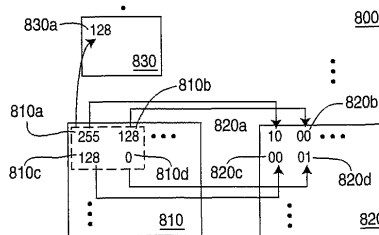
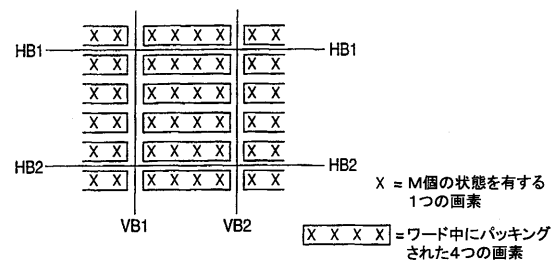


FIG. 8

【図 9】



【図 10】

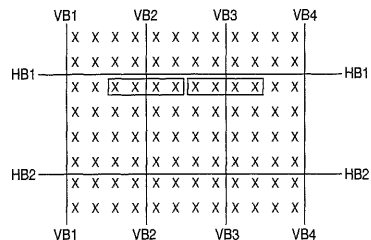


FIG. 10

【図 11】

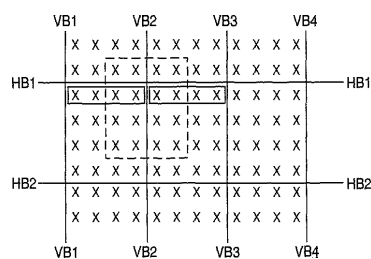
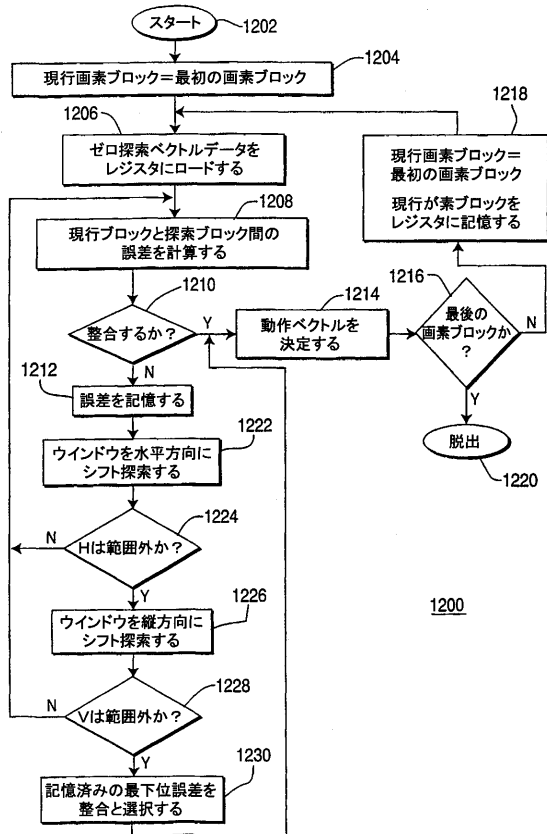
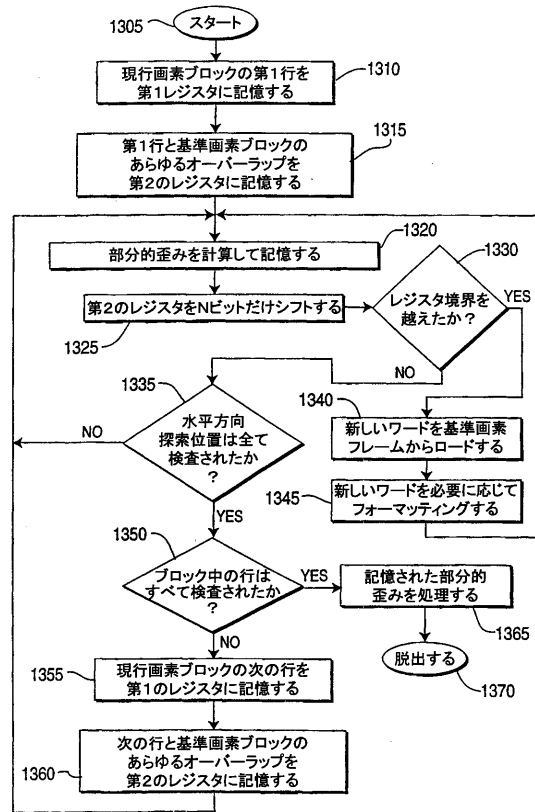


FIG. 11

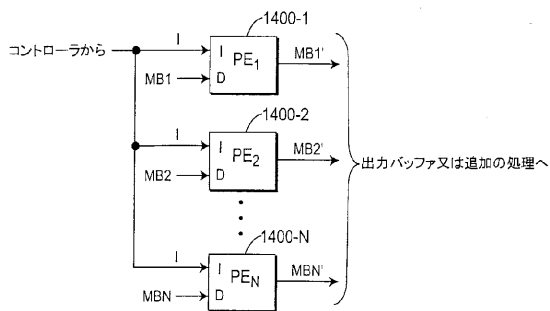
【図 1 2】



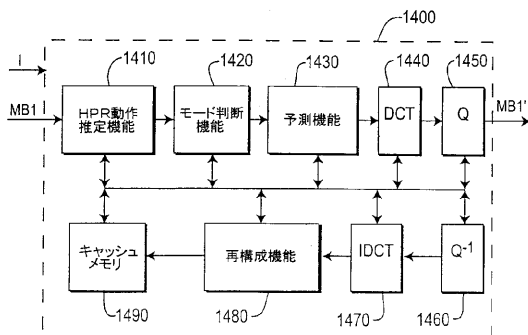
【図 1 3】



【図 1 4 A】



【図 1 4 B】



フロントページの続き

(31)優先権主張番号 09/151,425

(32)優先日 平成10年9月11日(1998.9.11)

(33)優先権主張国 米国(US)

(31)優先権主張番号 09/196,072

(32)優先日 平成10年11月19日(1998.11.19)

(33)優先権主張国 米国(US)

(31)優先権主張番号 09/285,582

(32)優先日 平成11年4月2日(1999.4.2)

(33)優先権主張国 米国(US)

(72)発明者 チャン, ティハオ

アメリカ合衆国, ニュー ジャージー州, ブレーンズボロ, フォックス ラン ドライヴ
5 - 0 4

(72)発明者 ソン, クドン

アメリカ合衆国, ニュー ジャージー州, プリンストン, トリニティ コート 3 0 9 ア
パートメント 6

(72)発明者 クリシュナムルシー, ラヴィ

アメリカ合衆国, ニュー ジャージー州, プリンストン, ブルースプリング ロード 7 0
6

(72)発明者 ハトラック, ポール

アメリカ合衆国, ニュー ジャージー州, ハイツタウン, オーク クリーク ロード 1 1
7

(72)発明者 ツァン, ヤー - チン

アメリカ合衆国, ニュー ジャージー州, クランベリー, サラトガ ドライヴ ノース 7
3

審査官 矢野 光治

(56)参考文献 特開平 0 8 - 0 7 9 7 4 6 (J P , A)

特開平 0 6 - 0 2 2 2 9 8 (J P , A)

特開平 1 0 - 0 7 5 4 5 2 (J P , A)

特開平 0 8 - 1 7 2 6 3 3 (J P , A)

特開平 0 9 - 2 6 1 6 4 8 (J P , A)

(58)調査した分野(Int.Cl., D B 名)

H04N 7/24-7/68