

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2004-532540

(P2004-532540A)

(43) 公表日 平成16年10月21日(2004.10.21)

(51) Int. Cl.<sup>7</sup>

H04N 7/32

F I

H04N 7/137

A

テーマコード(参考)

5C059

審査請求 未請求 予備審査請求 有 (全 163 頁)

(21) 出願番号	特願2002-570429 (P2002-570429)	(71) 出願人	301079899
(86) (22) 出願日	平成14年3月5日 (2002.3.5)		インタービデオインコーポレイテッド
(85) 翻訳文提出日	平成15年9月4日 (2003.9.4)		アメリカ合衆国カリフォルニア州フレモント市47350フレンモントブルバード
(86) 国際出願番号	PCT/US2002/006726	(74) 代理人	100065215
(87) 国際公開番号	W02002/071639		弁理士 三枝 英二
(87) 国際公開日	平成14年9月12日 (2002.9.12)	(74) 代理人	100094101
(31) 優先権主張番号	60/273, 443		弁理士 館 泰光
(32) 優先日	平成13年3月5日 (2001.3.5)	(74) 代理人	100114616
(33) 優先権主張国	米国 (US)		弁理士 眞下 晋一
(31) 優先権主張番号	60/275, 859	(74) 代理人	100124028
(32) 優先日	平成13年3月14日 (2001.3.14)		弁理士 松本 公雄
(33) 優先権主張国	米国 (US)	(74) 代理人	100124039
(31) 優先権主張番号	60/286, 280		弁理士 立花 顕治
(32) 優先日	平成13年4月25日 (2001.4.25)		
(33) 優先権主張国	米国 (US)		

最終頁に続く

(54) 【発明の名称】 誤り耐性のある符号化のためのシステム及び方法

## (57) 【要約】

本発明は、誤り耐性ビデオ符号化に関する。本発明の実施の形態は、ビデオデータストリームの誤り伝搬の低減、マクロブロックのリフレッシュ、ビデオデータストリーム内のフレームレートの低減、媒体ストリームの誤り訂正情報の生成、ビデオデータストリームのビットレート割り当ての実行、及びビデオデータストリーム内のシーン変化の検出を行うシステム（図1B、100B）及び方法を提供する。

## 【特許請求の範囲】

## 【請求項 1】

符号化されたビデオ送信の誤り耐性を高める方法であって、  
第 1 のフレームがシーン変化に対応することを示す指示を受信するステップと、  
前記第 1 のフレームがシーン変化に対応することを示す前記指示を受信することに、少なくとも部分的に応じて前記第 1 のフレームをイントラコーディングするステップと、  
次の連続するフレームを自動的にイントラコーディングするステップと、  
前記イントラコーディングされた第 1 のフレーム及び前記イントラコーディングされた次の連続するフレームを送信するステップとを含む方法。

## 【請求項 2】

前記イントラコーディングされた第 1 のフレーム及び前記イントラコーディングされた次の連続するフレームの前記送信が M P E G - 4 準拠である請求項 1 に記載の方法。

## 【請求項 3】

前記第 1 のフレームが I フレームとして指定される請求項 1 に記載の方法。

## 【請求項 4】

前記次の連続するフレームが I フレームとして指定される請求項 1 に記載の方法。

## 【請求項 5】

前記次の連続するフレームがシーン変化フレームでない請求項 1 に記載の方法。

## 【請求項 6】

前記次の連続するフレームの直後のフレームがインターコーディングされる請求項 1 に記載の方法。

## 【請求項 7】

前記第 1 のフレームがシーン変化に対応することを示す前記指示が、ファイルから読み込まれる請求項 1 に記載の方法。

## 【請求項 8】

ビデオコーディングコーディング装置であって、  
プロセッサ読み取り可能なメモリ内に格納される、第 1 のフレームをイントラコーディングすることを示す指示を受信するように構成されている第 1 の命令と、  
プロセッサ読み取り可能なメモリ内に格納される、前記指示に少なくとも部分的に応じて、前記第 1 のフレームをイントラコーディングするように構成されている第 2 の命令と、  
プロセッサ読み取り可能なメモリ内に格納され、前記第 1 のフレームをイントラコーディングすることを指示した結果として、前記第 1 のフレームの直後のフレームを自動的にイントラコーディングするように構成されている第 3 の命令とを有する装置。

## 【請求項 9】

前記ビデオコーディングコーディング装置が集積回路である請求項 8 に記載のビデオコーディングコーディング装置。

## 【請求項 10】

前記ビデオコーディングコーディング装置が携帯電話である請求項 8 に記載のビデオコーディングコーディング装置。

## 【請求項 11】

前記ビデオコーディングコーディング装置がデスクトップコンピュータである請求項 8 に記載のビデオコーディングコーディング装置。

## 【請求項 12】

前記ビデオコーディング装置がパーソナルデジタルアシスタントである請求項 8 に記載のビデオコーディング装置。

## 【請求項 13】

前記第 1 のフレームがシーン変化フレームである請求項 8 に記載のビデオコーディング装置。

## 【請求項 14】

前記第 1 のフレームをイントラコーディングすることを示す前記指示を格納するために使

10

20

30

40

50

用されるファイルをさらに有する請求項 8 に記載のビデオコーディング装置。

【請求項 15】

ビデオフレームを符号化する方法であって、

第 1 のフレームをイントラコーディングする第 1 の命令を受信するステップと、

前記第 1 の命令への応答として前記第 1 のフレームをイントラコーディングするステップと、

前記第 1 のフレームをイントラコーディングする前記第 1 の命令の結果として第 2 のフレームをイントラコーディングするステップとを含む方法。

【請求項 16】

前記イントラコーディングされた第 1 のフレームと前記イントラコーディングされた次のフレームを送信するステップをさらに含む請求項 15 に記載の方法。 10

【請求項 17】

前記第 1 のフレームがシーン変化フレームである請求項 15 に記載の方法。

【請求項 18】

前記第 2 のフレームに、前記第 1 の命令がなかった場合にそのフレームのインターコーディングを実行させる活動レベルが設定されている請求項 15 に記載の方法。

【請求項 19】

前記第 2 のフレームの直後にある第 3 のフレームを、少なくとも部分的に前記第 3 のフレームの相対的視覚的動きに基づいて、コーディングするステップをさらに含む請求項 15 に記載の方法。 20

【請求項 20】

符号化装置であって、

第 1 のフレームをイントラコーディングする第 1 の命令を受信する手段と、

前記第 1 の命令への応答として前記第 1 のフレームをイントラコーディングする手段と、

前記第 1 のフレームをイントラコーディングする前記第 1 の命令の結果として第 2 のフレームをイントラコーディングする手段とを備える符号化装置。

【請求項 21】

前記第 1 のイントラコーディングされたフレームと前記第 2 のイントラコーディングされたフレームとを供給する手段をさらに備える請求項 20 に記載の符号化装置。

【請求項 22】 30

前記符号化装置が集積回路に組み込まれている請求項 20 に記載の符号化装置。

【請求項 23】

複数のフレームを含むデジタルビデオシーケンス内のシーン変化を検出する方法であって、

第 2 のフレームに対する第 1 のフレーム及び第 3 のフレームに対する前記第 2 のフレームに関して、第 1 の二乗平均平方根 (RMS) 値を計算するステップと、

前記第 2 のフレームに対する前記第 1 のフレームに関して、第 1 の差の絶対値の平均 (MAD) 値を計算するステップと、

前記第 1 の RMS 値が第 1 の基準を満たしているか否かを判別するステップと、

前記第 1 の MAD 値が第 2 の基準を満たしているか否かを判別するステップと、 40

前記第 1 の RMS 値が前記第 1 の基準を満たし、前記第 1 の MAD 値が前記第 2 の基準を満たしていると判断することに、少なくとも部分的に応じて、前記第 2 のフレームをシーン変化フレームとして指定するステップとを含む方法。

【請求項 24】

前記第 1 の RMS 値が色重み付けされている請求項 23 に記載の方法。

【請求項 25】

前記第 2 のフレームについて I フレーム指定をファイルに格納し、第 3 のフレームについて P フレーム指定を前記ファイルに格納するステップをさらに含む請求項 23 に記載の方法。

【請求項 26】 50

前記第 1 の R M S 値が、少なくとも部分的に、前記第 1 及び前記第 2 のフレームのピクセルの輝度値及びクロミナンス (chrominance) 値に基づく請求項 2 3 に記載の方法。

【請求項 2 7】

前記第 1 の R M S 値が式

【数 1】

$$\begin{aligned} \text{RMS}(F_i, F_k) = & \frac{\alpha}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^w \sum_{y=1}^h |Y_i(x, y) - Y_k(x, y)|^2} + \\ & \frac{2\beta}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^{w/2} \sum_{y=1}^{h/2} |U_i(x, y) - U_k(x, y)|^2} + \\ & \frac{2\gamma}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^{w/2} \sum_{y=1}^{h/2} |V_i(x, y) - V_k(x, y)|^2} \end{aligned}$$

10

で定義され、 $F_i$  が前記第 1 のフレームであり、 $F_k$  が前記第 2 のフレームであり、 $F(x, y)$  がフレーム  $F$  内の  $(x, y)$  番目のピクセルを表し、 $w$  がフレーム幅であり、 $h$  がフレーム高さであり、 $Y(x, y)$  がピクセル輝度値に対応し、 $U(x, y)$  及び  $V(x, y)$  が色度成分に対応し、 $\alpha$ 、 $\beta$ 、及び  $\gamma$  がそれぞれ、明度、色度 - 青、及び色度 - 赤の成分に対する重み係数である方法。

20

【請求項 2 8】

$\alpha = \beta = \gamma = 1$  である請求項 2 7 に記載の方法。

【請求項 2 9】

輝度値を使用し、色度成分を除外して前記第 1 の M A D 値が計算される請求項 2 3 に記載の方法。

【請求項 3 0】

前記第 1 の基準が第 1 のしきい値であり、前記第 2 の基準が第 2 のしきい値である請求項 2 3 に記載の方法。

【請求項 3 1】

デジタルビデオシーケンス内のシーン変化を検出する方法であって、  
第 2 のフレームに対する第 1 のフレーム及び第 3 のフレームに対する前記第 2 のフレームに関して、二階時間微分 R M S 値を計算するステップと、  
前記二階時間微分 R M S 値に少なくとも部分的に基づき、前記第 2 のフレームがシーン変化フレームであると判断するステップとを含む方法。

30

【請求項 3 2】

前記第 2 のフレームがシーン変化フレームであるとの前記判断が、さらに、前記第 1 及び前記第 2 のフレームについて少なくとも明度情報を使用して計算された差の絶対値の平均値に基づく請求項 3 1 に記載の方法。

【請求項 3 3】

前記第 2 のフレームがシーン変化フレームであるとの前記判断が、さらに、第 1 の基準を満たす R M S 値及び第 2 の基準を満たす前記二階時間微分 R M S 値の両方に基づく請求項 3 1 に記載の方法。

40

【請求項 3 4】

前記二階時間微分 R M S 値が第 1 のしきい値以上である請求項 3 1 に記載の方法。

【請求項 3 5】

前記二階時間微分 R M S 値が負であり、且つ第 1 の値よりも大きい絶対値を持つ場合に、前記第 2 のフレームがシーン変化フレームとして指定される請求項 3 1 に記載の方法。

【請求項 3 6】

第 1 の R M S 値を計算するステップをさらに含み、前記第 1 の R M S 値が色重み付けされ、前記第 2 の時間微分 R M S 値が時間成分にのみ基づく請求項 3 1 に記載の方法。

50

## 【請求項 37】

前記二階時間微分 R M S 値が  $(R M S ( F_{i-1}, F_i ) - 2 R M S ( F_i, F_{i+1} ) + R M S ( F_{i+1}, F_{i+2} ))$  に等しく、 $F_{i-1}$  が前記第 1 のフレームであり、 $F_i$  が前記第 2 のフレームであり、 $F_{i+1}$  が第 3 のフレームであり、 $F_{i+2}$  が第 4 のフレームである請求項 31 に記載の方法。

## 【請求項 38】

ビデオシーケンス内のシーン変化を識別する装置であって、  
 プロセッサ読み取り可能なメモリに格納される、ビデオシーケンスの第 2 の部分に対する前記ビデオシーケンスの第 1 の部分に関して、第 1 の二乗平均平方根 ( R M S ) を計算するように構成されている第 1 の命令と、  
 プロセッサ読み取り可能なメモリ内に格納される、二階時間微分 R M S 値を計算するように構成されている第 2 の命令と、  
 前記二階時間微分 R M S 値に少なくとも部分的に基づいて、前記ビデオシーケンスの第 2 の部分をイントラコーディングするように構成されている第 3 の命令とを有する装置。

10

## 【請求項 39】

前記第 3 の命令が、第 1 の部分及び前記第 2 の部分の少なくとも明度情報を使用して計算された差の絶対値の平均値に基づき、前記ビデオシーケンスの前記第 2 の部分をイントラコーディングするように、さらに構成されている請求項 31 に記載の装置。

## 【請求項 40】

前記第 3 の命令が、第 1 の基準を満たす前記 R M S 値及び第 2 の基準を満たす前記二階時間微分 R M S 値の両方に、少なくとも部分的に応じて、前記ビデオシーケンスの前記第 2 の部分をイントラコーディングするように、さらに構成されている請求項 31 に記載の装置。

20

## 【請求項 41】

前記第 3 の命令が、前記第 2 の部分を I フレームとしてコーディングすることを指示するフラグを、プロセッサ読み取り可能なメモリ内に格納するように、さらに構成されている請求項 31 に記載の装置。

## 【請求項 42】

前記第 1 の部分が第 1 のフレームであり、前記第 2 の部分が第 2 のフレームである請求項 31 に記載の装置。

30

## 【請求項 43】

前記二階時間微分 R M S 値が負であり、且つその絶対値が第 1 の値よりも大きい場合に、前記第 3 の命令が、前記第 1 の部分をシーン変化を含むものとして識別する請求項 31 に記載の方法。

## 【請求項 44】

前記第 1 の R M S 値が色重み付けされている請求項 31 に記載の方法。

## 【請求項 45】

前記装置が集積回路である請求項 31 に記載の装置。

## 【請求項 46】

ビデオシーケンスのどの部分がイントラコーディングされるべきかを決定する方法であって、  
 ビデオシーケンスの第 1 の部分に対する第 1 の二乗平均平方根 ( R M S ) 値を計算するステップと、  
 前記ビデオシーケンスの前記第 1 の部分に関する第 1 の差の絶対値の平均 ( M A D ) を計算するステップと、  
 前記第 1 の R M S 値が第 1 の基準を満たしているか否かを判別するステップと、  
 前記第 1 の M A D 値が第 2 の基準を満たしているか否かを判別するステップと、  
 前記第 1 の M A D 値が第 3 の基準を満たしているか否かを判別するステップと、  
 前記第 1、前記第 2、及び前記第 3 の基準のうち少なくとも 2 つが満たされていることに、少なくとも部分的に応じて、イントラコーディング処理を実行させるステップとを含

40

50

む方法。

【請求項 4 7】

前記第 3 の基準が、前記 M A D 値が極大値であることである請求項 4 6 に記載の方法。

【請求項 4 8】

前記ビデオシーケンスの前記第 1 の部分が第 1 のフレームを含む請求項 4 6 に記載の方法。

【請求項 4 9】

前記ビデオシーケンスの前記第 1 の部分が第 1 の G O V を含む請求項 4 6 に記載の方法。

【請求項 5 0】

前記ビデオシーケンスの前記第 1 の部分が第 1 の G O P を含む請求項 4 6 に記載の方法。 10

【請求項 5 1】

シーン変化検出装置であって、

第 1 のフレーム情報入力及び R M S 出力を備え、前記第 1 のフレーム情報入力で受信したフレーム情報に基づき、少なくとも 2 つのフレームの間の差の二乗平均平方根 (root mean squared differences) 情報に対応する値を前記 R M S 出力から出力するように構成されている R M S 回路と、

第 2 のフレーム情報入力及び M A D 出力を備え、前記第 2 のフレーム情報入力で受信したフレーム情報に基づき、少なくとも 2 つのフレームの間の差の絶対値の平均 (mean absolute differences) 情報に対応する値を前記 M A D 出力から出力するように構成されている M A D 回路と、 20

前記 R M S 出力及び前記 M A D 出力に結合され、差の二乗平均平方根情報に対応する値及び差の絶対値の平均情報に対応する値に少なくとも部分的に基づいて、シーン変化フレームを検出し、シーン変化の指定を行うように構成されているエバリュエータ回路とを備えるシーン変化検出装置。

【請求項 5 2】

第 3 の情報入力及び二階微分 R M S 出力を備える二階微分 R M S 回路をさらに備え、前記二階微分 R M S 回路が、前記第 3 のフレーム情報入力で受信したフレーム情報に基づいて、前記二階微分 R M S 出力で差の二乗平均平方根の二階微分 (second derivative root mean squared differences) 情報に対応する値を出力するように構成され、

前記二階微分 R M S 出力が前記エバリュエータ回路に結合されている請求項 5 0 に記載の装置。 30

【請求項 5 3】

ビデオシーケンスとともに使用するビット割り当て方法であって、

少なくとも第 1 のクリップに対するクリップビット割当値を受信するステップと、

前記第 1 のクリップに対するシーン量を決定するステップと、

前記第 1 のクリップの第 1 のシーン内の予測フレームの量を決定するステップと、

前記第 1 のクリップに対する前記シーンの量、前記第 1 のクリップビット割当値、前記第 1 のシーン内のイントラコーディングフレームの量、及び前記第 1 のシーン内の予測フレームの量に少なくとも部分的に基づいて、前記第 1 のシーンのビット割当値を計算するステップとを含むビット割り当て方法。 40

【請求項 5 4】

前記第 1 のクリップ内の各シーンが、イントラコーディングフレームから始まる請求項 5 3 に記載のビット割り当て方法。

【請求項 5 5】

前記クリップビット割当値に予測フレームの数と第 1 の定数との和を乗算し、前記クリップ内のイントラコーディングフレーム及び予測フレームの量と第 2 の定数を乗算したクリップシーンの数との和で除算することに、少なくとも部分的に基づいて、前記ビット割当値を計算する請求項 5 3 に記載のビット割り当て方法。

【請求項 5 6】

前記第 1 のシーンが G O V である請求項 5 3 に記載のビット割り当て方法。 50

- 【請求項 57】  
前記第1のシーンがGOPである請求項53に記載のビット割り当て方法。
- 【請求項 58】  
前記予測フレームがP-VOPである請求項53に記載のビット割り当て方法。
- 【請求項 59】  
前記第1のシーンの前記ビット割当値及び現在のビット使用度に基づいて、前記第1のシーン内の第1の予測フレームについて量子化パラメータを調整するステップをさらに含む請求項53に記載のビット割り当て方法。
- 【請求項 60】  
前記量子化パラメータの前記調整が第1の範囲に制限される請求項59に記載のビット割り当て方法。 10
- 【請求項 61】  
現在のビット使用度が前記第1のシーンの前記ビット割当値に関係する値よりも大きいという判断に、少なくとも部分的に応じて、前記第1のシーン内の第1の予測フレームの量子化パラメータを上方向に調整するステップを、さらに含む請求項53に記載のビット割り当て方法。
- 【請求項 62】  
現在のビット使用度が前記第1のシーンの前記ビット割当値に関係する値よりも小さいという判断に、少なくとも部分的に応じて、前記第1のシーン内の第1の予測フレームの量子化パラメータを下方向に調整するステップを、さらに含む請求項53に記載のビット割り当て方法。 20
- 【請求項 63】  
チャンネルレートに少なくとも部分的に基づいて、マクロブロックの量子化パラメータを調整するステップを、さらに含む請求項53に記載のビット割り当て方法。
- 【請求項 64】  
ビデオクリップシーンとともに使用するビット割り当て方法であって、第1のシーンに関するビット割当値を計算するステップと、前記第1のシーン内の少なくとも第1のフレームを含む対応するフレームに関するビット割当値を計算するステップと、前記第1のフレーム内のマクロブロックに対応するビット割当値を計算するステップとを含むビット割り当て方法。 30
- 【請求項 65】  
前記第1のシーンがGOPである請求項64に記載のビット割り当て方法。
- 【請求項 66】  
前記第1のシーンがGOVである請求項64に記載のビット割り当て方法。
- 【請求項 67】  
前記第1のシーンに関する前記ビット割当値が、前記第1のシーンの複雑度判別に少なくとも部分的に基づく請求項64に記載のビット割り当て方法。
- 【請求項 68】  
前記第1のシーンに関する前記ビット割当値が、前記第1のシーンの複雑度判別及び複数のシーンの平均複雑度に少なくとも部分的に基づく請求項64に記載のビット割り当て方法。 40
- 【請求項 69】  
前記第1のシーンに関する前記ビット割当値が、バッファ状態に少なくとも部分的に基づく請求項64に記載のビット割り当て方法。
- 【請求項 70】  
前記第1のフレームに関する前記ビット割当値が、量子化パラメータ不変基準に少なくとも部分的に基づく請求項64に記載のビット割り当て方法。
- 【請求項 71】  
前記第1のフレームに関する前記ビット割当値が、前記第1のフレーム内にあるテクスチ 50

ビットの数及び前記第1のフレームに対する動きベクトルビットの数に少なくとも部分的に基づき請求項64に記載のビット割り当て方法。

【請求項72】

前記第1のフレームに関する前記ビット割当値がさらに、第2のフレームに対する前記第1のフレームに対応する差の絶対値の平均値に基づき請求項71に記載のビット割り当て方法。

【請求項73】

前記第1のフレームに関する前記ビット割当値が、前記第1のシーンの前記ビット割当値、前記第1のシーン内のすでにコーディングされているフレームに使用されているビット量、前記第1のシーンの複雑度、及び前記第1のシーン内のすでにコーディングされているフレームの複雑度に、少なくとも部分的に基づき請求項64に記載のビット割り当て方法。

10

【請求項74】

1つの量子化パラメータ値を使用して前記第1のフレーム内の全てのマクロブロックを量子化する請求項64に記載のビット割り当て方法。

【請求項75】

量子化パラメータが、現在のビット使用度及び割当値が設定されたビット使用度に、少なくとも部分的に基づき前記第1のフレームマクロブロックに関して変えられる請求項64に記載のビット割り当て方法。

【請求項76】

ビットを割り当てる装置であって、  
第1のシーン内に含まれるイントラコーディングフレームの数、前記第1のシーンに対する複雑度の判別、及び前記第1のシーン内に含まれるインターコーディングフレームの数に、少なくとも部分的に基づいて、前記第1のシーンに対する第1のビット割当値を割り当てるように構成されている第1の命令と、  
現在のビット使用度及び目標ビット使用度に少なくとも部分的に基づいて、前記第1のシーン内のフレームにビット割当値を割り当てるように構成されている第2の命令とを有する装置。

20

【請求項77】

前記現在のビット使用度及び前記目標ビット使用度に少なくとも部分的に基づいて、前記

30

【請求項78】

前記第1のシーン内の前記第1のフレームをイントラコーディングする請求項76に記載の装置。

【請求項79】

前記第1の命令が、前記インターコーディングフレームに割り当てる重みよりも大きい重みを前記インターコーディングフレームに割り当てる請求項76に記載の装置。

【請求項80】

前記装置が集積回路である請求項76に記載の装置。

【請求項81】

ビット割当値を計算する装置であって、  
少なくとも第1のビデオシーケンスに対するビット割当値を受信するように構成されている第1の命令と、

40

前記第1のビデオシーケンスに対するシーン量を決定するように構成されている第2の命令と、

前記第1のビデオシーケンスの第1のシーン内の予測フレーム数を決定するように構成されている第3の命令と、

前記第1のクリップの前記シーン量、前記第1のクリップビット割当値、前記第1のシーン内のイントラコーディングフレーム量、及び前記第1のシーン内の予測フレーム量に、  
少なくとも部分的に基づいて、前記第1のシーンに対するビット割当値を決定するように

50



構成されている第 4 の命令とを有する装置。

【請求項 8 2】

前記第 1 のシーンが G O V である請求項 8 1 に記載の装置。

【請求項 8 3】

前記第 1 のシーンに対する前記ビット割当値が、バッファ状態に少なくとも部分的に基づき請求項 8 1 に記載の装置。

【請求項 8 4】

前記第 1 のフレーム内にあるテクスチャビット数及び前記第 1 のフレームに対する動きベクトルビット数に少なくとも部分的に基づいて、第 1 のフレームに対するビット割当値を決定するように構成されている第 5 の命令を、さらに有する請求項 8 1 に記載の装置。 10

【請求項 8 5】

前記第 1 のシーン内の第 1 のフレームのマクロブロックについて量子化パラメータを決定するように構成されている第 5 の命令を、さらに有する請求項 8 1 に記載の装置。

【請求項 8 6】

前記装置が集積回路である請求項 8 1 に記載の装置。

【請求項 8 7】

符号化プロセスの実行中にスキップするビデオシーケンスフレームを選択する方法であって、

一連のフレームを受信するステップと、

前記一連のフレームの少なくとも一部の中のフレーム毎に、前記各フレームを挟むフレーム間の時間パラメータで重み付けされた対応する差の絶対値の平均を求めるステップと、 20  
フレーム毎に計算で求めた時間パラメータで重み付けされた前記対応する差の絶対値の平均に、少なくとも部分的に基づいて、前記フレーム列内の第 1 のフレームをスキップするステップとを含む方法。

【請求項 8 8】

ターゲットフレームレートを受信するステップと、

前記第 1 のフレームをスキップした場合に目標フレームレートの条件が満たされるか否かを判別するステップとをさらに含む請求項 8 7 に記載の方法。

【請求項 8 9】

少なくとも一部は前記第 1 のフレームがスキップされた場合に前記目標フレームレートの 30  
条件が満たされていないと判断したことに対する応答として、前記第 2 のフレームを挟むフレーム間の第 2 の時間パラメータによって重み付けされた第 2 の対応する差の絶対値の平均値に、少なくとも部分的に基づいて、スキップ対象の第 2 のフレームを選択するステップを含む請求項 8 8 に記載の方法。

【請求項 9 0】

スキップできる連続するフレーム数を第 1 の量に制限するステップをさらに含む請求項 8 7 に記載の方法。

【請求項 9 1】

第 1 のイントラコーディングフレームがスキップ対象から除外される請求項 8 7 に記載の方法。 40

【請求項 9 2】

前記一連のフレーム内の最後のフレームがスキップ対象から除外される請求項 8 7 に記載の方法。

【請求項 9 3】

前記第 1 のスキップされるフレームのヘッダを供給し、前記第 1 のフレームがコーディングされないことを指示するフラグをセットするステップをさらに含む請求項 8 7 に記載の方法。

【請求項 9 4】

ビデオシーケンス内のスキップするフレームを決定する方法であって、  
第 1 のフレームレート目標を受信するステップと、 50

ビデオシーケンスシーン内の前記第 1 のフレームと最後のフレームとの間にある、スキップする第 1 の潜在的フレームを選択するステップと、  
 前記第 1 の潜在的フレームをスキップしたことによって生じる視覚的影響に関係する、スキップする前記第 1 の潜在的フレームの第 1 のコスト値を計算するステップと、  
 前記第 1 のコスト値をコンピュータ読み取り可能なメモリ内に格納するステップと、  
 前記第 1 のフレームと前記最後のフレームとの間にある、スキップする第 2 の潜在的フレームを選択するステップと、  
 スキップする前記第 2 の潜在的フレームの第 2 のコスト値を計算するステップと、  
 前記第 2 のコスト値をコンピュータ読み取り可能なメモリ内に格納するステップと、  
 前記第 1 のフレームと前記最後のフレームとの間にある、スキップする第 3 の潜在的フレームを選択するステップと、  
 スキップする前記第 3 の潜在的フレームの第 3 のコスト値を計算するステップと、  
 前記第 3 のコスト値をコンピュータ読み取り可能なメモリ内に格納するステップと、  
 前記第 1 のコスト値、前記第 2 のコスト値、及び前記第 3 のコスト値の中のどれがコスト最小であるかを決定するステップと、  
 前記最小コスト値に対応する前記第 1、前記第 2、及び前記第 3 の潜在的フレームの中の 1 つをスキップするステップと、  
 前記第 1 のフレームレート目標が満たされているか否かを判別するステップと、  
 前記第 1 のフレームレート目標が満たされていないと判別した場合、そのことに少なくとも部分的に応じて、前記第 1、前記第 2、及び前記第 3 の潜在的フレームから残りのフレームと関連するコスト値を再計算し、前記再計算されたコスト値が最小である前記残りのフレームの中の 1 つをスキップするステップとを含む方法。

10

20

【請求項 95】

前記第 1 のコスト値が、差の絶対値の平均 (MAD) 値及び前記第 1 のフレームとスキップされる可能性のある前記第 2 のフレームとの時間差に、少なくとも部分的に基づく請求項 94 に記載の方法。

【請求項 96】

前記 MAD 値の判別の際に、前記第 1 のフレームとスキップされる可能性のある前記第 2 のフレーム内のピクセルの一部のみを評価する請求項 95 に記載の方法。

【請求項 97】

前記時間差が NTSC フレームレートに正規化される請求項 95 に記載の方法。

30

【請求項 98】

連続的にスキップできるフレーム数が第 1 の数に制限される請求項 94 に記載の方法。

【請求項 99】

スキップされるフレーム毎にヘッダを供給するステップをさらに含む請求項 94 に記載の方法。

【請求項 100】

一連のビデオパケット内の全てのパケットでシーケンス情報を含むヘッダ拡張コード (HEC) を送信するステップをさらに含む請求項 94 に記載の方法。

【請求項 101】

符号化プロセスの実行中にスキップするイメージフレームを選択する方法であって、  
 第 1 のフレームを含む一連のフレームを受信するステップと、  
 前記第 1 のフレームの境界となるフレームの間の時間差及び明度差に関係する第 1 の時間パラメータを含む、前記第 1 のフレームに対する第 1 のコスト関数値を計算するステップと、  
 第 2 のフレームの境界となるフレームの間の時間差及び明度差に関係する第 2 の時間パラメータを含む、前記第 2 のフレームに対する第 2 のコスト関数値を計算するステップと、  
 前記第 1 のコスト関数値と前記第 2 のコスト関数値とに少なくとも部分的に基づいて、前記第 1 のフレーム及び前記第 2 のフレームのうちのスキップする 1 つを選択するステップとを含む方法。

40

50

## 【請求項 102】

目標フレームレートを受信するステップと、

前記第1のフレーム及び前記第2のフレームのうちの1つをスキップする場合に前記目標フレームレートの条件が満たされるか否かを判別するステップとをさらに含む請求項101に記載の方法。

## 【請求項 103】

平均差平均関数 (mean of average difference function) を使用して前記第1のフレームに隣接するフレーム間の明度差を定義する請求項101に記載の方法。

## 【請求項 104】

スキップできる連続するフレーム数を第1の量に制限するステップをさらに含む請求項101に記載の方法。 10

## 【請求項 105】

前記第1の時間パラメータが第1のフレームレートに正規化される請求項101に記載の方法。

## 【請求項 106】

一連のフレーム内のスキップするビデオフレームを選択するように構成されているプロセッサ回路であって、

前記一連のフレームの一部分の中の各々のフレームを挟むフレーム間の時間パラメータで重み付けされた対応する差の絶対値の平均値を前記一連のフレームの少なくとも一部の中の前記各々のフレームについて計算するように構成されている第1の命令と、 20

前記時間パラメータで重み付けされた前記対応する差の絶対値の平均値に、少なくとも部分的に基づいて、スキップする第1のフレームを指定するように構成されている第2の命令とを有するプロセッサ回路。

## 【請求項 107】

前記第1のフレームがスキップされると仮定したときに目標フレームレートの条件が満たされるか否かを判別するように構成されている第3の命令をさらに有する請求項106に記載のプロセッサ回路。

## 【請求項 108】

スキップできる連続するフレーム数を第1の量に制限するように構成される第3の命令をさらに有する請求項106に記載のプロセッサ回路。 30

## 【請求項 109】

フレームレート制御装置であって、

目標フレームレートを受信するように構成されている第1の命令と、

一連のフレーム内の第1のフレームに対する第1のコスト関数値を計算するように構成されている第2の命令であって、前記第1のコスト関数値が、前記第1のフレームに隣接する前記一連のフレーム内のフレーム間の時間差及び明度差に関係する第1の時間パラメータに、少なくとも部分的に基づく前記第2の命令と、

前記一連のフレーム内の第2のフレームに対する第2のコスト関数値を計算するように構成されている第3の命令であって、前記第2のコスト関数値が、前記第2のフレームに隣接する前記一連のフレーム内のフレーム間の時間差及び明度差に関係する第2の時間パラメータに、少なくとも部分的に基づく第3の命令と、 40

前記第1のコスト関数値と前記第2のコスト関数値に少なくとも部分的に基づいて、前記第1のフレーム及び前記第2のフレームのうちのスキップする一方を選択するように構成されている第4の命令と、

前記第1のフレーム及び前記第2のフレームのうちの選択された一方がスキップされる場合に、前記目標フレームレートが満たされるか否かを判別するように構成されている第5の命令とを有するフレームレート制御装置。

## 【請求項 110】

前記フレームレート制御装置が集積回路である請求項109に記載のフレームレート制御装置。

## 【請求項 1 1 1】

平均差平均関数を使用して前記第 1 のフレームに隣接するフレーム間の明度差を定義する請求項 1 0 9 に記載のフレームレート制御装置。

## 【請求項 1 1 2】

複数のフレームパケットについて前方誤り訂正 ( F E C ) を実行する方法であって、第 1 のフレームに関する複数のフレームパケットに対応するパケットデータの選択された部分を連結するステップと、パケットデータの前記連結され選択された部分に対する前方誤り訂正ビットを生成するステップと、ユーザデータ識別子コードで識別された別々のパケット中で前方誤り訂正ビットを送信するステップとを含む方法。

10

## 【請求項 1 1 3】

前記別のパケット中での前記前方誤り訂正ビットの前記送信が M P E G - 4 準拠である請求項 1 1 2 に記載の方法。

## 【請求項 1 1 4】

前記別のパケットが前記複数のフレームパケットの後に送信される請求項 1 1 2 に記載の方法。

## 【請求項 1 1 5】

B C H コードを使用して前記前方誤り訂正ビットが生成される請求項 1 1 2 に記載の方法。

20

## 【請求項 1 1 6】

体系的コードを使用して前記前方誤り訂正ビットが生成される請求項 1 1 2 に記載の方法。

## 【請求項 1 1 7】

パケットデータの前記選択された部分が動きベクトルデータ及び D C T データを含む請求項 1 1 2 に記載の方法。

## 【請求項 1 1 8】

パケットデータの前記選択された部分がヘッダデータ、動きベクトルデータ、及び D C T データのみを含む請求項 1 1 2 に記載の方法。

## 【請求項 1 1 9】

パケットデータの前記選択された部分が再同期フィールドと動きマーカーとの間に置かれたパケットデータに対応する請求項 1 1 2 に記載の方法。

30

## 【請求項 1 2 0】

固定ビデオオブジェクトプレーン ( V O P ) の増分を使用することを指示するフラグをセットするステップと、対応する固定時間増分値を与えるステップとをさらに含む請求項 1 1 2 に記載の方法。

## 【請求項 1 2 1】

前記前方誤り訂正ビットが生成された前記第 1 のパケット内のある量のビットに関する前記複数のフレームパケットのうちの少なくとも第 1 のパケットに対する値を、前記別のパケット中で送信することを含む請求項 1 1 2 に記載の方法。

40

## 【請求項 1 2 2】

誤り訂正生成回路であって、対応する複数のフレームパケット中で送信されるパケットデータの選択された部分に対する前方誤り訂正データを生成するように構成されている、プロセッサ読み取り可能なメモリ内に格納される第 1 の命令と、前記複数のフレームパケットと別の第 1 のパケット内に前記前方誤り訂正データを格納するように構成されているプロセッサ読み取り可能なメモリに格納される第 2 の命令と、第 1 のデータ識別子コードで前記第 1 のパケットを識別するように構成されている、プロセッサ読み取り可能なメモリ内に格納される第 3 の命令とを有する誤り訂正生成回路。

## 【請求項 1 2 3】

50

前記第 1 の命令が前記前方誤り訂正データを生成する前に、パケットデータの選択した部分を連結するように構成されている第 4 の命令をさらに有する請求項 1 2 2 に記載の誤り訂正生成回路。

【請求項 1 2 4】

固定ビデオオブジェクトプレーン ( V O P ) の増分を使用することを指示するフラグをセットし、対応する固定時間増分値を与えるように構成されている第 4 の命令をさらに備える請求項 1 2 2 に記載の誤り訂正生成回路。

【請求項 1 2 5】

第 1 の一連のパケット内の全てのパケット中でヘッダ拡張コード ( H E C ) を供給するように構成されている第 4 の命令をさらに有する請求項 1 2 2 に記載の誤り訂正生成回路。

10

【請求項 1 2 6】

前記誤り訂正生成回路が集積回路である請求項 1 2 2 に記載の誤り訂正生成回路。

【請求項 1 2 7】

前記第 1 のパケットが M P E G - 4 準拠である請求項 1 2 2 に記載の誤り訂正生成回路。

【請求項 1 2 8】

B C H コードを使用して、前記前方誤り訂正データが生成される請求項 1 2 2 に記載の誤り訂正生成回路。

【請求項 1 2 9】

体系的コードを使用して、前記前方誤り訂正データが生成される請求項 1 2 2 に記載の誤り訂正生成回路。

20

【請求項 1 3 0】

パケットデータの前記選択された部分が動きベクトルデータ及び D C T データを含む請求項 1 2 2 に記載の誤り訂正生成回路。

【請求項 1 3 1】

パケットデータの前記選択された部分が、ヘッダデータ、動きベクトルデータ、及び D C T データのみを含む請求項 1 2 2 に記載の誤り訂正生成回路。

【請求項 1 3 2】

パケットデータの前記選択された部分が、再同期フィールドと動きマーカーとの間に置かれたパケットデータに対応する請求項 1 2 2 に記載の誤り訂正生成回路。

【請求項 1 3 3】

30

エンコーダ回路であって、

複数のフレームパケットから選択されたパケットデータの部分について前方誤り訂正データを生成する手段と、

前記複数のフレームパケットと別の第 1 のパケット内に前記前方誤り訂正データを格納する手段と、

第 1 のデータ識別子コードで前記第 1 のパケットを識別する手段とを備えるエンコーダ回路。

【請求項 1 3 4】

前記選択されたパケットデータの部分を連結する手段をさらに備える請求項 1 3 3 に記載のエンコーダ。

40

【請求項 1 3 5】

前方誤り訂正ビットが生成された前記第 1 のパケット内のある量のビットに関係する少なくとも第 1 の値を前記第 1 のパケット中で送信する手段をさらに備える請求項 1 3 3 に記載のエンコーダ。

【請求項 1 3 6】

適応型イントラリフレッシュを実行する方法であって、

パケット損失確率値を受信するステップと、

第 1 のフレーム内の第 1 のマクロブロックの動きベクトルを受信するステップと、

前のフレーム内の他の複数のマクロブロックの一部に、前記動きベクトルをマッピングするステップと、

50

前記マッピングに少なくとも部分的に基づいて、少なくとも第 1 の遷移係数値を計算するステップと、

少なくとも前記パケット損失確率値及び前記第 1 の遷移係数値に基づいて、前記第 1 のマクロブロックの推定インター歪み (inter distortion) 値を計算するステップと、

少なくとも前記パケット損失確率値に基づいて前記第 1 のマクロブロックの推定イントラ歪み (intra distortion) 値を計算するステップと、

前記第 1 のマクロブロックをイントラコーディングするために使用するビット量に対応する第 1 のビット量値を受信するステップと、

前記第 1 のマクロブロックをインターコーディングするために使用するビット量に対応する第 2 のビット量値を受信するステップと、

前記推定インター歪み値、前記推定イントラ歪み値、前記第 1 のビット量値、及び前記第 2 のビット量値に、少なくとも部分的に基づいて、前記第 1 のマクロブロックをイントラマクロブロック及びインターマクロブロックのうちの一つとして送信するステップとを含む方法。

10

【請求項 137】

量子化歪み値に一部基づいて前記第 1 のマクロブロックをイントラマクロブロック及びインターマクロブロックのうちの一つとして送信するステップをさらに含む請求項 136 に記載の方法。

【請求項 138】

前記推定インター歪み値と前記推定イントラ歪み値との差に基づいて第 1 の差の値を、前記第 1 のビット量値と前記第 2 のビット量値との差に基づいて第 2 の差の値を、決定するステップをさらに含む請求項 136 に記載の方法。

20

【請求項 139】

前記第 2 の差の値が正であること、及び前記第 1 の差の値を前記第 2 の差の値で除算して得られる値が負で、その絶対値が第 1 のしきい値よりも大きいことに、少なくとも部分的に応じて、前記第 1 のマクロブロックをイントラマクロブロックとして送信する請求項 137 に記載の方法。

【請求項 140】

前記第 2 の差の値が 0 であること、及び前記第 1 の差の値が負であることに、少なくとも部分的に応じて、前記第 1 のマクロブロックをイントラマクロブロックとして送信する請求項 137 に記載の方法。

30

【請求項 141】

前記第 2 の差の値が負であること、及び前記第 1 の差の値を前記第 2 の差の値で除算して得られる値が第 1 のしきい値よりも大きいことに、少なくとも部分的に応じて、前記第 1 のマクロブロックをイントラマクロブロックとして送信する請求項 137 に記載の方法。

【請求項 142】

前記第 1 のしきい値が重み係数である請求項 141 に記載の方法。

【請求項 143】

前記第 2 の差の値が 0 であること、及び前記第 1 の差の値が正であることに、少なくとも部分的に応じて、前記第 1 のマクロブロックをインターマクロブロックとして送信する請求項 137 に記載の方法。

40

【請求項 144】

前記第 2 の差の値が負であること、及び前記第 1 の差の値を前記第 2 の差の値で除算して得られた値が第 1 のしきい値よりも小さいことに、少なくとも部分的に応じて、前記第 1 のマクロブロックをイントラマクロブロックとして送信する請求項 137 に記載の方法。

【請求項 145】

前記第 2 の差の値が正であること、及び前記第 1 の差の値を前記第 2 の差の値で除算して得られた値が第 1 のしきい値よりも小さいことに、少なくとも部分的に応じて、前記第 1 のマクロブロックをイントラマクロブロックとして送信する請求項 137 に記載の方法。

【請求項 146】

50

前記推定イントラ歪み値がさらに初期誤りエネルギーに基づく請求項 1 3 6 に記載の方法。

【請求項 1 4 7】

前記推定イントラ歪み値が、前記パケット損失確率値で乗算され、平方された初期誤りエネルギーにさらに基づく請求項 1 3 6 に記載の方法。

【請求項 1 4 8】

前記推定インター歪み値が、再帰的に計算され、複数の前のフレームから伝搬される歪みを含む請求項 1 3 6 に記載の方法。

【請求項 1 4 9】

前記第 1 の遷移係数が、第 1 の半ピクセル水平垂直伝搬強度に対応する請求項 1 3 6 に記載の方法。 10

【請求項 1 5 0】

前記第 1 の遷移係数が第 1 の半ピクセル水平強度に対応する請求項 1 3 6 に記載の方法。

【請求項 1 5 1】

前記第 1 の遷移係数が第 1 の半ピクセル垂直伝搬強度に対応する請求項 1 3 6 に記載の方法。

【請求項 1 5 2】

前記第 1 のフレーム内でイントラコーディングされるマクロブロックの数を第 1 の量に制限するステップをさらに含む請求項 1 3 6 に記載の方法。

【請求項 1 5 3】 20

マクロブロックを選択的にイントラコーディングする方法であって、

パケット損失確率値を受信するステップと、

第 1 のフレーム内の第 1 のマクロブロックの動きベクトルを受信するステップと、

少なくとも部分的に前記動きベクトルに基づいて、前記第 1 のマクロブロックを予測する際に前のフレーム内でどの部分のマクロブロックを使用するかを決定するステップと、

前記第 1 のマクロブロックを予測する際に前記前のフレーム内でどの部分のマクロブロックを使用するかを決定することに、少なくとも部分的に基づいて、少なくとも第 1 の伝搬強度を計算するステップと、

少なくとも前記パケット損失確率値及び前記第 1 の伝搬強度値に基づいて前記第 1 のマクロブロックの推定インター歪み値を計算するステップと、 30

少なくとも前記パケット損失確率値に基づいて前記第 1 のマクロブロックの推定イントラ歪み値を計算するステップと、

前記第 1 のマクロブロックの量子化歪み値を計算するステップと、

前記第 1 のマクロブロックをイントラコーディングするために使用するビット量に対応する第 1 のビット量値を受信するステップと、

前記第 1 のマクロブロックをインターコーディングするために使用するビット量に対応する第 2 のビット量値を受信するステップと、

前記推定インター歪み値、前記推定イントラ歪み値、前記量子化歪み値、前記第 1 のビット量値、前記第 2 のビット量値に少なくとも部分的に基づいて、前記第 1 のマクロブロックを、送信のためにイントラマクロブロック及びインターマクロブロックのうちの一つとして供給するステップとを含む方法。 40

【請求項 1 5 4】

前記推定イントラ歪み値が、初期誤りエネルギーにさらに基づく請求項 1 5 3 に記載の方法。

【請求項 1 5 5】

前記推定インター歪み値が、再帰的に計算され、複数の前のフレームから伝搬される歪みを含む請求項 1 5 3 に記載の方法。

【請求項 1 5 6】

前記推定イントラ歪み値が、前記パケット損失確率値で乗算され、平方された初期誤りエネルギーにさらに基づく請求項 1 5 3 に記載の方法。 50

## 【請求項 157】

前記第1のマクロブロックが、さらに重み係数に基づいて、イントラマクロブロック及びインターマクロブロックのうちの一つとして供給するステップを含む請求項153に記載の方法。

## 【請求項 158】

前記重み係数が、少なくとも部分的に、対応する異なる量子化パラメータを使用して複数のイントラ歪み値を計算し、第1のビットレート条件を満たす最小の歪み値となる重み係数値を選択することによって、生成される請求項157に記載の方法。

## 【請求項 159】

第1のフレーム内の複数のマクロブロック内のマクロブロックを選択的にイントラコーディングする方法であって、

パケット損失確率値を受信するステップと、

複数のマクロブロック内の各々のマクロブロックの対応する動きベクトルを受信するステップと、

少なくとも部分的に前記対応する動きベクトルに基づき、各々の前記マクロブロックを予測する際に前のフレーム内のどの部分のマクロブロックを使用するかを決定するステップと、

各々の前記マクロブロックを予測する際に前記前のフレーム内のどの部分のマクロブロックを使用するかを決定したことに、少なくとも部分的に基づいて、各々の前記マクロブロックについて少なくとも第1の対応する伝搬強度値を計算するステップと、

少なくとも前記パケット損失確率値及び前記対応する第1の伝搬強度値に基づいて、各々の前記マクロブロックについて前記推定インター歪み値を計算するステップと、

少なくとも前記パケット損失確率値に基づいて、各々の前記マクロブロックについて推定イントラ歪み値を計算するステップと、

各々の前記マクロブロックについて、量子化歪み値を計算するステップと、

前記複数のマクロブロック内の各々のマクロブロックの前記推定インター歪み値、前記推定イントラ歪み値、及び前記量子化歪み値に、少なくとも部分的に基づいて、前記複数のマクロブロックのサブセットをイントラコーディングの対象として指定するステップとを含む方法。

## 【請求項 160】

前記サブセットが所定数のマクロブロックに制限される請求項159に記載の方法。

## 【請求項 161】

前記マクロブロック毎に、前記各マクロブロックをイントラコーディングするために使用されるビット量に対応する第1のビット量値を受信するステップと、

前記マクロブロック毎に、前記各マクロブロックをインターコーディングするために使用されるビット量に対応する第2のビット量値を受信するステップとをさらに含み、

前記サブセットが、前記第1のビット量値及び前記第2のビット量値に部分的に基づいて指定される請求項159に記載の方法。

## 【請求項 162】

マクロブロックの選択的にイントラコーディングするように構成された回路であって、

パケット損失確率値を受信するように構成されている第1の命令と、

第1のフレーム内の第1のマクロブロックの動きベクトルを受信するように構成されている第2の命令と、

少なくとも部分的に前記動きベクトルに基づき、前記第1のマクロブロックを予測する際に前のフレーム内でどの部分のマクロブロックを使用するかを決定するように構成されている第3の命令と、

前記第1のマクロブロックを予測する際に前記前フレーム内でどの部分のマクロブロックを使用するかを決定したことに、少なくとも部分的に基づいて、少なくとも第1の伝搬強度を計算するように構成されている第4の命令と、

少なくとも前記パケット損失確率値及び前記第1の伝搬強度値に基づいて、前記第1のマ



クロブロックの推定インター歪み値を計算するように構成されている第5の命令と、少なくとも前記パケット損失確率値に基づいて、前記第1のマクロブロックの推定イントラ歪み値を計算するように構成されている第6の命令と、前記推定歪み値及び前記推定歪み内値に少なくとも部分的に基づいて、前記第1のマクロブロックを、送信するためにイントラマクロブロック及びインターマクロブロックの何れかとして選択的に供給するように構成されている第7の命令とを有する回路。

【請求項163】

前記イントラ歪み値が、さらに初期誤りエネルギーに基づく請求項162に記載の回路。

【請求項164】

前記第1のマクロブロックの量子化歪み値を計算するようにさらに構成されている第8の命令を備え、

前記第7の命令がさらに、少なくとも部分的に前記量子化歪み値に基づいて、前記第1のマクロブロックを、送信するためにイントラマクロブロック及びインターマクロブロックの何れかとして選択的に供給するように構成されている請求項162に記載の回路。

【請求項165】

前記第5の命令が、前記推定インター歪み値を再帰的に計算するように構成され、それによって前記推定インター歪み値が複数の前のフレームから伝搬される歪みを含む請求項162に記載の回路。

【請求項166】

前記イントラ歪み値が、前記パケット損失確率値で乗算され、平方された初期誤りエネルギーにさらに基づく請求項162に記載の回路。

【請求項167】

前記第7の命令が、重み係数に少なくとも部分的に基づいて、前記第1のマクロブロックを、送信するためにイントラマクロブロック及びインターマクロブロックの何れかとして選択的に供給するようにさらに構成されている請求項162に記載の回路。

【発明の詳細な説明】

【技術分野】

【0001】

著作権

本出願文書の開示の一部に、著作権保護の対象となる資料が含まれている。著作権所有者は、特許商標庁の特許ファイル又は記録として、誰が特許文書又は特許開示のファクシミリ複製を行おうと異存はないが、それ以外では、いかなる形であれ全ての著作権を留保する。

【0002】

本発明は、ビデオ及びイメージのコード化に関するものであり、特に、誤り耐性のある圧縮方法でビデオイメージ情報をコード化するシステム及び方法に関する。

【背景技術】

【0003】

MPEGはMPEG (Moving Picture Experts Group) によって策定されたISO/IEC規格である。MPEG規格にはMPEG-1、MPEG-2、MPEG-4、及びMPEG-7などいくつかのバージョンがあり、イメージ及びオーディオ圧縮の特定の側面を標準化することを目的としたものである。H.261、H.262、H.263、H.263+、H.263++、H.26L、MPEG圧縮などの他のビデオ圧縮形式では、冗長データや関連性のないデータの排除を試みる。例えば、MPEGエンコーダでは、ビデオデータ全体を縮小するために、選択したフレームから得られる情報であって、他のフレームのために送信される必要のある情報を使用する。

【0004】

通常、ビデオフレームは、イントラフレーム、予測フレーム、及び双方向フレームの3つの方法のうちの一つで符号化される。さらに、得られるファイルサイズ又はビットレートを縮小するために、ビデオフレームをスキップすることもできる。イントラフレームには

、通常、そのフレームの完全なイメージデータが含まれ、したがって、他のフレームからのイメージデータに依存しない。イントラフレーム符号化では、圧縮は最小である。予測フレームは、一般に、デコーダが最新の先行するイントラフレーム又は予測フレームに基づいてフレームを表示できるだけの情報を含む。即ち、予測フレームは、前のフレームからイメージがどのように変化したかということに関係するデータ及び残余の誤り訂正データを含む。双方向フレームは、残余の誤り訂正データを含む前後のイントラフレーム及び/又は予測フレームからの情報から生成される。デコーダは、前後のフレームからのデータを使用し、補間により、それぞれのピクセルの位置と色を計算する。

**【0005】**

MPEG-4規格は、低ビットレートと高ビットレートの両方のアプリケーションで使用することを目的に策定された。例えば、MPEG-4は、インタラクティブビデオゲーム、テレビ会議、テレビ電話、インタラクティブ記録媒体、マルチメディアメール、無線マルチメディア、及び放送アプリケーションで使用できるように拡張されている。MPEG-4は、オブジェクトスケラビリティ、誤りに対する堅牢性の向上、圧縮率向上という特徴を持つ。

10

**【0006】**

有線/無線インターネットを介したマルチメディア通信に対する需要が増大し続けており、パケット損失の問題だけでなく帯域幅の変動の問題にも直面している。イメージフレーム同士は互いに依存しているため、圧縮されたビデオストリームは、損失パケットが少なくても影響を受ける。したがって、MPEG-4は、低ビットレート(<64kbs)のモバイル、無線アプリケーションなどの誤りの起こりやすいアプリケーション、及び誤りの起こりやすいATM(非同期転送モード)ネットワークアプリケーションで使用できるように特に強化されている。モバイル機器のオペレーションは、送信誤りの影響を受けやすい傾向があるが、それは、ビットレートを下げるためにデータ冗長性が低いこと、及び「雑音」の発生源が大きいことが多いからである。例えば、環境雑音により無線チャンネルが壊れることがあり、またモバイルアプリケーションの場合には、建物やその他の構造物が原因のマルチパスのフェージングとシャドウイングから生じるバースト雑音により無線チャンネルが壊れることがある。ATMネットワークアプリケーションに関しては、ネットワークの輻輳及びバッファオーバーフローによりセルが喪失することがある。

20

**【0007】**

MPEG-4では、旧バージョンのMPEGに比べて誤り耐性が高められており、そのような誤りの発生しやすいネットワーク上で、ビデオデータがより正常に送信される。例えば、MPEG-4規格で用意している誤り耐性手法の1つは、ビデオビットストリーム内で再同期マーカ(resync marker)を使用する。特に、MPEG-4では、固定時間間隔同期(fixed interval synchronization)を採用しており、ビデオオブジェクトプレーン(VOP)開始コードと再同期マーカ(即ち、ビデオパケットの開始)がビットストリーム内の正当な固定間隔位置にのみ出現するように規定している。これにより、開始コードエミュレーションに関係する問題を回避することができる。エンコーダによりビデオデータに挿入された再同期マーカを使用することにより、誤りの発生後失われた同期をデコーダによって回復することができる。

30

40

**【0008】**

MPEG-4規格で用意している誤り耐性手法の1つは、リバーシブル可変長コード(reversible variable length code)を使用する。このコードは、逆方向に読み込んだときでも復号化することができるため、デコーダは、新しく見つかった再同期マーカから、逆に戻ったデータ内の誤りが発生した地点までの破損していない情報を、使用することができる。MPEG-4で採用しているさらに別の誤り耐性方法は、動き情報及びテクスチャ情報の間に挿入されている第2の再同期マーカを使用して、テクスチャ情報から動き情報を分離するために使用されるデータ分割(data partitioning)がある。したがって、誤りが発生し、テクスチャ情報が復号化できないか失われている場合、デコーダは、動き情報を利用して前に復号化されたフレーム又はVOPを補正することによって、誤りを

50

隠すことができる動き又は。

【0009】

しかし、上記の拡張を利用したとしても、多くのMPEG-4エンコーダ及びデコーダは、携帯電話アプリケーションなど誤りの発生しやすいアプリケーションで多くの場合望まれているほど十分な誤り耐性を持たない。したがって、セルラーネットワークでMPEG-4準拠ビデオストリームを送信した場合、回復不可能なデータの破損が発生し、受信者に届くビデオデータの品質が著しく劣化することが多い。このようなビデオの劣化が生じると、誤りの起きやすいネットワークによるビデオ通信がユーザーの立場から望ましくないものとなり、都合が悪いことに、誤りの起きやすいネットワークでのビデオ送信の採用及び利用が妨げられる。

10

【0010】

さらに、目標のビットレートを得るために、従来のエンコーダでは、単純なスキップアルゴリズムに従って、複数のフレームを落とし、フレームレートを下げている。例えば、従来のエンコーダは、ビデオクリップ内の5つのフレームうちの4つを落として、ビデオクリップを毎秒30フレームのフレームレートから毎秒6フレームのフレームレートに変換する。しかし、この単純な方法のスキップでは、復号化したときに視覚的品質に著しい悪影響を及ぼすことが多い。

【発明の開示】

【課題を解決するための手段】

【0011】

本発明は、ビデオ符号化に関するものであり、特にビデオ情報を、圧縮し、及び/又は誤り耐性のある手段で符号化して送信するシステム及び方法に関する。本発明の実施の形態には、低ビットレート、高雑音環境であってもビデオ情報を送信できるという利点がある。例えば、本発明の実施の形態を利用すると、セルラーネットワークなどで正常にビデオ送信を実行することができる。

20

【0012】

本発明の一実施の形態では、シーンの変化があったときに、2つの連続するイントラコード化されたIフレームを自動的に挿入することにより、誤り耐性を高めることができる。シーンの変化があったとき、2つのイントラコード化されたフレームを挿入することにより、2つの連続するイントラコード化されたフレームのうちの一方が失われたり、復号化できない場合に、デコード側での誤り伝搬が少なくなる。

30

【0013】

例えば、一実施の形態では、符号化されたビデオ送信における誤り耐性を高める方法が提供され、この方法は、第1のフレームがシーンの変化に対応することを示す指示を受信するステップと、前記第1のフレームがシーンの変化に対応することを示す前記指示を受信したことに、少なくとも部分的に応じて、前記第1のフレームのイントラコーディングするステップと、連続する次のフレームのイントラコーディングを自動的に行うステップと、イントラコーディングされた前記第1のフレーム及びイントラコーディングされた前記連続する次のフレームを送信するステップとを含む。

【0014】

他の実施の形態では、プロセッサ読み取り可能なメモリに格納され、第1のフレームがイントラコーディングされることを示す指示を受信するように構成されている第1の命令と、プロセッサ読み取り可能なメモリに格納され、前記指示に少なくとも部分的に応じて、前記第1のフレームをイントラコーディングするように構成されている第2の命令と、プロセッサ読み取り可能なメモリに格納され、前記第1のフレームがイントラコーディングされることを示した結果、自動的に前記第1のフレームの直後のフレームをイントラコーディングするように構成されている第3の命令を含むビデオコーディング装置が提供される。

40

【0015】

別の実施の形態では、ビデオフレームを符号化する方法が提供され、この方法は、第1の

50

フレームをイントラコーディングする第1の命令を受信するステップと、前記第1の命令に対する応答として前記第1のフレームをイントラコーディングするステップと、前記第1のフレームをイントラコーディングする前記第1の命令の結果として第2のフレームをイントラコーディングするステップとを含む。

【0016】

さらに別の実施の形態では、第1のフレームをイントラコーディングする第1の命令を受信する手段と、前記第1の命令に対する応答として前記第1のフレームをイントラコーディングする手段と、前記第1のフレームをイントラコーディングする前記第1の命令の結果として第2のフレームをイントラコーディングする手段とを備えている符号化装置が提供される。

【0017】

拡張された適応型イントラリフレッシュ(AIR)プロセスは、予測フレーム内のマクロブロックの選択的且つ適応的な符号化を実行することによって効率のよい誤り耐性を実装する。一実施の形態では、イントラコーディングされるべきマクロブロックを決定するために、適応型動き領域プロセスが実行される。適応型動き変化検出により、動き領域に発生する誤りを含む、大きな伝搬誤りを効率よく低減することができる。インターコード歪み値及びイントラコード歪み値が、インターコードビット量及びイントラコード量と同様に計算される。計算して求めたインターコード歪み値とイントラコード歪み値の比較、及び各マクロブロックのインターコードビット量とイントラコードビット量との比較に基づいて、どの予測フレームマクロブロックがイントラコーディングされるべきであるかに関する決定を下す。

【0018】

本発明の一実施の形態は、適応型イントラリフレッシュを実行する方法であって、パケット損失確率値を受信するステップと、第1のフレーム内の第1のマクロブロックの動きベクトルを受信するステップと、前記動きベクトルを前のフレーム内の別の複数のマクロブロックの一部にマッピングするステップと、前記マッピングに基づいて少なくとも第1の遷移係数値を、少なくとも部分的に計算するステップと、少なくとも前記パケット損失確率値と前記第1の遷移係数値に基づいて前記第1のマクロブロックの推定インター歪み値(estimated inter distortion value)を計算するステップと、少なくとも前記パケット損失確率値に基づいて前記第1のマクロブロックの推定イントラ歪み値(estimated intra distortion value)を計算するステップと、前記第1のマクロブロックをイントラコーディングするために使用するビット量に対応する第1のビット量値を受信するステップと、前記第1のマクロブロックをインターコーディングするために使用するビット量に対応する第2のビット量値を受信するステップと、少なくとも部分的に、前記推定インター歪み値、前記推定イントラ歪み値、前記第1のビット量値、及び前記第2のビット量値に基づいて、前記第1のマクロブロックをイントラマクロブロック及びインターマクロブロックのうちの一つとして送信するステップとを含む。

【0019】

本発明の別の実施の形態は、マクロブロックを選択的にイントラコーディングする方法であって、パケット損失確率値を受信するステップと、第1のフレーム内の第1のマクロブロックの動きベクトルを受信するステップと、少なくとも部分的に動きベクトルに基づいて、前記第1のマクロブロックを予測する際に、前のフレーム内のどの部分のマクロブロックを使用するかを決定するステップと、前記第1のマクロブロックを予測する際に、前記前のフレーム内のどの部分のマクロブロックを使用するかを決定に少なくとも部分的に基づき少なくとも第1の伝搬強度値を計算するステップと、少なくとも前記パケット損失確率値及び前記第1の伝搬強度値に基づき前記第1のマクロブロックの推定インター歪み値を計算するステップと、少なくとも前記パケット損失確率値に基づき前記第1のマクロブロックの推定イントラ歪み値を計算するステップと、第1のマクロブロックの量子化歪み値を計算するステップと、前記第1のマクロブロックをイントラコーディングするために使用するビット量に対応する第1のビット量値を受信するステップと、前記第1のマク

10

20

30

40

50

ロブロックをインターコーディングするために使用するビット量に対応する第2のビット量値を受信するステップと、前記推定インター歪み値、前駆推定イントラ歪み値、前記量子化歪み値、前記第1のビット量値、前記第2のビット量値に少なくとも部分的に基づいて、第1のマクロブロックを、送信するためにイントラマクロブロック及びインターマクロブロックのうちの一つとして提供するステップとを含む。

**【0020】**

本発明のさらに別の実施の形態は、第1のフレーム内の複数のマクロブロック内のマクロブロックを選択的にイントラコーディングする方法であって、パケット損失確率値を受信するステップと、複数のマクロブロック内の各マクロブロックの対応する動きベクトルを受信するステップと、少なくとも部分的に対応する動き前記動きベクトルに基づき、前記各マクロブロックを予測する際に前のフレーム内のどの部分のマクロブロックを使用するかを決定するステップと、前記各マクロブロックを予測する際に前のフレーム内のどの部分のマクロブロックを使用するかを決定することに少なくとも部分的に基づいて、少なくとも第1の対応する伝搬強度値を前記マクロブロックのそれぞれについて計算するステップと、少なくとも前記パケット損失確率値及び対応する少なくとも前記第1の伝搬強度値に基づき、推定インター歪み値を前記マクロブロックのそれぞれについて計算するステップと、少なくとも前記パケット損失確率値に基づき推定イントラ歪み値を前記マクロブロックのそれぞれについて計算するステップと、量子化歪み値を前記マクロブロックのそれぞれについて計算するステップと、少なくとも部分的に複数のマクロブロック内の各マクロブロックの前記推定インター歪み値、前記推定イントラ歪み値、及び前記量子化歪み値に基づき、複数のマクロブロックのサブセットをイントラコーディングされるべき対象として指定するステップとを含む。

10

20

**【0021】**

本発明の一実施の形態は、マクロブロックを選択的にイントラコーディングするように構成された回路であって、パケット損失確率値を受信するように構成されている第1の命令と、第1のフレーム内の第1のマクロブロックの動きベクトルを受信するように構成されている第2の命令と、少なくとも部分的に動きベクトルに基づき、前記第1のマクロブロックを予測する際に、前のフレーム内のどの部分のマクロブロックを使用するかを決定するように構成されている第3の命令と、前記第1のマクロブロックを予測する際に、前記前のフレーム内でどの部分のマクロブロックを使用するかを決定することに少なくとも部分的に基づいて少なくとも第1の伝搬強度値を計算するように構成された第4の命令と、少なくとも前記パケット損失確率値及び前記第1の伝搬強度値に基づいて前記第1のマクロブロックの推定インター歪み値を計算するように構成されている第5の命令と、少なくとも前記パケット損失確率値に基づいて前記第1のマクロブロックの推定イントラ歪み値を計算するように構成されている第6の命令と、前記推定インター歪み値及び前記推定イントラ歪み値に少なくとも部分的に基づいてイントラマクロブロック及びインターマクロブロックのうちの一つとして、前記第1のマクロブロックを、送信するために選択的に提供するように構成されている第7の命令とを備えている。

30

**【0022】**

本発明の実施形態は、定義済みのコスト関数に基づいて符号化プロセスにおいてフレームの適応型スキップを実行する。特にシーン変化領域内の視覚的品質は向上しており、しかもシーンは効率よく符合化されている。スキッププロセスの一実施の形態は、注目しているフレームに隣接するフレーム間の差の絶対値の平均を計算し、その計算に対して時間パラメータの重み付けを行うことにより、フレーム列からフレームを選択的に落とすという点で反復的であり、シーンに対する影響が最小であるフレームが落とされる。この手順は、望みのビットレート及びフレームサイズに関係する目標のフレームレートが得られるまで繰り返し実行される。差の絶対値の平均の和(SMAD)又は二乗平均平方根の和(SRMS)の値などの他の測定方法を、差の絶対値の平均の方法の代わりに又はそれに加えて使用することができる。

40

**【0023】**

50

本発明の一実施の形態は、符号化プロセスの実行中にスキップするビデオシーケンスフレームを選択する方法であって、一連のフレームを受信するステップと、前記一連のフレームの少なくとも一部分の中のフレーム毎に、時間パラメータによって重み付けされた、前記フレームを挟むフレーム間の対応する差の絶対値の平均を求めるステップと、フレーム毎に計算で求めた時間パラメータで重み付けされた対応する前記差の絶対値の平均に少なくとも部分的に基づいて、一連のフレーム内の第1のフレームをスキップすることを含む。

**【0024】**

本発明の別の実施の形態は、ビデオシーケンス内のスキップするフレームを決定する方法であって、第1のフレームレート目標を受信するステップと、ビデオシーケンスシーン内の先頭フレームと末尾フレームとの間にある、スキップされるべき第1の潜在的フレームを選択するステップと、前記第1の潜在的フレームをスキップしたことによって生じる視覚的影響に関係する、スキップされるべき前記第1の潜在的フレームの第1のコスト値を計算するステップと、前記第1のコスト値をコンピュータ読み取り可能なメモリ内に格納するステップと、前記先頭フレームと前記末尾フレームとの間にある、スキップされるべき第2の潜在的フレームを選択するステップと、スキップされるべき前記第2の潜在的フレームの第2のコスト値を計算するステップと、前記第2のコスト値をコンピュータ読み取り可能なメモリ内に格納するステップと、前記先頭フレームと前記末尾フレームとの間にある、スキップされるべき第3の潜在的フレームを選択するステップと、スキップされるべき前記第3の潜在的フレームの第3のコスト値を計算するステップと、前記第1のコスト値、前記第2のコスト値、前記第3のコスト値のうちのどれがコスト最小であるかを決定するステップと、最小コスト値に対応する前記第1、第2、及び第3の潜在的フレームの中の1つをスキップするステップと、前記第1のフレームレート目標が満たされているかを判別するステップと、前記第1のフレームレート目標が満たされていないと判別したことに、少なくとも部分的に応じて、前記第1、第2、及び第3の潜在的フレームから残りのフレームと関連するコスト値を再計算し、再計算したコスト値が最小である残りのフレームのうちの1つをスキップするステップとを含む。

**【0025】**

本発明のさらなる別の実施の形態は、符号化プロセスの実行中にスキップするイメージフレームを選択する方法であって、第1のフレームを含む一連のフレームを受信するステップと、前記第1のフレームに関して、前記第1のフレームに隣接するフレーム間の時間差及び明度差に関係する第1の時間パラメータを含む第1のコスト関数値を計算するステップと、第2のフレームに関して、該第2のフレームに隣接するフレーム間の時間差及び明度差に関係する第2の時間パラメータを含む第2のコスト関数値を計算するステップと、前記第1のコスト関数値及び前記第2のコスト関数値に少なくとも部分的に基づいて前記第1のフレーム及び前記第2のフレームの内の1つをスキップすることを選択するステップを含む。

**【0026】**

本発明のさらなる別の実施の形態は、一連のフレーム内のスキップするビデオフレームを選択するように構成されているプロセッサ回路であって、前記一連のフレームの少なくとも一部分の中のそれぞれのフレームについて、時間パラメータで重み付けされた、前記一連のフレームの一部分の中のそれぞれのフレームを挟むフレーム間の、対応する差の絶対値の平均を計算するように構成されている第1の命令と、時間パラメータで重み付けされた前記対応する差の絶対値の平均に少なくとも部分的に基づいてスキップされるべき第1のフレームを指定するように構成されている第2の命令とを備えている。

**【0027】**

本発明の一実施の形態は、フレームレート制御装置であって、目標フレームレートを受信するように構成されている第1の命令と、一連のフレーム内の第1のフレームに関して、前記第1のフレームに隣接する前記一連のフレーム内のフレーム間の時間差及び明度差に

10

20

30

40

50

関係する第1の時間パラメータに少なくとも部分的に基づき第1のコスト関数値を計算するように構成されている第2の命令と、前記一連のフレーム内の第2のフレームに関して、前記第2のフレームに隣接する前記一連のフレーム内のフレーム間の時間差及び明度差に関係する第2の時間パラメータに少なくとも部分的に基づき第2のコスト関数値を計算するように構成されている第3の命令と、前記第1のコスト関数値及び前記第2のコスト関数値に少なくとも部分的に基づいて、前記第1のフレーム及び前記第2のフレームのうち、スキップする方を選択するように構成されている第4の命令と、前記第1のフレーム及び前記第2のフレームのうちのスキップするように選択された一方に、前記目標フレームレートが整合するか否かを判別するように構成されている第5の命令を有する装置である。

10

## 【0028】

一実施の形態では、前方誤り訂正(FEC: forward error correction)情報を使用して誤り耐性を高める。FECコーディングは、リアルタイムで、動きベクトル、DC係数、及びヘッダ情報などの重要なデータに効率良くかつ選択的に適用され、重要でない又はあまり重要でないデータに対してはFECビットを生成しない。この選択された重要データは、パケット再同期フィールドと動きマーカーとの間に配置され得る。特に、指定されたフレーム又はVOPについては、FECコーディングをターゲットとする選択されたパケットビットが連結されて1つになり、その連結されたビットに対してFECコードビットが生成される。オプションとして、その結果得られるFECビットは通常のフレーム又はVOPパケットの後の追加パケット内に配置され、MP EG互換性が保証される。

20

## 【0029】

本発明の一実施の形態は、複数のフレームパケットについて前方誤り訂正(FEC)を実行する方法であって、第1のフレームに対する複数のフレームパケットに関して、パケットデータの選択された部分を連結するステップと、連結された前記パケットデータの選択された部分に対する前方誤り訂正ビットを生成するステップと、MP EG規格委員会などにより将来的に割り当てられる他の一意的な識別子コードを含む、ユーザーデータ識別子コードなどで識別された別々のパケット中で前記前方誤り訂正ビットを送信するステップを含む。

## 【0030】

本発明の別の実施の形態は、誤り訂正生成回路であって、対応する複数のフレームパケット中で送信される、パケットデータの選択された部分に対する前方誤り訂正データを生成するように構成されている、プロセッサ読み取り可能なメモリ内に格納される第1の命令と、複数の前記フレームパケットと別の第1のパケット内に前記前方誤り訂正データを格納するように構成されている、プロセッサ読み取り可能なメモリ内に格納される第2の命令と、第1のデータ識別子コードで前記第1のパケットを識別するように構成されている、プロセッサ読み取り可能なメモリ内に格納される第3の命令とを有している誤り。

30

## 【0031】

本発明のさらなる別の実施の形態は、エンコーダ回路であって、複数のフレームパケットから選択されたデータパケットの一部について前方誤り訂正データを生成する手段と、複数の前記フレームパケットと別の第1のパケット内に前記前方誤り訂正データを格納する手段と、第1のデータ識別子コードで前記第1のパケットを識別する手段とを備えている。

40

## 【0032】

さらに、本発明の実施の形態では、ヘッダ拡張コード(HEC: Heddrrer Extention Code)を、従来のエンコーダのようにVOPヘッダの後の第1のビデオパケット上だけでなく、一連のビデオパケット又は全てのビデオパケット内で使用することができる。この方法だと、パケットが失われたり破損したりしても、後続のパケットを復号化して使用できるので都合がよい。さらに、多くの従来のデコーダでも、HECの使用頻度が高い場合に対応することができる。

## 【0033】

50

オプションとして、Video-Object-Layer (VOL) ヘッダは、固定のビデオオブジェクトプレーン (VOP: Fixed Video Object Plane) 増分 (increment) が使用されるべきであることを示すようにセットされ、その後固定時間増分の値 (fixed time increment value) が続くフラグを持つ。このため、デコーダが欠損フレーム、即ち、圧縮を高めるためにエンコーダによってスキップされたフレーム又は送信中に失われたフレームを検出しやすくなる。

**【0034】**

本発明の実施の形態では、シーンレベル、フレームレベル、及び/又はマクロブロックレベルでビット割り当てを行う。ビット割り当てモジュールは、固定のビット割当値 (bit budget) の適切な分配を決定する。一実施の形態では、イントラコーディングされたフレーム数及びインターコーディングされたフレーム数に基づき重みがシーンに割り当てられる。イントラコーディングされたフレームは、インターコーディングされたフレームよりも重い重み付けがなされ、イントラフレームを符号化するために必要なビットをより多く占有する。ビット割り当てモジュールは、現在のビット使用度と目標のビット使用度とを比較してシーン内の固定のビット割当値を配分し、その比較に基づいて、現在のフレームに対する量子化パラメータ又はステップサイズを調整する。

10

**【0035】**

さらにコーディングの効率を高めるために、符号量子化ステップ又は量子化パラメータ (QP) を動的に調整する。これにより、短いシーン又はビデオオブジェクトプレーンのグループ (GOV) の場合でも、望みのビット割当値に高速に収束させることができる。さらに、各シーン又はGOVを符号化した後、前のシーン又はGOVの実際のビット使用度に基づいてIフレームに対するQPを動的に調整する。このIフレームのQP適応は、シーンシーケンス中に高い動きセグメントがあるときにビット割当値を達成するか、又はビット割当値達成に近づけるのに非常に役立つことがある。

20

**【0036】**

本発明の一実施の形態は、ビデオシーケンスとともに使用するビット割り当て方法であって、少なくとも第1のクリップに対するクリップビット割当値を受信するステップと、前記第1のクリップに対するシーン量を決定するステップと、前記第1のクリップの第1のシーン内の予測フレームの量を決定するステップと、前記第1のクリップに対するシーンの量、前記第1のクリップビット割当値、前記第1のシーン内のイントラコーディングフレームの量、及び前記第1のシーン内の予測フレームの量に少なくとも部分的に基づいて前記第1のシーンのビット割当値を計算するステップとを含む。

30

**【0037】**

本発明の別の実施の形態は、ビデオクリップシーンとともに使用するビット割り当て方法であって、第1のシーンに対するビット割当値を計算するステップと、前記第1のシーン内の少なくとも第1のフレームを含む対応するフレームに対するビット割当値を計算するステップと、前記第1のフレーム内のマクロブロックに対応するビット割当値を計算するステップを含む。

**【0038】**

本発明のさらなる別の実施の形態は、ビットを割り当てる装置であって、第1のシーン内に含まれるイントラコーディングフレームの数、前記第1のシーンに対する複雑度の判別、及び前記第1のシーン内に含まれるインターコーディングフレームの数に少なくとも部分的に基づいて、前記第1のシーンに対する第1のビット割当値を割り当てるように構成されている第1の命令と、現在のビット使用度及び目標のビット使用度に少なくとも部分的に基づいて前記第1のシーン内のフレームにビット割当値を割り当てるように構成されている第2の命令とを含む。

40

**【0039】**

本発明のさらなる別の実施の形態は、ビット割当を設定する (bit budgeting) 装置であって、少なくとも第1のビデオシーケンスに対するビット割当値を受信するように構成されている第1の命令と、前記第1のビデオシーケンスに対するシーンの量を決定するよう

50



に構成されている第2の命令と、前記第1のビデオシーケンスの第1のシーン内の予測フレーム数を決定するように構成されている第3の命令と、前記第1のクリップのシーン量、前記第1のクリップビット割当値、前記第1のシーン内のイントラコーディングフレームの量、及び前記第1のシーン内の予測フレームの量に少なくとも部分的に基づいて、前記第1のシーンに対するビット割当値を決定するように構成されている第4の命令とを含む。

【0040】

本発明の実施の形態は、シーンの変化を特定するための正確なシステム及び方法を提供する。誤り耐性を高めつつ、一般にインターコーディングよりも多いビット数を利用するシーン変化フレームがイントラコーディングされるため、シーン変化フレームを正確に判別することが、誤り耐性を効率よく実現するうえで重要である。一実施の形態では、第1の二乗平均平方根(RMS)値は、第2のフレームに対して第1のフレームについて計算される。二階時間微分RMS値は、第2のフレームに対する第1のフレームおよび第3のフレームに対する第2のフレームに関して計算される。二階時間微分RMS値に少なくとも部分的に基づき、第2のフレームはシーン変化フレームとして指定される。フレームは、シーン変化検出に基づいて、適応的にグループ化される。後述するように、シーン変化を検出するために、差の絶対値の平均(MAD)に基づく、及び/又はRMS極大値を求める、及び/又はMAD極大値を求める別の基準を使用することができる。

10

【0041】

本発明の一実施の形態は、複数のフレームを持つデジタルビデオシーケンス内のシーン変化を検出する方法であって、第2のフレームに対する第1のフレームおよび第3のフレームに対する第2のフレームに関して第1の二乗平均平方根(RMS)値を計算するステップと、前記第2のフレームに対する前記第1のフレームに関して第1の差の絶対値の平均(MAD)値を計算するステップと、前記第1のRMS値が第1の基準を満たしているか否かを判別するステップと、前記第1のMAD値が第2の基準を満たしているか否かを判別するステップと、前記第1のRMS値が前記第1の基準を満たし、且つ前記第1のMAD値が前記第2の基準を満たしていると判断したことに、少なくとも部分的に応じて、前記第2のフレームをシーン変化フレームとして指定するステップとを含む。

20

【0042】

本発明の別の実施の形態は、デジタルビデオシーケンス内のシーン変化を検出する方法であって、第2のフレームに対する第1のフレーム及び第3のフレームに対する第2のフレームに関して二階時間微分RMS値を計算するステップと、前記二階微分値に少なくとも部分的に基づき、前記第2のフレームがシーン変化フレームであると判断するステップを含む。

30

【0043】

本発明の別の実施の形態は、ビデオシーケンス内のシーン変化を識別する装置であって、プロセッサ読み取り可能なメモリ内に格納され、ビデオシーケンスの第2の部分に対する前記ビデオシーケンスの第1の部分に関して第1の二乗平均平方根(RMS)値を計算するように構成されている第1の命令と、プロセッサ読み取り可能なメモリに格納され、二階時間微分RMS値を計算するように構成されている第2の命令と、前記二階微分RMS値に少なくとも部分的に基づいて前記ビデオシーケンスの前記第2の部分がイントラコーディングされるように構成されている第3の命令とを含む。

40

【0044】

本発明の一実施の形態は、ビデオシーケンスのどの部分がイントラコーディングされるべきかを決定する方法であって、ビデオシーケンスの第1の部分に対する第1の二乗平均平方根(RMS)値を計算するステップと、前記ビデオシーケンスの前記第1の部分に対する第1の差の絶対値の平均(MAD)値を計算するステップと、前記第1のRMS値が第1の基準を満たしているか否かを判別するステップと、前記第1のMAD値が第2の基準を満たしているか否かを判別するステップと、前記第1のRMS値が第3の基準を満たしているか否かを判別するステップと、前記第1、第2、及び第3の基準のうちの少なくとも

50

も2つが満たされていることに、少なくとも部分的に応じて、イントラコーディング処理を実行させるステップとを含む。

【0045】

本発明の別の実施の形態は、シーン変化検出装置であって、第1のフレーム情報入力及びRMS出力を備え、前記第1のフレーム情報入力を受信したフレーム情報に基づいて、少なくとも2つのフレーム間の2乗平均平方根差情報に対応する値を前記RMS出力から出力するように構成されているRMS回路と、第2のフレーム情報入力及びMAD出力を備え、前記第2のフレーム情報入力を受信したフレーム情報に基づいて、少なくとも2つのフレーム間の差の絶対値の平均の情報に対応する値を前記MAD出力から出力するように構成されているMAD回路と、前記RMS出力及び前記MAD出力に結合され、二乗平均平方根差情報に対応する値及び差の絶対値の平均の情報に対応する値に少なくとも部分的に基づいて、シーン変化フレームを検出し、シーン変化指定を行うように構成されているエバリュエータ回路とを備えている。

10

【発明を実施するための最良の形態】

【0046】

本発明の好ましい実施の形態について、以下に図面を参照して説明する。これらの図面及び関連する説明は、本発明の実施例を説明するために用意したものであり、本発明の範囲を制限するものではない。

【0047】

本発明は、ビデオ符号化に関するものであり、特にビデオ情報を圧縮し、かつ/又は誤り耐性のある方法で符号化して送信するシステム及び方法に関する。従って、本発明の実施の形態には、低ビットレート、雑音、誤りを起こしやすい環境であってもビデオ情報を送信できるという利点がある。本発明の実施の形態は、例えば、MPEG-4規格、MPEG-1、MPEG-2、H.261、H.262、H.263、H.263+、H.263++、及びH.26L、及びさらに今後策定されるであろうビデオ規格など、さまざまなビデオ圧縮規格とともに使用することができる。MPEG-4規格の態様は、“Coding of Audio-Visual Objects: Systems” 14496-1、ISO/IEC JTC1/SC29/WG11 N2501、November 1998、及び“Coding of Audio-Visual Objects: Visual” 14496-2、ISO/IEC JTC1/SC29/WG11 N2502、November 1998で定義されており、またMPEG-4ビデオ検証モデルは、“MPEG-4 Video Verification Model 17.0” ISO/IEC JTC1/SC29/WG11 N3515、Beijing、China、July 2000で定義されている。

20

30

【0048】

図1Aは、本発明の一実施の形態に係るビデオ配給システムを実装するためのネットワーク接続されたシステムを示している。符号化コンピュータ102が、ビデオ信号を受信し、この信号は比較的コンパクトで堅牢な形式に符号化される。符号化コンピュータ102として、ソフトウェアを実行する汎用コンピュータを含む種々のタイプのマシン、及び専用ハードウェアを使用できる。符号化コンピュータ102は、衛星放送受信機104、ビデオカメラ106、テレビ会議端末108などを介して種々のソースからビデオシーケンスを受信することができる。ビデオカメラ106としては、ビデオカメラレコーダ、Webカメラ、無線デバイスに内蔵されたカメラなど、種々のカメラを使用できる。ビデオシーケンスはさらに、データストア110に格納することもできる。データストア110は、符号化コンピュータ102に内蔵のものでも、また外付けのものでもよい。データストア110は、テープ、ハードディスク、光ディスクなどのデバイスを備えることができる。当業者であれば、図1Aに示されているデータストア110などのデータストアは、符号化されていないビデオ、符号化されたビデオ、又はその両方を格納できることを理解できるであろう。一実施の形態では、符号化コンピュータ102は、データストア110などのデータストアから符号化されていないビデオを取り出して、その符号化されていないビデオを符号化し、符号化されたビデオをデータストアに格納する。このデータストアは同じデータストアであっても別のデータストアであってもよい。ビデオのソースとして、最初にフィルム形式で撮影されたソースを使用できることが理解されるであろう。

40

50

## 【0049】

符号化コンピュータ102は、符号化されたビデオを受信装置に配給し、受信装置が符号化されたビデオを復号化する。受信装置として、ビデオを表示することができる種々の装置を使用できる。例えば、ネットワーク接続されたシステムの図に示されている受信装置として、携帯電話112、パーソナルデジタルアシスタント(PDA)114、ラップトップコンピュータ116、及びデスクトップコンピュータ118がある。受信装置は、通信ネットワーク120を通じて符号化コンピュータ102と通信できる。この通信ネットワークとして、無線通信ネットワークを含む種々の通信ネットワークを使用できる。当業者であれば、携帯電話112などの受信装置もビデオ信号を符号化コンピュータ102に送信するために使用できることを理解できるであろう。

10

## 【0050】

符号化コンピュータ102、受信装置又はデコーダとしては、種々のコンピュータを使用できる。例えば、符号化コンピュータ102は、パーソナルコンピュータ、ワークステーション、サーバー、クライアント、ミニコンピュータ、メインフレームコンピュータ、ラップトップコンピュータ、個別コンピュータのネットワーク、モバイルコンピュータ、パームトップコンピュータ、ハンドヘルドコンピュータ、テレビ用セットトップボックス、インタラクティブテレビ、インタラクティブキオスク、パーソナルデジタルアシスタント、インタラクティブ無線通信デバイス、モバイルブラウザ、Web対応携帯電話、パーソナルデジタルアシスタント(PDA)、又はそれらの組み合わせなどの端末装置をはじめとするマイクロプロセッサ又はプロセッサ(以下、プロセッサと呼ぶ)制御装置とすることができる。例えば、エンコーダコンピュータをビデオカメラ106、携帯電話112、PDA 114、ラップトップコンピュータ116、及び/又はデスクトップコンピュータ118に組み込むこともできる。コンピュータ102はさらに、キーボード、マウス、トラックボール、タッチパッド、又はタッチスクリーンなどの入力装置、及びコンピュータ画面、プリンタ、スピーカーなどの出力装置、又は既存又は今後開発されるその他の入力装置を備えることができる。

20

## 【0051】

符号化コンピュータ102、ならびに、デコーダコンピュータとして、ユニプロセッサ又はマルチプロセッサマシンを使用できる。さらに、エンコーダ及びデコーダコンピュータは、ランダムアクセスメモリ(RAM)、電氣的消去可能プログラム可能読み出し専用メモリ(EEPROM)、マスク読み出し専用メモリ、1回だけプログラム可能なメモリ、ハードディスク、フレキシブルディスク、レーザーディスクプレーヤー、デジタルビデオ装置、コンパクトディスクROM、DVD-ROM、その他の光媒体、ビデオテープ、オーディオテープ、磁気記録トラック、電子ネットワーク、及び例えば、プログラム及びデータなどの電子的内容を送信又は格納するためのその他の手法などのアドレス指定可能なストレージ媒体又はコンピュータアクセス可能な媒体を備えることができる。一実施の形態では、符号化及び復号化コンピュータは、ネットワークインターフェイスカード、モデム、赤外線(IR)ポート、無線ネットワークインターフェイス、又はネットワークに接続するのに適しているその他のネットワーク接続装置などのネットワーク通信装置を備える。さらに、コンピュータはLinux、Unix、Microsoft(登録商標) Windows(登録商標)3.1、Microsoft(登録商標) Windows(登録商標)95、Microsoft(登録商標) Windows(登録商標)98、Microsoft(登録商標) Windows(登録商標)NT、Microsoft(登録商標) Windows(登録商標)2000、Microsoft(登録商標) Windows(登録商標)Me、Microsoft(登録商標) Windows(登録商標)XP、Apples(登録商標)MacOS(登録商標)、IBM(登録商標)OS/2(登録商標)、Microsoft(登録商標) Windows(登録商標)CE、又はPalm OS(登録商標)などの適切なオペレーティングシステムを実行する。従来のように、適切なオペレーティングシステムは、無線ネットワークを含むネットワーク上で受け渡される全ての着信及び送信メッセージトラフィックを処理す

30

40

50

る、通信プロトコル実装を備えると都合がよい。他の実施の形態では、オペレーティングシステムはコンピュータの種類によって異なることもあるが、オペレーティングシステムは、ネットワークとの通信リンクを確立するために必要な適切な通信プロトコルを提供し続ける。

#### 【0052】

図1Bは、本発明の一実施の形態に係る符号化システム100Bの一例を示す。本明細書で使用している、符号化システムという用語には、1つ又は複数のエンコーダが含まれる。符号化システム100Bは、例えば、本明細書で説明しているように動作するプロセッサ、プログラムロジック、又はデータ及び命令を表すその他の基板構成のうちの1つ又は複数を備える。別の実施の形態では、符号化システム100Bは、コントローラ回路、集積回路、ゲートアレイ、特定用途向け集積回路、プロセッサ回路、プロセッサ、汎用シングルチップ又はマルチチップマイクロプロセッサ、デジタルシグナルプロセッサ、組み込み型マイクロプロセッサ、マイクロコントローラなどを備え、コンピュータ読み取り可能なメモリに格納されている命令及びデータを含む、ソフトウェアコードを実行することができる。例えば、これには限定されないが、符号化システム100Bは、1つ又は複数のリード有り、リード無し、又はボールグリッドアレイ半導体パッケージに、1つ又は複数の回路基板上に、及び/又は1つ又は複数のハイブリットパッケージを使用して収容されることができる。符号化システム100Bの全部又は一部を、デスクトップコンピュータなどの固定端末、又は携帯電話、携帯コンピュータ、パーソナルデジタルアシスタント、ビデオカメラなどの携帯端末に組み込むことができる。符号化システム100Bは、実施例では、符号化コンピュータ102に相当する。さらに、例えば、本発明による符号化システムを使用して、テレビ会議を実施し、動画又はその他のイメージを格納又は送信するのを補助することなどが可能である。

#### 【0053】

符号化システム100Bは、ビデオ情報を符号化し、圧縮した後、デコーダに送信する。符号化システム100Bは、前処理モジュール又は回路102B、ビット割り当てモジュール又は回路104B、及びエンコーダモジュール又は回路106Bを備えている。ビデオシーケンスアナライザを含む、前処理モジュール又は回路102Bを使用して、シーン変化の発生を検出し、指定されたフレーム、VOP又はピクチャを、どのように符号化するかを決定する。

#### 【0054】

ビデオオブジェクトレイヤには、MPEG-4ではビデオオブジェクトプレーン(VOP)と呼ぶ、異なる時間間隔での任意の形状の2D表現のシーケンスが含まれる。VOP領域のそれぞれは、非矩形領域であり、シーン内の物体など、注目している特定のイメージ又はビデオコンテンツに対応することができる。ビデオオブジェクトプレーン(VOP)は、16×16のサイズのマクロブロックに分割される。マクロブロックは、8×8サイズの6つのブロック単位で符号化され、そのうち4つのブロックは明度用、2つのブロックは色度用である。任意形状のVOPからマクロブロック構造を得るためには、VOPの境界ボックス(bounding box)を計算し、マクロブロックサイズの倍数単位で拡張する。

#### 【0055】

ただし、最新のアプリケーションについては、また特に、いわゆる「単純プロファイル」を使用する無線アプリケーションについては、一般に、フレーム毎に1つのVOPしかなく、これは矩形VOPである。わかりやすくするために、フレームという用語は、本明細書で使用しているように、MPEG-4 VOPなどのVOP、又はピクチャを含むこともできる。同様に、VOPという用語は、本明細書で使用しているように、フレームを意味する場合もある。MPEG-4では、VOPは、ビデオオブジェクトプレーンのグループ(GOV)として構造化することができる。MPEG-2の用語を使うと、フレーム又はピクチャを、ピクチャのグループ(GOP)に配列できるということである。わかりやすくするために、本明細書で使用している「シーン」という用語は、GOV又はGOPをも意味し、またその逆にGOV又はGOPはシーンを意味する。

## 【 0 0 5 6 】

フレーム又はビデオオブジェクトを、イントラコーディングフレーム（「Iフレーム」又は「I - V O P」）、予測フレーム（「Pフレーム」又は「P - V O P」）、又は双方向フレーム（「Bフレーム」又は「B - V O P」）として符号化することができる。M P E G - 1はさらに、Dフレームにも対応している。Dフレームとは、動きベクトルを持たないフレームのことで、0ベクトルが仮定され、テクスチャD C Tデータを持つ。空間的冗長性を活かすため、離散コサイン変換（D C T）を符号化されたフレームに対して実行し、その結果得られた係数を量子化する。

## 【 0 0 5 7 】

M P E G - 4の単純プロファイルでは、Bフレーム又はB - V O Pをサポートしていない。しかし、単純プロファイルではフレームのスキップ操作をサポートしている。得られるファイルサイズ又はビットレートを縮小するために、ビデオフレームをスキップすることができる。M P E G - 4の単純プロファイルではBフレーム又はDフレームをサポートしていないため、以下の説明ではこのようなフレームを対象としない。しかしながら、本発明の実施の形態は、他のプロファイル及びその他の規格に従って、Bフレーム及びDフレームとともに使用することができる。

## 【 0 0 5 8 】

フレームという用語は、インタレースフレーム又はノンインタレースフレーム、即ちプログレッシブフレームに対応する。インタレースフレームでは、各フレームは2つの別々のフィールドで構成され、それらはフレームを形成するために一緒に組み合わせられる。このようなインタレースは、ノンインタレース又はプログレッシブフレームでは実行されない。ノンインタレース又はプログレッシブビデオに関して説明しているが、当業者であれば、本明細書で説明している原理及び利点は、インタレースビデオ及びノンインタレースビデオの両方に適用できることが理解できるであろう。さらに、本発明の実施の形態は、M P E G - 4に関して説明しているが、本明細書で説明している原理及び利点の種々の態様は、例えば、M P E G - 1、M P E G - 2、H . 2 6 1、H . 2 6 2、H . 2 6 3、H . 2 6 3 +、H . 2 6 3 + +、及びH . 2 6 L、及び今後策定されるであろうビデオ規格など、他の種々のビデオ規格にも適用することができる。

## 【 0 0 5 9 】

イントラコーディングIフレームは、通常、イメージ自体から得られる情報を含み、したがってIフレームは、他のフレームと独立して復号化することができる。P及びBフレームは、インターコーディングフレームとも呼ばれるが、それは、他のフレームから得られるデータに基づいて符号化されるからである。前処理モジュール102Bは、フレームに対応するフレームタイプ指定を含む、入力フレームタイプファイルと呼ばれるファイルを生成する。他の実施の形態では、フレームタイプ情報は、変数などを使用して符号化システム100Bの他の部分に渡される。図1Bでは前処理モジュール102Bは符号化システム100Bに含まれているように示されているが、前処理モジュール102Bは符号化システム100Bの他の部分と物理的に切り離すことができる。そのような実施の形態では、前処理モジュール102Bは、復号化システム100Bの残りの部分によって入力されるフレームタイプ指定を含むテキストファイルを、生成することができる。

## 【 0 0 6 0 】

多くの標準M P E G - 4エンコーダは、1つのシーン、即ち、1つのIフレームの後にPフレーム又はBフレームが続くもののみを処理することができるか、又はM P E G - 2符号化で通常行っているように、フレームk枚毎に通常のIフレームを導入する。これらのアプローチでは、エンコーダの実装が簡単になるが、複数のシーンからなるクリップがどのように処理されるべきかを決定するという負担が、ユーザーにかかる。コーディングの効率を十分に高めるために、Iフレームの数を減らすか、又は最小限に抑えなければならない。誤り状態がなければ、Iフレームをシーン変化のみで使用するのが好ましい。したがって、シーン変化を正しく、正確に検出することが有益である。

## 【 0 0 6 1 】

本発明の一実施の形態に係るシーン変化検出プロセスの一例について説明する。実施例では、このプロセスは Y U V - 4 : 2 : 0 ファイルに作用し、テキストファイルを出力する。実施例では、Y U V - 4 : 2 : 0 ファイルはフレームが連結されたヘッダのないファイルであり、フレーム毎に、(明度) Y ピクセル値が最初に与えられ、その後、(色度 - 青) C b 値、そして(色度 - 赤) C r 値が与えられる。「4 : 2 : 0」という表現は、色度値が明度に関して、因数 4 でサブサンプリングされることを示す。特に、フレームのサイズ(ピクセル単位)を W x H (W : 幅、H : 高さ)とすると、W \* H 個の Y 値(フレームピクセル毎に 1 つ)、(W / 2) \* (H / 2) 個の C b 値、(W / 2) \* (H / 2) 個の C r 値がフレーム毎にある。その結果、サイズ W x H のフレームを格納するために必要なフレームバッファサイズは合計で 3 \* W \* H / 2 バイトとなる。色度成分のサブサンプリングを行うために、垂直及び水平次元にそって因数 2 でサブサンプリングする。したがって、2 x 2 ブロックは、4 つの明度値、1 つの色度 - 青、及び 1 つの色度 - 赤を持つ。他の実施の形態では、イメージデータを格納するのに別の形式を使用することもできる。

10

【0062】

前処理モジュール 102 B について詳述する。前処理モジュール 102 B は、フレームの評価と符号化の指定を実行する。後述するように、それぞれのフレームは、前処理モジュール 102 B によって I フレーム、P フレーム、又はスキップフレームとして指定される。別の実施の形態では、前処理モジュール 102 B はさらに、フレームを B フレーム又は D フレームとして指定することもできる。B フレーム符号化は、計算能力が十分に高く、帯域幅が確保されていて(B フレームはスキップフレームよりも相当多くの帯域幅を占有する)、対応する規格によって許されている場合に実行できる。例えば、無線ネットワークで使用される M P E G - 4 の単純プロファイル構文は、B フレームに対応していない。前処理モジュール 102 B によって生成されるファイル形式例では、入力フレーム毎に 1 ラインを含み、各ラインにはフレームタイプ指定文字 0、1、又は 2 を含んでいる。「0」は I フレームを、「1」は P フレームを、「2」はスキップフレームを表す。他の実施の形態では、双方向フレーム及び D フレームに対して指定を行うことができる。

20

【0063】

既に述べたように、シーン変化フレームは一般にイントラコーディングされる。シーン変化フレームを見つけるために、前処理モジュールのシーン変化分析により、色重み付け二乗平均平方根(RMS)の計算と差の絶対値の平均(MAD)の計算とを、i 番目のフレーム F<sub>i</sub> と k 番目のフレーム F<sub>k</sub> との間で実行する。RMS は次のように定義することができる。

30

【0064】

【数 2】

$$\begin{aligned}
 \text{RMS}(F_i, F_k) = & \frac{\alpha}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^w \sum_{y=1}^h |Y_i(x, y) - Y_k(x, y)|^2} + \\
 & \frac{2\beta}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^{w/2} \sum_{y=1}^{h/2} |U_i(x, y) - U_k(x, y)|^2} + \\
 & \frac{2\gamma}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^{w/2} \sum_{y=1}^{h/2} |V_i(x, y) - V_k(x, y)|^2}
 \end{aligned}
 \tag{式 1}$$

40

ここで、F(x, y) はフレーム F 内の(x, y) 番目のピクセルを表し、w 及び h はそれぞれフレームの幅及び高さを表す。Y(x, y) は、輝度値を示し、U(x, y) 及び V(x, y) は 2 つの色度成分である。係数 α、β、γ は、それぞれ明度、色度 - 青、及び色度 - 赤成分の重み係数である。計算量を少なくするために、重み係数を固定することができる。例えば、重み係数を、α = β = γ = 1 と設定することができる。

【0065】

50

差の絶対値の平均 ( M A D ) 測度は次のように定義することができる。

【 0 0 6 6 】

【 数 3 】

$$MAD(F_i, F_k) = \frac{1}{w \times h} \sum_{x=1}^w \sum_{y=1}^h |Y_i(x, y) - Y_k(x, y)| \quad \text{式2}$$

この例では、M A D は、2つの色度成分を含んでいる必要はない。

【 0 0 6 7 】

M A D ( F<sub>i</sub> , F<sub>k</sub> ) 及び / 又は R M S ( F<sub>i</sub> , F<sub>k</sub> ) が選択した基準よりも大きい場合、このことは、F<sub>i</sub> の内容が実質的に F<sub>k</sub> と異なることを示している。従って、一実施の形態では、連続するフレーム間の M A D、M A D ( F<sub>i-1</sub> , F<sub>i</sub> ) があらかじめ指定したしきい値よりも大きい場合、F<sub>i</sub> はシーン変化フレームとして指定される。シーン変化フレームを指定するしきい値の一例は約 2.5 である。

【 0 0 6 8 】

オプションにより、R M S の二階時間微分を使用して、以下のように、フレームがシーン変化フレームであるかを判別することができる。

【 0 0 6 9 】

【 数 4 】

$$\frac{d^2(RMS)}{dt^2}(i) = RMS(F_{i-1}, F_i) - 2RMS(F_i, F_{i+1}) + RMS(F_{i+1}, F_{i+2}) \quad \text{式3}$$

式 3 によって定義されているように、R M S の二階時間微分は、現在のフレーム F<sub>i</sub> に対する前のフレーム F<sub>i-1</sub> の R M S 値、次のフレーム F<sub>i+1</sub> に対する現在のフレーム F<sub>i</sub> の R M S 値、それ以降のフレーム F<sub>i+2</sub> に対する次のフレーム F<sub>i+1</sub> の R M S 値に基づいている。

【 0 0 7 0 】

R M S 値の二階時間微分は、F<sub>i</sub> が図 3 に示されているようにシーン変化フレームである場合に比較的高い振幅を持つ負の値となる。従って、R M S 値の二階時間微分の絶対値があらかじめ指定したしきい値よりも大きい場合、F<sub>i</sub> はシーン変化フレームとして指定される。図 3 に示されているように、菱形で示されている R M S 値と三角形で示されている R M S の二階微分との間に相関関係がある。したがって、一般に R M S 値と R M S の二階微分の値からシーン変化の正しい指示が得られる。シーン変化を判別するための R M S しきい値の二階微分の一例は - 6.5 である。

【 0 0 7 1 】

R M S の二階微分は良いピーク検出手段となるが、幾分雑音に敏感である。シーン変化判別の精度をさらに高めるために、一実施の形態では、M A D の時間的活動測定と R M S の二階微分との両方が対応するしきい値以上である場合に、フレームがシーン変化フレームであると指定される。後述するように、シーン変化フレームが I フレーム又は I - V O P としてイントラコーディングされる。

【 0 0 7 2 】

特に、一実施例では、フレームの M A D が 2.0 を超えていて、R M S の二階微分が負であり、その絶対値が 4 よりも大きい場合、そのフレームはシーン変化として指定され、イントラモードでコーディングされる。他の実施例では、フレームの R M S が 4.0 を超えている、及び / 又は R M S の二階微分が負であり、その絶対値が 8 よりも大きい場合、そのフレームはシーン変化として指定され、イントラモードでコーディングされる及び / 又は。他の実施の形態では、他のしきい値を使用することができる。それとは別に、又はそれに加えて、フレームがシーン変化に該当するか否かをさらに示すものとして、R M S の二階微分に関して上記したの同様に、M A D の二階微分を使用することができる。

## 【0073】

シーン変化が発生したことを判別するために、追加基準を使用することができる。例えば、一実施の形態では、MAD値が極大値であるかどうか、即ち、前のフレームから問題のフレームに上昇し、その後問題のフレームから次のフレームに減少した否かを判別する。もしそうであれば、これは、問題のフレームがシーン変化フレームである可能性が高く、イントラコーディングされるべきであることを示している。さらに、RMS値に関して類似の判別を行うことができる。例えば、RMS値が極大値であるか否か、即ち、前のフレームから問題のフレームに上昇し、その後問題のフレームから次のフレームに減少したか否かを判別する。もしそうであれば、これもまた、問題のフレームがシーン変化フレームである可能性が高く、イントラコーディングされるべきであることを示している。

10

## 【0074】

オプションとして、RMS、RMSの二階微分、及びMADのうち少なくとも2つが対応する基準を満たしている場合に、フレームがイントラコーディングすべきシーン変化として指定される投票プロセス(voting process)を使用することができる。他の実施の形態では、RMS及びRMSの二階微分が対応する基準を満たしている場合、及びMADが極大値である場合に、フレームはシーン変化フレームとして指定される。さらに他の実施の形態では、RMS及びMADが対応する基準を満たしている場合、及びMADが極大値である場合に、フレームはシーン変化フレームとして指定される。

## 【0075】

コーディングの効率をさらに高め、ターゲットのビットレートを満たすために、1秒当たりの符号化すべきフレーム数をできる限り減らすのが好ましい。1秒当たりの符号化されるフレーム数を減らすために使用される方法の1つとして、符号化プロセスのスキップフレームがある。フレームスキップ手法の例として、固定フレームスキップと適応型フレームスキップの2つがある。従来のエンコーダでは、単純なスキップアルゴリズムに従って、複数のフレームを落としてフレームレートを下げている。例えば、従来のエンコーダは、ビデオクリップ内の5つのフレーム中の4つを落として、ビデオクリップを毎秒30フレームのフレームレートから毎秒6フレームのフレームレートに変換する。

20

## 【0076】

後述するように、固定フレームスキップは雑音の多い環境では優れた誤り耐性を持つ傾向があるが、適応型フレームスキップは雑音の少ない環境において良い視覚的結果を生じる傾向がある。オプションとして、前処理モジュール102Bは、下の式7で定義されているビットレート/フレームレートの式に基づき、ターゲット符号化フレームレートを計算し、適応型スキップと固定スキップを切り替えて、ターゲットの符号化フレームレートに追従することができる。

30

## 【0077】

固定フレームスキップでは、入力ビデオフレームシーケンスは、 $k$ をサブサンプリング係数とすると、フレーム $k$ 個毎に1つ保持することにより時間軸に沿ってサブサンプリングされる。例えば、

$k = 5$ 、及び

元のビデオシーケンスのフレームレート = 25 フレーム / 秒 (fps) であれば、サブサンプリングされたシーケンスのフレームレート = 5 fps である。

40

## 【0078】

適応型フレームスキップでは、固定フレームスキップのように、入力ビデオフレームシーケンスは、望みの、又は所定の平均フレームレートを得るために、時間軸に沿ってサブサンプリングされる又は。しかし、固定方式のスキップフレームではなく、適応型フレームスキップを使用すると、フレームスキップの速度は、不規則になり、シーケンス長に沿って変化する場合がある。低活動のフレームを識別してスキップし、シーン変化フレームを保持してイントラコーディングするのが好ましい。何らかの活動フレームのある非シーン変化はインターコーディングされる。そのスキップフレームは、視覚的活動に対する変化に基づいてインテリジェントに選択されているために、デコーダによって再現されたとき

50



の視覚的結果は、発生する誤りがないか、又は比較的少ないと仮定すると、固定フレームスキップの場合よりも優れている。

【0079】

一実施の形態では、前処理モジュール102Bは、MPEGビットストリーム内のビデオオブジェクトプレーン(VOP)ヘッダにセットされている“not\_coded”ビットフラグ又はインジケータを使用して、スキップフレームをコーディングする。MPEG-4ビデオパケットは、VOPヘッダ又はビデオパケットヘッダから始まり、その後、motion\_shape\_texture()が続き、そしてnext\_resync\_marker()又はnext\_start\_code()で終わる。したがって、VOPは、特定のイメージシーケンス内容を指定し、輪郭、動き、及びテクスチャの情報をコーディングすることにより、別々のビデオオブジェクトレイヤ内にコーディングされる。

10

【0080】

別の実施の形態では、スキップフレームと一緒にスキップされ、ビットストリーム内にVOPヘッダ情報は挿入されない。スキップフレームを、デコーダで補間を使用することにより、又は前のフレームを繰り返すことによって再作成してもよい。デコーダは、先行フレームと後続の復号化フレームとの間でピクセル平均を取り、それらの時間差で重み付けすることによって補間を実行してもよい。

【0081】

ビデオシーケンスのフレームレートに関する情報は、通常、Video-Object-Layer(VOL)ヘッダで搬送される。特に、vop\_time\_increment\_resolutionというパラメータにより、それぞれの符号化サイクルの時間単位数を決める。Video-Object-Plane(VOP)ヘッダ内のvop\_time\_increment値により、フレーム毎にタイムスタンプが付けられる。

20

【0082】

vop\_time\_increment\_resolutionは、例えば、16ビット符合なし整数値であってもよい。例えば、25フレーム/秒(fps)の場合、vop\_time\_increment\_resolution=25で、vop\_time\_incrementは値0...24を循環する。7.5fpsの場合、vop\_time\_increment\_resolution=75で、vop\_time\_incrementは値0、10、20、30、40、50、60、70、5、15、25、35、45、55、65と循環する。したがって、シーケンスの最初の2つのフレームの復号が成功すると、正確なフレームレートが求められる。しかし、誤りを起こしやすい無線通信環境又はその他の誤りを起こしやすい環境では、任意の2つの連続するフレームを正常に受信し、復号することは保証できない。したがって、デコーダは、シーケンス全体のフレームレートを正しく判別できない場合がある。そのような理由から、一実施の形態では、エンコーダモジュール106BがVOLヘッダ内の“fixed\_vop\_rate”フラグをセットし、デフォルトのフレームレートを(fixed\_vop\_time\_incrementの値を使って)供給する。この方法を使用すると、VOLヘッダが正常に復号された後、フレームレートの復号又は判別を正常に行える場合が多くなる。後に、fixed\_vop\_time\_increment値をデコーダのグローバル変数に格納し、その値を使用して特定のフレームを補間する必要があるかどうかを判別できる。補間するフレームは、エンコーダによってスキップされるフレームであるか、又は送信中に失われたフレームのいずれかである。したがって、正しい数のフレームを復号し、それによってオーディオストリームでの同期喪失問題を回避できるため、MPEG-4デコーダの誤り耐性の性能が高められる。

30

40

【0083】

上記したように、誤りを起こしやすい環境では、誤り耐性を高めるために、適応型フレームスキップの代わりに固定フレームスキップを使用する。固定フレームスキップでは、フレームが落とされている、又はスキップされているときに、デコーダ側で判別しやすくな

50

る。誤り耐性を高める別のアプローチとして、適応型フレームスキップを使用するが、ただしVOPヘッダを供給する際にスキップフレームについて`not_coded`フラグをセットするという方法がある。このアプローチの欠点の1つは、頻繁なVOPヘッダによりビットレートがわずかに増大するという点である。

【0084】

図4Aは、適応型フレームスキップの一例のプロセス400を示している。このプロセスは、注目しているフレームに隣接するフレーム間の差の絶対値の平均を計算し、その計算に対して時間パラメータの重み付けを行うことによってフレーム列からフレームを選択的に落とすという点で反復的であり、シーンに対する影響が最小であるフレームが落とされる。この手順は、目的のビットレート及びフレームサイズに関係するターゲットのフレームレートが得られるまで繰り返し実行される。

10

【0085】

プロセス400は、開始状態402から始まり、状態404に進む。目的のフレームレートを設定又は指定する。目的のフレームレートは、ユーザーが指定することもでき、又は動的に決定することもできる。状態406に進み、特定のフレームを落としたことによって生じる、コスト関数、又は悪影響をシーン内の第1のフレームから最後のフレームまでの間の各フレームについて計算する。後で詳しく説明するが、コスト関数は、注目する特定のフレームを、接近して、もしくは最も接近して挟む、又はそのフレームに隣接するフレーム間の差の絶対値の平均(MAD)、あるいは差の絶対値の平均の和(SMAD)に少なくとも一部分は基づくことができる。それとは別に、又はそれに加えて、コスト関数は、注目する特定のフレームを挟むフレームに関するRMSの和(SRMS)に基づくことができる。

20

【0086】

状態408に進むと、最小コストに関連付けられた、即ち、視覚的品質に対する悪影響が最小であるフレームはスキップされるか、又は落とされる。状態410において、残りのフレームによってターゲットフレームレートの条件が満たされるか否かを判別する。ターゲットフレームレートの条件が満たされる場合、適応型フレームレートスキッププロセス400は終了状態414に進む。そうでない場合、プロセス400は状態412に進み、コスト最小の残りのフレームが落とされる。最初のフレームから最後のフレームまでの間に残っている全てのフレームのコストが、現在残っているフレームに基づいて状態412で再計算され、コストが最小のフレームが落とされる。プロセス400は、状態410と412を繰り返し、ターゲットフレームレートの条件が満たされるか、又はすでに連続してスキップされているフレームの数が指定の最大値に達すると、その繰り返しを終了する。プロセス400はスキップするフレームを選択することに関して説明されているが、プロセス400を同様に使用して、どのフレームを双方向符号化すべきかを決定することができる。

30

【0087】

差の絶対値の平均の和に重みを付けたものを使用する適応型フレームスキップ選択プロセスの一実施例について詳しく説明する。フレームレート $r_{orig}$ と望みのフレームレート $r_{des}$ (ここで、 $r_{orig} > r_{des}$ )の入力シーケンスが与えられた場合、ビデオシーケンスアナライザは、「積極的に」フレームをスキップする、即ち $r_{des}$ 条件が満たされるまでスキップする。スキップする次のフレームを識別するために、コスト関数を指定する際にMADの重み付き和と2つのフレームの間の時間差を使用する。コスト関数が所定の基準を満たしているか、又は残りのスキップの候補のうち最小であるフレームはスキップされる。

40

【0088】

例えば、図4Bを参照すると、積極的なアプローチを使用して $F_3$ 、 $F_5$ 、 $F_6$ 、及び $F_8$ をすでにスキップしている場合、 $F_4$ は、次のスキップフレームの候補とみなされる。 $F_4$ がスキップされるかどうかを判別するために、 $F_4$ がスキップされたと仮定してコスト関数を計算する。 $F_4$ をスキップした後、 $F_2$ 及び $F_7$ が $F_3 \sim F_6$ のスキップフレームの境界で

50

ある左右のフレームになる。そのコスト関数は次のように定義される。

【 0 0 8 9 】

【 数 5 】

$$\text{コスト} = \text{MAD}(F_2, F_7) + \lambda \frac{29.97}{r_{\text{orig}}} \text{TD} \quad \text{式 4}$$

ここで、項

【 数 6 】

$$\frac{29.97}{r_{\text{orig}}}$$

10

は、NTSCフレームレート29.97フレーム/秒に関して元のフレームレート $r_{\text{orig}}$ を正規化するために使用され、TDは、時間差の単位(time difference measure)を表す。もちろん、別の正規化又は異なる正規化も使用できる。

【 0 0 9 0 】

したがって、この例では、TDは5(=7-2)であり、 $\lambda$ は重み係数である。この例では、実験で求められた重み係数値 $\lambda = 5.0$ で、適切な結果が得られる。それとは別に、重み係数値 $\lambda$ は、動的に求めることもできる。現在の状態で、コスト関数が候補の中で最小であるフレームがスキップされる。このプロセスを繰り返し実行して、目的のフレームレートを得る。時間差の単位TDを修正して、 $n$ 個を超える連続フレームがスキップされないようにする。例えば、同時にスキップできる連続フレームの最大数が予め4に設定されている場合、TDは次のように修正することができる。

20

【 0 0 9 1 】

【 数 7 】

$$TD' = \begin{cases} TD \leq 5 \text{ の場合} & TD \\ \text{そうでない場合} & \infty \end{cases}$$

30

ここで、 $\infty$ は無有限大である。

【 0 0 9 2 】

オプションとして、計算の中で偶数(又は奇数)の座標ピクセルのみを使用すれば、MAD計算の複雑度を減らすこともできるが、精度は低下する。たとえば、以下のとおりである。

【 0 0 9 3 】

【 数 8 】

$$\text{MAD}(F_i, F_k) = \frac{1}{w/2 \times h/2} \sum_{x=1}^{w/2} \sum_{y=1}^{h/2} |Y_i(2x, 2y) - Y_k(2x, 2y)| \quad \text{式 6 a}$$

40

スキップするフレームを適応的に決定する別のプロセス例では、誘起する空間的及び時間的な全歪みを、スキップ対象のフレーム毎に推定し、そのフレームが存在しなければ歪みが最小になるフレームをスキップする。後述するように、このプロセスではMADの和(SMAD)又はRMSの和(SRMS)を使用する。都合がよいことに、全てのMADを再計算しなくてもよい。その代わりに、すでに計算されている適切なMADは、スキップ対象とみなされるフレームに依存して異なるように加算される。

【 0 0 9 4 】

一実施の形態では、このプロセスは次のように実行される。 $F_i$ が現在スキップ対象とみなされているフレームであると仮定する。例えば、 $F_i$ は図4B内のフレーム $F_4$ であると

50

仮定する。次に、 $F_j$ を前の非スキップフレーム、この例では $F_2$ とし、フレーム $F_k$ は次の非スキップフレーム、この例では $F_7$ を示すものとする。すると、コスト関数は次のように定義することができる。

【0095】

【数9】

$$\text{コスト} = \left| \text{SMAD}_{\text{est}}(F_j, F_k) - \text{SMAD}_{\text{est}}(F_j, F_1) - \text{SMAD}_{\text{est}}(F_1, F_k) \right| + \lambda \frac{29.97}{r_{\text{orig}}} \text{TD} \quad \text{式6 b}$$

ここで、 $\text{SMAD}_{\text{est}}(F_i, F_k)$ はフレーム $(i+1), \dots, (k-1)$ をスキップしたときの推定空間歪みである。

【0096】

式6 bからわかるように、コスト関数は前のスキップフレームから寄与分を差し引く。歪みは次のように計算される。

【0097】

【数10】

$$\text{SMAD}_{\text{est}}(F_i, F_k) = \varphi(k-i-1) \cdot \frac{k-i-1}{k-i} \sum_{j=i+1}^k \text{MAD}(F_{j-1}, F_j) \quad \text{式6 c}$$

ここで、 $\varphi(n)$ は連続するスキップフレームの数に依存する係数であり、平均して、又は統計的サンプリングに基づいて、デコーダ側の補間フレームがどれだけ元のフレームと異なるかを考慮するものである。以下の表は、代表的なビデオシーケンスを使用して実験的に求められた係数値の例である。

【0098】

【表1】

n	0	1	2	3	4	5	6	>=7
$\varphi(n)$	0.0	0.746	0.886	1.010	1.198	1.276	1.348	1.500

適応型フレームスキップ係数の表

これからわかるように、 $\varphi(n)$ はスキップフレームの数が増えると増加する。

【0099】

必要な計算複雑度及びリソースを低減するために、オプションとして、連続するフレーム $(j-1$ と $j)$ の間のMADのみを使用して、上の式6 bのように、スキップフレームの完全なセグメントの歪みを推定する。特定のフレームをスキップしたときに生じる追加歪みを計算し、時間成分を加える。

【0100】

一実施の形態では、ユーザーが望みの符号化フレームレートを指定する。その望みのフレームレートは、時間的及び空間的複雑度、フレームサイズ、フレームレート、及びターゲットビットレート又は圧縮比などのビデオシーケンス統計量に基づくことができる。別の実施の形態では、発見的手法により望みのフレームレートを選択することができる。以下は、144行、1行当たり176ピクセルを含む、Quarter Common Intermediate Format (QCIF) フレームサイズに関して正規化された、フレームレートを計算する発見的方程式例である。

【0101】

【数11】

$$\text{符号化フレームレート} = \frac{\text{ターゲットビットレート (kbps)}}{8 \sqrt{\frac{176 * 144}{w * h}}} \quad \text{式7}$$

ここで、 $w$ 及び $h$ はフレーム次元である。符号化フレームレートが妥当な境界範囲内にあ

るためには、符号化フレームレートは次の範囲にあることが好ましい。

[ 1 からソースフレームレートまで ]

さらに、適応型フレームスキップを特定のシーンの時間的複雑度に依存しないものとするために、オプションとして重み係数 が、指定されたシーケンス全体の平均 R M S 又は M A D に等しくなるように設定される。

【 0 1 0 2 】

したがって、どのフレームがシーン変化フレームであるかを判別し、スキップするフレームを適応的に選択するために、ビデオシーケンスアナライザに、ビデオシーケンス、フレーム幅、フレーム高さ、ソースフレームレート、ターゲットビットレート、及び誤り耐性フラグを以下の形式で与える。

【 0 1 0 3 】

< 入力ファイル > < 幅 > < 高さ > < ソースフレームレート > < ターゲットビットレート > < 誤り耐性フラグ >

誤りが全くないか少ない場合に、誤り耐性は低いが見覚的結果に優れる適応型フレームスキップと、誤り耐性に優れるが見覚的結果が劣る固定フレームスキップとを切り替えるために、誤り耐性フラグはユーザーによってセットされる。

【 0 1 0 4 】

一実施の形態では、図 1 C に示されているように、R M S 回路 1 0 2 C を使用して、上述のように R M S 値を計算し、R M S の二階微分回路 1 0 4 C を使用して、上述のように R M S の二階微分を計算し、M A D 回路 1 0 8 C を使用して、上述のように M A D 値を計算し、M A D の加算回路 1 1 0 C を使用して、上述のように M A D 値の和を計算し、M A D の二階微分回路 1 1 4 C を使用して、上述のように M A D の二階微分を計算する。R M S 回路 1 0 2 C、R M S の二階微分回路 1 0 4 C、M A D 回路 1 0 8 C、及び M A D の加算回路 1 1 0 C、及び M A D の二階微分回路 1 1 4 C の出力に結合されているエバリュエータ回路 1 1 2 C を使用して、いつシーン変化が発生したか、どのようなフレームをスキップするかを、上記した 1 つ又は複数の出力に基づいて決定する。もちろん、異なる実施の形態は、図 1 C に示されている回路の全部又は一部を含んでいる必要はない。

【 0 1 0 5 】

次に、ビット割り当てモジュール又は回路 1 0 4 B について説明する。ビット割り当てモジュール又は回路 1 0 4 B は、シーン、フレーム、及びノ又はマクロブロックレベルでビット割り当てを行う。ビット割り当てモジュール 1 0 4 B はファイルを読み込むか、又は他の方法で、フレームタイプ指定を含む、前処理モジュール 1 0 2 B によって生成された情報を受け取り、コーディングフレームに基づきそれぞれのシーン、G O V、又は G O P についてビット割当値 ( bit budget ) を計算する。ビット割り当てモジュールにより、固定されたビット割当値の適切な分配が決定される。

【 0 1 0 6 】

一実施例に関して後述するように、第 1 のイントラコーディングフレームは、シーンの先頭を定義する。イントラコーディングフレームの数とインターコーディングフレームの数に基づいてシーンに重みが割り当てられ、イントラコーディングされたフレームは、インターコーディングされたフレームよりも重い重み付けがなされ、イントラフレームを符号化するために必要なビットをより多く占有する。ビット割り当てモジュールは、現在のビット使用度と目標のビット使用度とを比較してシーン内の固定ビット割当値を配分し、その比較に基づいて、現在のフレームに対する量子化パラメータ又はステップサイズを調整する。

【 0 1 0 7 】

特に、ビット割り当てモジュール 1 0 4 B はまず、前処理モジュール 1 0 2 B から入力されたフレームタイプファイルを解析する。次に、G O V の数又は量を計算する。計算されたビット割当値に基づき、エンコーダモジュール 1 0 6 B が、後述のエンコーダパラメータファイルによる新規レート制御プロセスを使用してそれぞれの G O V を符号化する。

【 0 1 0 8 】

10

20

30

40

50

与えられた G O V 又はシーンについて、コーディングされたフレームの数及び最初のフレーム及び最終フレームが知られている。以下の定義を使用し、以下の式 8 により、G O V 又はシーンのビット割当値を計算する。

【 0 1 0 9 】

$N_{c_i}$  = scene  $i$  又は G O V  $i$  内のコーディングされた P - V O P ( 予測、インターコーディング V O P ) の数。

$B_i$  = シーン  $i$  のビット割当値

$B$  = 1 つ又は複数のシーンを含むクリップのビット割当値

$N_c$  = クリップのコーディングフレームの数

$N_s$  = クリップ内のシーンの数。通常は、誤り耐性を高める目的で連続する I フレームを含めない場合、 $N_s = I - V O P$  ( イントラコーディング V O P ) の数 10

$T_c$  = クリップ内の V O P の相当する総数

【 0 1 1 0 】

シーン又は G O V のビット割当値を決定する方法の例を以下に示す。 $B_i$  ビットを G O V (  $i$  ) 毎に割り当てる。この例の割り当てでは、1 つの I フレーム又は I - V O P に対するビット使用度は、P フレーム又は P - V O P のビット使用度に近いか又は等しい (  $R a t i o \_ O f \_ I \_ t o \_ P = 1 0$  ) という想定がなされる。しかし、この方法は、1 : 1 0 の比を使用することに制限されていない。 $B_i$  及び  $T_c$  は、次のように決定される。

【 0 1 1 1 】

$B_i = B * ( N_{c_i} + R a t i o \_ O f \_ I \_ t o \_ P ) / T_c$  式 8 20

及び

$T_c = N_c + ( R a t i o \_ O f \_ I \_ t o \_ P - 1 ) * N_s$  式 9

式 8 で定義されているように、与えられたシーンのビット割り当ては、シーン内のフレームの総数に基づいており、イントラコーディングフレームは、複数の予測フレームと同等に正規化される。複雑度及び計算のオーバーヘッドを減らすために、このビット割り当て公式の例では、各 G O V 又は G O P の空間的及び時間的複雑度を考慮しない。別の実施の形態では、十分な計算的及び時間的資源が利用できる場合に、ビット割り当て公式において、時間的及び空間的複雑度を考慮し、G O V 毎にさらに理にかなったビット割り当てを行うことができる。

【 0 1 1 2 】

例えば、一実施の形態では、空間的及び時間的複雑度を考慮する 2 パス方式の符号化プロセスを使用する。第 1 のパスで、シーン変化を検出し、フレーム複雑度を収集する。第 2 のパスで、複雑度によって導き出されるビット割り当てを使用して実際の符号化を行う。 30

【 0 1 1 3 】

第 1 のパスのプロセスについて詳述する。新しい G O V 又は G O P は、シーン変化インスタンスから始まる。2 パスレート制御プロセスは、それぞれの時間的にセグメント分割された G O V について実質的に一様な品質を備えるので、品質の変動はうまく G O V 又は G O P 境界に制限される。このアプローチを取ったのは、さまざまなシーン間のピーク信号対雑音比 ( P S N R )、二乗平均平方根誤差、又はその他のイメージ忠実度測定基準によって測定されるような品質変動の最小化を行うことが、人間の視覚認知に関してあまり有利とはいえないからである。 40

【 0 1 1 4 】

相対的フレーム複雑度を特徴付ける場合、本発明の一実施の形態による複雑度測定は、使用する量子化パラメータ ( Q P ) について比較的不変である。特に、 $H_i$  で表されるフレームヘッダ / 構文及び動きベクトルなどの非テクスチャ情報のビットカウントは、量子化パラメータ ( Q P ) の変化に関して一定であるか、又はほとんど一定であることが一般的である。これは、Q P の変化とともに変化する、テクスチャ情報のビットカウントと対照的である。例えば、M P E G - 4 V M R - Q モデル [ M P E G 4 V M ] に基づいて、現在のフレーム  $i$  をコーディングするのに使用されるビット総数を  $R_i$  とすると、テクスチャビット  $T_i = R_i - H_i$  は次のように表すことができる。 50

【 0 1 1 5 】

【 数 1 2 】

$$\frac{R_i - H_i}{M_i} = \frac{a_1}{Q_i} + \frac{a_2}{Q_i^2} \quad \text{式10}$$

ここで、 $M_i$ は、 $QP$ （即ち、 $Q_i$ ）に関して実質的に不変な動き補正残差で計算した  $MA D$  であり、 $a_1$  及び  $a_2$  は  $QP$  上のテクスチャビット  $T_i$  の  $T a y l o r$  展開係数である。係数  $a_1$  及び  $a_2$  は、通常は同じ次数である、即ち同様の値を持つ。これからわかるように、 $QP$  が低いほど、与えられたフレームを符号化するのに必要なテクスチャビット数は多くなる。

10

【 0 1 1 6 】

複雑度の測定  $C_{g,i}$  は、動き及びテクスチャビットカウントの両方を対象とし、実質的に  $QP$  不変である。一実施の形態では、 $C_{g,i}$  は、以下のように、与えられたフレームのテクスチャビットカウントと平均テクスチャビットカウントとの比、及び与えられたフレームの動きベクトルビットカウントと平均動きベクトルビットカウントとの比によって定義される。

【 0 1 1 7 】

【 数 1 3 】

$$C_{g,i} = (R_{g,i} - H_{g,i}) / \overline{R_g - H_g} + MV_{(g,i)} / \overline{MV_g} \quad \text{式11}$$

20

ここで、 $MV_{(g,i)}$  はフレーム  $(g, i)$  の動きベクトルビットカウントであり、

【 数 1 4 】

$$\overline{MV_g}$$

は平均動きベクトルビットカウントであり、

【 数 1 5 】

$$\overline{R_g - H_g}$$

30

は平均テクスチャビットカウントである。得られた複雑度の測定  $C_{g,i}$  は実質的に  $QP$  不変なので、フレーム複雑度は 1 パスで、 $QP$  で生成することができる。計算で求められたフレーム複雑度は、これから説明するように、第 2 の符号化パスで利用される。

【 0 1 1 8 】

図 9 に示されているように、第 2 パスのレート制御プロセス 900 は 3 レベル階層、即ちシーン ( $GOV$  又は  $GOP$ ) レベルビット割り当て 902、フレームレベルビット割り当て 904、及びマクロブロックレベル  $QP$  調整 906 に分かれており、得られたフレーム複雑度値  $C_{g,i}$  を使用する。

【 0 1 1 9 】

複雑度によって導かれるシーン、 $GOV$ 、又は  $GOP$ 、レベルビット割り当てについて、図 10 を参照しながら説明する。ビットをそれぞれのシーンに割り当てるには、以下のよう

40

【 数 1 6 】

$$\overline{C}^{(g)}$$

を計算して  $GOV$  又は  $GOP$  毎に複雑度の測定を定義する。

【 数 1 7 】

$$\overline{C}^{(g)} = \sum_i C_{g,i} / N_g \quad \text{式12}$$

50

## 【 0 1 2 0 】

次に、図 1 0 に示されているように、G O V 又は G O P レベル再帰的ビット割り当てプロセス 1 0 0 0 を適用する。状態 1 0 0 2 で、以下の設定で、初期化プロセスを実行する。  
= 0

ビット割当値  $B_r$  (特定の数の G O V 又は G O P に対応する与えられた時間枠に対するビット割り当て) = B

初期送信バッファ飽和度  $\rho_1 = T_d \times R$

インデックス 1 の G O V 又は G O P の開始、即ち最初の G O V 又は G O P。

## 【 0 1 2 1 】

状態 1 0 0 4 で、以下の式によりビットがインデックス  $g$  のシーン (G O V 又は G O P) に割り当てられる。 10

## 【 0 1 2 2 】

## 【 数 1 8 】

$$B_{t(g)} = \lambda \times (R/F) \times N_{(g)} + (1-\lambda) \times \frac{C_g \times N_g}{\sum_i C_i \times N_i} \times B_r \quad \text{式 1 3}$$

ここで、

R = チャネルレート

F = 選択されたフレームレート 20

$N_{(g)}$  = インデックス  $g$  の G O V 又は G O P 内のフレーム数

= バッファ変動と複雑度の要求との間の重み係数

であり、

## 【 数 1 9 】

$$\sum_i C_i \times N_i$$

は、注目している G O V 又は G O P に対応する与えられた時間枠の全シーン複雑度を定義する。

## 【 0 1 2 3 】

= 0 のケースは、送信バッファ制約条件を満たすことができる場合に好ましいフレーム複雑度に直接従うビット割り当て方式を表す。 = 1 . 0 の指定は、フレーム複雑度を考慮することなくビット割当値が均等に分配されるケースを表す。この場合、最初のフレームのみプリフェッチするだけでよいので、プリロードはほとんど必要なく、小容量のデコーダバッファが必要である。  $0 < < 1 . 0$  のケースは、バッファと品質制約条件との間のビット割り当てのトレードオフ関係を表す。 30

## 【 0 1 2 4 】

状態 1 0 0 6 では、次の場合に、仮に割り当てられたビット割当値  $B_{t(g)}$  でバッファステータスを検査する。

$$\rho_{g-1} + B_{t(g)} - (R/F) \times N_{(g)} < \text{MarginFactor}1 \times \rho_{\text{max}}$$

ここで、例えば、MarginFactor1 = 0 . 8 が、バッファ調整の安全マージン (最大バッファサイズの 0 . 8 ) となる。 40

## 【 0 1 2 5 】

そして、その割り当てが許容され、プロセス 1 0 0 0 が状態 1 0 0 8 に進む。そうでない場合、 $\rho_{g-1}$  の値に 0 . 1 などの係数を加算して調整し、プロセス 1 0 0 0 は状態 1 0 0 4 に戻る。

## 【 0 1 2 6 】

状態 1 0 0 8 で、バッファステータスは次のように更新される。

$$\rho_g = \rho_{g-1} + B_{t(g)} - (R/F) \times N_{(g)}$$

残りの割当値  $B_r$  は次のように調整される。 50



$B_r - B_{t(g)}$

プロセス 1008 は、状態 1002 に戻り、 の値は 0 に設定される。次のインデックス  $g + 1$  の GOV 又は GOP の割り当てが実行される。

【0127】

GOV 又は GOP シーンレベルビット割り当てにより、バッファと品質制約条件の両方を満たしながら、各 GOV 又は GOP にビット割当値を都合よく適切に割り当てることができる。しかし、それぞれの GOV 又は GOP 内の一定又は一貫した品質を得るために、フレームの複雑度に基づき、バッファ制約条件を満たしながら、それぞれの GOV 又は GOP 内のフレームに従ってビット割当値を割り当てるのが好ましい。図 9 に示されているフレームレベルビット割り当てプロセス 904 は、このようなフレームビット割り当てプロセスを実行し、これは GOV 又は GOP レベルのビット割り当てのためのプロセス 1000 に類似している。しかし、GOP 又は GOV に対応する変数を使用する代わりに、フレームに対応する変数を使用する。そこで、フレームについて以下が成り立つ。

$$B_{f_i} = \alpha \times (R / F) + (1 - \alpha) \times C_{(g,i)} \times B_{g'} / (C_{g'}) \quad \text{式 14}$$

ここで、

$B_{g'}$  = ランニングビット = (現在の GOP へのビット割り当て - 既にコーディングされたフレームに使用されたビット数)

$C_{g'}$  = ランニング複雑度 = (現在の GOP の複雑度 - 既にコーディングされたフレームの複雑度)

R = ターゲットビットレート

F = フレームレート

一実施の形態では、以下の説明に従ってマクロブロックレベル QP 調整 906 を代わりに実行することができる。シーン及び上記で説明したフレームレベルビット割り当てプロセスでは、安全マージン (例えば最大バッファの 0.8) がバッファ調整用に用意される。計算複雑度を低減するために、他の実施の形態では、1パスレート制御を使用して、全てのマクロブロックを同じ量子化パラメータ (QP) で量子化する。QP 値は、以下の反復プロセスを使用してフレームレベルレート制御の一部として決定される。

【0128】

If  $B_{\text{actual}}(i) > 1.15 * B_t(i)$ , then  $QP_{i+1} = QP_i + 1$ .

If  $B_{\text{actual}}(i) \leq 1.15 * B_t(i)$

then if  $B_{\text{actual}}(i) < 0.85 * B_t(i)$

$QP_{i+1} = QP_i - 1$

else  $QP_{i+1} = QP_i$ .

// $QP_{i+1}$  が有効な QP 範囲 1~31 にあることを保証するため、以下のクリッピング操作を実行

$QP_{i+1} = \max(QP_{i+1}, 1)$ ;

$QP_{i+1} = \min(QP_{i+1}, 31)$ .

しかし、バッファが比較的小さい場合、後述するマクロブロックレベルレート制御プロセスを使用することができる。

【0129】

$N_{MB}$  が 1 フレーム内のマクロブロックの数とすると、 $MAD_k$  は  $MB_k$  の平均の差の絶対値であり、 $QP_{k-1}$  は前のマクロブロックの QP である。従って、現在の  $MB_k$  の QP は、以下の規則に従って決定されるように、範囲  $[QP_{k-1} - 2, QP_{k-1} + 2]$  にあると考えられる。

【0130】

10

20

30

40

50

【数 2 0】

$$QP_k = \begin{cases} R_{k-1} > 1.5 \times B_{k-1} \times (MAD_{k-1} / \sum_{k-1}^{N_{MB}} MAD_m) \text{ ならば } QP_k - 2, \text{ あるいは} \\ R_{k-1} > 1.25 \times B_{k-1} \times (MAD_{k-1} / \sum_{k-1}^{N_{MB}} MAD_m) \text{ ならば } QP_k - 1, \text{ あるいは} \\ R_{k-1} < 0.67 \times B_{k-1} \times (MAD_{k-1} / \sum_{k-1}^{N_{MB}} MAD_m) \text{ ならば } QP_k + 2, \text{ あるいは} \\ R_{k-1} < 0.8 \times B_{k-1} \times (MAD_{k-1} / \sum_{k-1}^{N_{MB}} MAD_m) \text{ ならば } QP_k + 1, \text{ あるいは} \\ QP_k \end{cases} \quad \text{式 1 5}$$

10

オプションとして、雑音の大きな環境では、マクロブロックビット割り当てプロセスを無効にするか、又は使用しないことで、QPがそれぞれのマクロブロックについて同じであるとデコーダ側で仮定することができる。これにより、フレームの一部が破損していたり失われているときにデコーダは正しくないQPを使用しないようにできる。

【0131】

各GOV又はGOPのビット割当値を決定した後、IMP4とも呼ばれる新規レート制御プロセスを使用して、計算で求められたビット割当値を満たすか又は実質的に満たす。従来のMPEG-4レート制御は、多くのアプリケーションに対して適切なパフォーマンスを発揮する。例えば、多くの従来型のレート制御プロセスは多重シーンを明示的にサポートしていない。その代わりに、これらの従来型のレート制御プロセスでは、シーケンス全体が単一のシーンを含むと仮定しており、したがって満足できるレート制御を行うことができない。それに対して、本発明の一実施の形態では、シーン変化の発生を考慮し、レート制御機能を高めている。

20

【0132】

本発明の一実施の形態では、自己収束レート制御プロセスを使用し、QPを量子化ステップサイズの半分に等しいとした場合に、各フレームの量子化パラメータQPを調整することによってそれぞれのGOVのビット割当値を満たす。MPEG-4では、量子化パラメータQPは31個の値[1~31]を持つことができる。レート制御プロセスでは、過去のビット使用度、未コーディングフレームの数、与えられたGOVに対する残りのビットに基づいてQPを決定する。したがって、現在のビット使用度が割り当てられたビット割当値からある量又はパーセンテージ以上超えた場合に、量子化パラメータ、したがって量子化ステップサイズが増やされる。その代わりに、現在のビット使用度が割り当てられたビット割当値からある量又はパーセンテージを超えて下がった場合、量子化パラメータ、したがって量子化ステップサイズは減らされる。このプロセスは、以下の疑似コード例に従って実行することができる。

30

【0133】

```
if (現在のビット使用度 > 割り当てられたビット割当値 * Margin1)
QPnext = min(31, INT(QPcur + StepUpPrcnt * QPcur));
Else if (現在のビット使用度 < 割り当てられたビット割当値 * Margin2)
QPnext = max(1, INT(QPcur - StepDwnPrcnt * QPcur));
End If.
```

40

ここで、

Margin1は、現在のビット使用度が割り当てられたビット割当値を超えることを許す係数であり、システムが安定化する機会が得られる。例えば、Margin1を1.15に設定することで、現在のビット使用度が割り当てられたビット割当値を15%だけ超えられる。

Margin2は、現在のビット使用度が割り当てられたビット割当値を下回ることを許す係数であり、システムが安定化する機会が得られる。例えば、Margin2を0.85に設定することで、現在のビット使用度が割り当てられたビット割当値を15%だけ下回ることができる。

50

StepUpPrcntは、量子化パラメータをどれだけ増やすかということに関係する定数である。例えば、StepUpPrcntを0.1に設定することができる。

StepDwnPrcntは、量子化パラメータをどれだけ減らすかということに関係する定数である。例えば、StepDwnPrcntを0.1に設定することができる。

【0134】

それぞれのGOVを符号化した後、実際のビット使用度 $B_{act}$ を事前に割り当てられたビット割当値 $B_{ass}$ と比較し、実際のビット使用度がビット割当値からある大きさ又はパーセンテージ(Margin3、Margin4)を超えて変化する場合に、Iフレームの量子化パラメータQP(QPI)は、必要に応じて、ある量(StepUp、StepDwn)又はパーセンテージだけ上又は下に調整される。量子化パラメータ調整プロセスは、以下の疑似コード例に従って実行することができる。

10

【0135】

```
If( $B_{act} > \text{Margin3} * B_{ass}$ )
```

```
    QPI=QPI+StepUp;
```

```
Else if( $B_{act} < \text{Margin4} * B_{ass}$ )
```

```
    QPI=QPI-StepDwn;
```

```
End If.
```

ここで、例えば、

20

```
Margin3=1.2
```

```
Margin4=0.8
```

```
StepUp=2
```

```
StepDwn=-2
```

である。

【0136】

QPIの初期値を10に設定し、P-VOPのQPの初期値を12に設定できる。I-VOPの量子化パラメータQPIが変更されると、続くPフレームのQP割り当ても変更される場合があることに注意されたい。

30

【0137】

一実施の形態では、エンコーダモジュール106Bは適応型動き変化検出を実行し、大きな伝搬誤りを効率よく低減することができる。特に、適応型イントラリフレッシュ(AIR)を使用し、Pフレーム内のマクロブロックの選択的イントラコーディングによってMEGデータストリーム内の誤り伝搬を低減する。そこで、AIRを使用することによって、フレームの検出された動き領域内でイントラコーディングしなければならないマクロブロックの数を決定するのを助けることができる。マクロブロックのイントラリフレッシュを実行することによって圧縮されたビットストリームの誤り耐性が高まるが、イントラコーディングされるマクロブロックの数を増やせば、これらのマクロブロックを符号化するために使用するビット数も増えることになる。さらに、固定ビットレートである場合、他の非イントラコーディングされたマクロブロックに対して量子化誤差が増えることになる。したがって、イントラコーディングするマクロブロックのパーセンテージ又は数を決定するために、帯域幅及びビット誤り確率(BER)が考慮される。

40

【0138】

さらに、エンコーダモジュール106Bは、オプションとして、循環イントラリフレッシュ(CIR)を使用して、各フレーム内の所定数のマクロブロック(MB)を符号化している。したがって、CIRは周期的イントラリフレッシュを実行して、誤り伝搬の発生の可能性を制限している。

50

## 【 0 1 3 9 】

一実施の形態では、循環イントラリフレッシュ (CIR) 及び適応型イントラリフレッシュ (AIR) は以下のように実行される。ユーザーがVOP内のイントラマクロブロックの数をエンコーダパラメータファイルで指定する。VOP内のイントラマクロブロックの数は、ターゲットビットレート、フレームレート、ビットバッファ使用度、チャンネル雑音フィードバック、及びその他の送信関連パラメータによって決まる。エンコーダモジュール106Bは、各マクロブロックの動き量を推定し、イントラモードで符号化する重い動き領域を選択して、誤り耐性を高める。推定の結果は、マクロブロックレベルでリフレッシュマップに記録される。

## 【 0 1 4 0 】

従来のリフレッシュマップ202の例は図2Aに示されている。エンコーダモジュール106Bは、リフレッシュマップを参照し、イントラモードで現在のVOPの指定されたマクロブロックを符号化するか否かを選択的に決定する。SAD (差の絶対値の和) と  $SAD_{th}$  とを比較することによって動きの推定を実行する。SADとは、現在のマクロブロックと前のVOPの同じ場所のマクロブロックとの間の差の絶対値を加算した値のことである。都合のよいことに、SADは、動き推定を実行したときにすでに計算されている。したがって、SAD計算をAIRプロセスの一部として繰り返す必要はない。 $SAD_{th}$  は、与えられたマクロブロックが動き領域であるか否かを判別する際のしきい値として使用される。現在のマクロブロックのSADが $SAD_{th}$  よりも大きい場合、このマクロブロックは動き領域とみなされる。

## 【 0 1 4 1 】

マクロブロックが動き領域としてみなされると、これは、所定の回数だけイントラモードで符号化されるまで、動き領域の候補として残る。図2Aに示されているリフレッシュマップ202の例では、この「所定の回数」の値は「1」に設定され、他の実施の形態では、この所定の回数は2又はそれ以上の値に設定することができる。図2Bのマップ204に示されているように、水平走査は、動き領域内でイントラモードで符号化される候補であるマクロブロックの間で決定のに使用される。

## 【 0 1 4 2 】

従来のAIR処理について、4つのサンプルの連続するVOPに適用されている図8Aを参照して、これから詳しく説明する。AIRリフレッシュレート、即ち、VOP内のイントラマクロブロックの固定された数が、あらかじめ決定されているのが好ましい。この例では、VOP内のイントラマクロブロックの数は「2」に設定される。

## 【 0 1 4 3 】

## [ 1 ] 第1のVOP - 図8A [ a ] 及び [ b ]

第1のVOPは、要素802、804を含むシーン変化フレームである。したがって、第1のVOP内の全てのマクロブロックが、図8A [ a ] に示されているように、イントラモードで符号化される。図8A [ b ] に示されているように、リフレッシュマップは「0」に設定されるが、0はイントラリフレッシュが実行されないことを示し、1はイントラリフレッシュを実行することを示しており、これは第1のVOPが前のVOPを参照せずに符号化されるからである。

## 【 0 1 4 4 】

## [ 2 ] 第2のVOP - 図8A [ c ] ~ [ f ]

第2のVOPはP VOPとしてインターコーディングされる。要素802、804は、1マクロブロック分下、1ブロック分右に移動している。イントラリフレッシュは、このVOPでは実行されないが、それは、図8A [ c ] に示されているように、リフレッシュマップ内の全ての値がまだ0であるからである。エンコーダモジュール106Bは、それぞれのマクロブロックの動きを推定する。与えられたマクロブロックのSADが $SAD_{th}$  よりも大きい場合、図8A [ e ] の斜線部に示されているように、与えられたそのマクロブロックは動き領域とみなされ、したがって、リフレッシュマップは、図8A [ f ] に示されているように更新され、動きマクロブロックに対応するリフレッシュマップエントリ

10

20

30

40

50

は 1 に設定される。

【 0 1 4 5 】

[ 3 ] 第 3 の V O P - 図 8 A [ g ] ~ [ k ]

要素 8 0 2、8 0 4 は、さらに 1 マクロブロック分下、1 ブロック分右に移動している。第 3 の V O P が符号化されると、エンコーダモジュール 1 0 6 B は図 8 A [ g ] で示されているリフレッシュマップを参照する。マクロブロックがイントラリフレッシュされるべきことをリフレッシュマップが示している場合、そのマクロブロックは、図 8 A [ h ] で「X」を含むマクロブロックにより示されているように、イントラモードで符号化される。イントラコーディングされたマクロブロックの対応するリフレッシュマップの値が、図 8 A [ i ] に示されているように、1 だけ減らされる。

10

【 0 1 4 6 】

減らされた値が 0 であれば、対応するマクロブロックは動き領域としてみなされない。先へ進むと、図 8 A [ j ] ~ [ k ] に示されているように、処理は実質的に第 2 の V O P の処理と同じであり、与えられたマクロブロックの S A D が  $S A D_{th}$  よりも大きい場合、その与えられたマクロブロックは動き領域とみなされる。リフレッシュマップは、図 8 A [ k ] に示されているように更新され、動きマクロブロックに対応するリフレッシュマップエントリは 1 に設定される。

【 0 1 4 7 】

[ 4 ] 第 4 の V O P - 図 8 A [ l ] ~ [ p ]

この処理は、第 3 の V O P と実質的に同じである。現在のマクロブロックがリフレッシュマップ内でそれと関連する 1 を持つ場合、これは、図 8 A [ m ] の「X」を含むマクロブロックにより示されているようにイントラモードで符号化される。イントラコーディングされたマクロブロックの対応するリフレッシュマップの値が、図 8 A [ n ] に示されているように、1 だけ減らされる。

20

【 0 1 4 8 】

減らされた値が 0 であれば、対応するマクロブロックは動き領域としてみなされない。与えられたマクロブロックの S A D が  $S A D_{th}$  よりも大きい場合、この与えられたマクロブロックは動き領域としてみなされる。リフレッシュマップは、図 8 A [ p ] に示されているように更新される。

【 0 1 4 9 】

他の実施の形態では、新規の拡張された A I R プロセスを次のように実行して、予測フレーム内のイントラコーディングされるべきマクロブロックを選択する。インターコード歪み値及びイントラコード歪み値が、インターコードビットレート及びイントラコードビットレートのように計算される。計算して求めたインターコード歪み値及びイントラコード歪み値の比較と、各マクロブロックのインターコードビットレート及びイントラコードビットレートの比較に基づいて、イントラコーディングされるべきマクロブロックに関する決定を下す。次に拡張された A I R プロセスについて詳述する。

30

【 0 1 5 0 】

イントラコーディングする予測フレームマクロブロックを選択するために、マクロブロックが失われた、又は破損した場合生じるであろうと予想される歪みを推定する。予測又はイントラコーディングされたマクロブロックについては、予測範囲内の参照マクロブロックがイントラコーディングされる場合、歪みは低減され得る。

40

【 0 1 5 1 】

図 8 B を参照すると、予測パスと共に再帰的追跡を使用して、マクロブロックの予想される歪みを決定することができる。8 0 4 B から 8 1 8 B、8 0 6 B から 8 2 0 B、8 2 0 B から 8 2 6 B、8 1 2 B から 8 2 2 B、8 1 4 B から 8 2 4 B、8 2 2 B から 8 2 8 B、8 2 8 B から 8 3 0 B への波線は、動きベクトル ( M V ) が、前のフレーム内のマクロブロックから現在のフレーム内のマクロブロックへの符号化されたビットストリームの一部であることを示している。8 0 2 B から 8 1 8 B、8 0 6 B から 8 2 0 B、8 1 0 B から 8 2 2 B、8 1 6 B から 8 2 4 B、8 1 8 B から 8 2 6 B、8 2 4 B から 8 2 8 B、及

50

び 8 2 6 B から 8 3 0 B への線など、傾斜した実線は、ゼロ動きベクトルであることを示し、そこでは失われた動きベクトルはゼロに設定される。ゼロ動きベクトルは、誤り条件でデコーダによって使用され、そこでデコーダは、誤り補正のため、回復不可能なマクロブロックを前のフレームからの対応するマクロブロックで置き換える。これは、「基本補正」と呼ばれる、利用可能な誤り補正方法の 1 つにすぎないことに注意されたい。時間的補正や補助的動きベクトル補正などの他の補正方法も実行できる。オプションとして、再帰的追跡を実行するとき、これらの他の誤り補正戦略の効果を別々に並行して考慮する。図 8 B において、「p」はパケット損失確率又はパケット損失率であり、 $q = (1 - p)$  である。

#### 【 0 1 5 2 】

与えられた現在マクロブロックを符号化しながら、エンコーダモジュール 1 0 6 B は前のフレーム上で動き検索を実行し、現在のマクロブロックとの一致度が最も高い、又はそうでなければ適切な予測フレームであると判断されるマクロブロックを見つける。斜線の入っていない円で示されているマクロブロック 8 0 2 B、8 0 6 B、8 1 0 B、8 1 6 B、8 1 8 B、8 2 0 B、8 2 6 B などの前のフレームから見つけられたこのマクロブロックは、予測マクロブロックと呼ばれる。動き検索が実行された後、残差が計算され、さらに離散コサイン変換 (DCT) を使用して符号化され、その後、選択した量子化ステップ又は量子化パラメータ (QP) を使用して量子化され、可変長コーディング (VLC) を使用してエントロピーコーディングされる。符号化されたビットストリームは、動きベクトル情報、残差のエントロピーコーディングされた量子化 DCT 係数、及び対応するヘッダ情報からなる。

#### 【 0 1 5 3 】

デコーダは、符号化されたビットストリームを受信すると、コーディングされた情報を処理し、マクロブロックを再構成する。パケット喪失又はその他の誤り条件のせいで、マクロブロックに対する情報が欠損している場合、デコーダが、上記の基本補正などの 1 つ又は複数の誤り補正方法を使用して、対応するマクロブロックを補正するのが好ましい。上記のように、マクロブロックが欠損している場合、基本補正により、前のフレームから同じ空間位置にマクロブロックをコピーする。これは、ゼロ動きベクトルとゼロ DCT 係数を受け取ることに相当する。

#### 【 0 1 5 4 】

どのマクロブロックがイントラコーディングされるべきかを決定するために、一実施の形態では、エンコーダシステム 1 0 0 に、対応するデコーダ回路を備えているので、デコーダのプロセスを模倣し、誤りが存在しない場合と、現在のマクロブロック (「MBC」) だけに影響を及ぼす単一誤りのような、誤り 1 つ又は複数の誤りが存在する場合との両方で、デコーダが再構成するものを再構成することができる。例えば、誤りのない再構成及び 1 つの誤りを仮定した再構成との差は、「補正誤り」又は EC と呼ばれる。EC は以下のように定義される。

$$EC = MBQ - MBC \quad \text{式 16}$$

ここで、MBQ は誤りのない再構成であり、MBC は単一誤りの再構成である。

#### 【 0 1 5 5 】

与えられたマクロブロックを次のフレームの予測マクロブロックとして使用する場合、与えられたマクロブロック上に存在する誤りは、動きベクトル及び次のフレームマクロブロックの DCT 係数にさらに誤りがなくても、予測のため与えられたマクロブロックを使用する次のフレーム内のマクロブロックに伝搬する。与えられたフレーム内のマクロブロックから次のフレーム内の別のマクロブロックに誤りが伝搬するメカニズムを「誤り伝搬モデル」と呼ばれる。

#### 【 0 1 5 6 】

垂直、又は水平方向、又は垂直方向及び水平方向の両方のいずれかで、予測に半ピクセル精度を使用するとき、誤り減衰が発生する。ローパスフィルタに匹敵する誤り減衰は、半ピクセル動きが使用されるときに適用されるピクセル平均化操作のローパス周波数特性

10

20

30

40

50

の結果生じる。したがって、エンコーダシステム 100B で計算した補正誤り EC が与えられると、水平方向 Ech / 2 の半ピクセル動きを介して伝搬する誤り、垂直方向 ECv / 2 の半ピクセル動きを介して伝搬する誤り、及び水平方向と垂直方向 Echv / 2 の半ピクセル動きを介して伝搬する誤りを決定することができる。

【0157】

半ピクセル補間が図 8D に示されており、これは、整数値のピクセル位置、水平方向の半ピクセル位置、垂直方向の半ピクセル位置、水平方向及び垂直方向の半ピクセル位置を示している。

【0158】

通常はピクセル値に適用される半ピクセル平均化フィルタを補正誤り EC に適用し、以下の 4 種類の伝搬誤り配列を定義することができる。 10

EC0 = EC

Ech / 2 = 水平半ピクセル動きによる誤り (図 8D のばつ「X」の位置で計算された値)

ECv / 2 = 垂直半ピクセル動きによる誤り (図 8D の菱形の位置で計算された値)

Echv / 2 = 水平及び垂直半ピクセル動きによる誤り (図 8D の正方形の位置で計算された値)

4 種類の誤り配列のそれぞれについて、0 平均の仮説のもとで誤差分散を近似する対応するエネルギーを計算する。

【0159】

これらの 4 つのケースに対する 4 つの誤差分散は、それぞれ次のように定義することができる。 20

(式 17)

$$\sigma_{Ec}^2, \sigma_{Ech/2}^2, \sigma_{Ec v/2}^2 \text{ 及び } \sigma_{Ech v/2}^2$$

その後、以下の 4 つの遷移又は強度係数を次のように定義することができる。

【0160】

【数 21】

$$\gamma_{Ec} = \frac{\sigma_{Ec}^2}{\sigma_{Ec}^2} - 1 = 0, \gamma_{h/2} = \frac{\sigma_{Ec}^2}{\sigma_{h/2}^2} - 1, \gamma_{v/2} = \frac{\sigma_{Ec}^2}{\sigma_{v/2}^2} - 1 \text{ および } \gamma_{hv/2} = \frac{\sigma_{Ec}^2}{\sigma_{hv/2}^2} - 1 \quad \text{式 18}$$

これらは、現在のマクロブロックの動きの可能な 4 つのケースに対応する。これらの量は、テーブル、ファイル、又はその他のレコード内に、現在のマクロブロック (mx, my)、初期誤りエネルギー  $\sigma_{Ec}^2$ 、及びコーディングモード (イントラ/インター) を符号化するために使用される動きベクトルとともに保存される。 30

【0161】

半ピクセル水平及び垂直伝搬強度は以下の式で近似することができる。

$$\gamma_{hv/2} = \gamma_{h/2} + \gamma_{v/2} \quad \text{式 19}$$

半ピクセル水平及び半ピクセル垂直動きの遷移係数を使用すれば、これによって、半ピクセル水平及び垂直伝搬強度又は遷移係数の計算に要する計算時間を短縮し、リソースを減らすことができる。さらに、伝搬強度は正でなければならない。したがって、負の伝搬強度の結果は丸めるか、又は 0 に設定される。 40

【0162】

図 8E に示されているように、動きベクトル MV は、16 ピクセルの行と列からなるグリッドに揃えられている現在のフレーム Frame<sub>n</sub> 内のマクロブロックを、必ずしも同じグリッドに揃えられていない予測フレーム Frame<sub>n-1</sub> 内の 16 x 16 ピクセルにマッピングすることができる。実際、図 8E に示されているように、Frame<sub>n</sub> 内のマクロブロックは、予測フレーム内 Frame<sub>n-1</sub> の最大 4 つまでのマクロブロックの一部にマッピングされ得る。

【0163】

現在のフレーム内のマクロブロックの予測に使用される前のフレームからの可能な 4 つの 50

マクロブロックのうちの1つ又は複数に存在する誤りは、現在のフレーム内のマクロブロックに反映される。誤りの関係はオーバーラップ面積と比例する。例えば、誤りの関係は、オーバーラップするピクセルの数に比例又はそれに基づくことができる。したがって、現在のフレーム内のマクロブロック毎に、インターモードでマクロブロックを符号化するときに使用することになる最大4つまでの予測マクロブロックが識別される。対応する動きベクトル情報を使用して、オーバーラップ面積を決定し、その面積に等しい又は関係する重み係数を使用して、次の式で定義されているように、オーバーラップ面積をマクロブロック面積全体に正規化する、例えば、256(16×16)に正規化する。

【0164】

【数22】

$$\sigma_v^2(i,j) = w(i,j) \frac{\sigma_u^2(i)}{1+\gamma_{i,j}}$$

これは、前のフレーム内のブロック*i*による現在のフレーム内のマクロブロック*j*上の予想歪みを推定する。 $\gamma_{i,j}$ は、水平、垂直、又はその両方の方向に沿う動きのタイプ(半又は整数ピクセル動きベクトル)に応じて、すでに定義されている遷移係数 $\gamma_{EC}$ 、 $\gamma_{h/2}$ 、 $\gamma_{v/2}$ 、及び $\gamma_{hv/2}$ のうちの一つであり、 $w(i,j) = w1 \cdot h1 / 256$ はマクロブロック*j*とマクロブロック*i*の間のオーバーラップ面積( $w1 \times h1$ )に関係する重み係数であることに注意されたい。項 $\sigma_u^2(i)$ は、マクロブロック*i*の補正誤り $\sigma_{EC}^2$ である。

【0165】

図8Bを参照すると、現在のFrame<sub>n</sub>内のマクロブロック830Bから始まり、Frame<sub>n-1</sub>内に、マクロブロック830Bを再生成するためにデコーダによって使用される2つのマクロブロック、即ち、通常の復号化のために使用されるマクロブロック826B、及び補正に使用されるマクロブロック828Bがある。Frame<sub>n-1</sub>内のマクロブロック826B、828Bはそれぞれ、上記のように、最大4つまでの揃えられたマクロブロックに対応する。同じ「復号化又は補正」方法をFrame<sub>n-1</sub>内の2つのマクロブロック826B、828Bに対して再帰的に適用し、Frame<sub>n-2</sub>内の4つのマクロブロック818B、824B、822B、820Bを見つけ、その後、8つのマクロブロック802B、804B、806B、808B、810B、812B、814B、816Bを含むFrame<sub>n-3</sub>に到達できる。ある一連の誤り/パケット損失が送信時に発生した場合、Frame<sub>n-3</sub>内の8つのマクロブロックのそれぞれが、現在のマクロブロックでFrame<sub>n</sub>に出現する可能性がある。これら8つの経路のそれぞれの確率は対応する分岐確率( $p/q$ )の積であり、ここで

はパケット損失確率であり、 $q = (1 - p)$ である。マクロブロック830Bへの特定の経路が発生する確率を求めるために、その経路に沿って

及び $q$ の値を乗算する。したがって、1つの行内で2つのパケット損失が発生する場合のような確率 $p^2$ を持つ経路と、812B~830Bにより定義された確率 $p^3$ である経路とが存在する。

【0166】

誤りの確率が比較的小さい(例えば、0.1)と仮定すると、確率 $p^2$ 又は $p^3$ などの高次( $p$ に関して)の経路は無視することができ、図8Bを図8Fに示されている経路に簡略化することができる。図8Bの経路の簡略化は、補正に使用するマクロブロックはそれ自体破損していない、即ち、2つのマクロブロックの間のある1つの経路における複数の誤り/パケット損失の確率を無視できるという仮定に基づいている。この仮定は常に正しいわけではないが、多くの場合正しい。

【0167】

このように簡略化されたマクロブロックの関係に基づき、上記の伝搬モデルを使用して、Frame<sub>n</sub>の現在のマクロブロックに対する予想歪みを推定することができる。予想歪みは、次の式で定義される。

$$D(n) = p^2 \sigma_{EC}^2(n) + q D'(n-1) \quad \text{式 20}$$

ここで、 $D'(n-1)$ は、Frame<sub>n-1</sub>からFrame<sub>n</sub>までの可能な半ピクセル動き

10

20

30

40

50



を考慮するために遷移係数によって修正された  $Frame_{n-1}$  の参照マクロブロックの予想歪みである。  $Frame_{n-1}$  の参照ブロックに関してこの式を展開し、予想歪みを次のように定義する。

【0168】

【数23】

$$D(n) = p\sigma_{EC}^2(n) + q \left( p \frac{\sigma_{EC}^2(n-1)}{1+\gamma_{(n-1,n)}} + qD''(n-2) \right) \quad \text{式21}$$

$\gamma_{(n-1,n)}$  は、  $Frame_{n-1}$  から  $Frame_n$  までの動きベクトルに依存する、  $Frame_{n-1}$  内の参照マクロブロックの4つの遷移係数 ( $\sigma_{EC}$ 、  $\sigma_{h/2}$ 、  $\sigma_{v/2}$ 、 及び  $\sigma_{hv/2}$ ) のうちの1つである。上で述べたのと同様に、  $D''(n-2)$  は、  $Frame_{n-2}$  から  $Frame_{n-1}$  及び  $Frame_{n-1}$  から  $Frame_n$  までの可能な半ピクセル動きを考慮するために、遷移係数によって修正された  $Frame_{n-2}$  の参照マクロブロックに関する予想歪みである。この項をさらに展開すると、予想歪みは次のように定義される。

【0169】

【数24】

$$D(n) = p\sigma_{EC}^2(n) + q \left( p \frac{\sigma_{EC}^2(n-1)}{1+\gamma_{(n-1,n)}} + q \left( p \frac{\sigma_{EC}^2(n-2)}{1+\gamma_{(n-2,n-1)} + \gamma_{(n-1,n)}} + qD'''(n-3) \right) \right) \quad \text{式22}$$

$Frame_{n-3}$  が I フレームであるか、又はフレームバッファが3フレームに制限又は制約されている場合、  $D'''(n-3)$  は0に等しい。そうでない場合、同じ手順を前のフレームのマクロブロックに再帰的に適用する。同様に、再帰的プロセスの実行中にイントラマクロブロックに遭遇した場合、歪みは  $p^2\sigma_{EC}^2$  に等しいと仮定するが、それは、動きベクトルがなく、したがって誤り伝搬項がないからである。

【0170】

上の歪み方程式では、各フレーム内の最大4つの予測マクロブロックのうちのそれぞれの1つからの寄与分を全て加算し、  $Frame_n$  内のターゲットマクロブロックとこれらのマクロブロックとの間でオーバーラップする面積に関係する、対応する重み係数を掛ける。

【0171】

前のフレームのそれぞれのマクロブロックについて格納されている情報を利用して、現在の  $Frame_n$  のそれぞれのマクロブロックの予想歪みを計算することができる。この予想歪みは送信誤りによるものであり、それぞれのマクロブロックの量子化による歪みとは相関しないことに注意されたい。したがって、予想歪み項を量子化誤りに加えて、マクロブロック毎に全歪みを決定する必要がある。この全歪みのことを、「全インターモード歪み」又は  $D_{TINTER}$  と呼び、マクロブロックのインターモード符号化に関係する。

【0172】

イントラモードでは、予想歪みは次のように第1の項に簡略化される。

$$D(n) = p^2\sigma_{EC}^2(n) \quad \text{式23}$$

これは、  $D_{INTRA}$  とも呼ばれる、マクロブロックの「全イントラモード歪み」を得るために、対応するイントラモード量子化歪みに加える必要がある予想誤り歪みを反映している。

【0173】

それぞれ、  $R_{TINTER}$  及び  $R_{TINTRA}$  と呼ばれるインターモード符号化及びイントラモード符号化に、一定数のビットが必要である。これらのビットレートの差  $R = R_{INTRA} - R_{INTER}$  の差を、全歪みの差  $D = D_{INTRA} - D_{INTER}$  とともに使用することにより、最良のコーディングモードを選択することができる。

【0174】

10

20

30

40

50

エンコーダに十分なリソースと能力があれば、以下の式で与えられる、各マクロブロックのコスト関数を評価するために使用される最適な重み係数 を判別する必要のある完全な歪み率最適化を実行できる。

$$C = D + R \quad \text{式 2 4}$$

その後、以下のイントラ/インター決定規則を求める。

【0175】

次の場合にイントラモードを選択する。

- ・  $R = 0$  ならば  $D < 0$
- ・  $R > 0$  ならば  $D / R < -$
- ・  $R < 0$  ならば  $D / R > -$

そうでなければインターモードを選択する。

【0176】

最適なパラメータの決定は、オプションとして、可能な全てのQPと の組み合わせを試すことによって達成されることに注意されたい。目的のビットレートよりも低い、又はオプションとして目的のビットレートに等しいビットストリームを出力する全ての組み合わせの中から歪みが最小である特定の組み合わせを選択する。

【0177】

その代わりに、エンコーダシステム100は最初に、上記ケース(  $R = 0$  ならば  $D < 0$  )のうち、 の全ての値に適用できる第1のケースを満たすマクロブロックと、さらに、自動的に第3のケースに適用される、  $R < 0$  ならば  $D / R \geq 0$  という条件を満たす全てのマクロブロックを選択することができる。次に、  $R > 0$  のマクロブロックを、ひとまとめにし、比  $D / R$  の昇順に順序付ける。同様に、  $R < 0$  のマクロブロックを、ひとまとめにし、同じ比  $D / R$  の降順に順序付ける。

【0178】

これは、各マクロブロックに対する比  $D / R$  の値を表している図8Gに示されているグラフによって示されており、「x」が  $R > 0$  のマクロブロックを示し、「o」が  $R < 0$  のマクロブロックを示す。

【0179】

最大の負の値又は対応する定義済み基準を満たす負の値を持つ「x」、及び最小の負の値又は対応する定義済み基準を満たす負の値を持つ「o」が、イントラリフレッシュの候補として選択される。正の値を持つ「o」はすでにイントラコーディングとして選択されており、正の値を持つ「x」は自動的にイントラコーディングされるので完全に除外されることに注意されたい。実験結果から、「x」で示されている  $R > 0$  であるマクロブロックが最も一般的なものであることが分かるが、一般に、コストをビット単位で考えた場合に、イントラモードのコストがインターモードのコストに比べて高いからである。したがって、オプションとして、  $R > 0$  であるマクロブロックのみがイントラリフレッシュの対象とみなされる。イントラモードで符号化する追加マクロブロックの数を指定する、いわゆるリフレッシュレートは、最終的に選択される候補マクロブロックの数を指定するものである。

【0180】

リソースをあまり使わないプロセスでは、送信誤りによる各マクロブロックの予想歪みを計算するが、量子化誤りは無視するか、又は除外する。次に、イントラモードとインターモードとの予想歪みの差を、イントラコーディングするマクロブロックを選択する基準として使用でき、この基準に従って順序付けることによってマクロブロックが選択される。

【0181】

よって、適応型イントラリフレッシュ(AIR)を使用することによって、フレームの検出された動き領域内でイントラコーディングされなければならないマクロブロックの数を決定することを補助することができる。ユーザーによってセットされ、エンコーダモジュール106Bによって読み込まれるAIRビットを使用して、エンコーダパラメータファイル内でAIRを有効、無効に設定できる。AIRが有効であれば、ユーザーはさらに、

10

20

30

40

50

別のパラメータであるAIRリフレッシュレートを指定する。AIRリフレッシュレートにより、1つのフレームの検出された動き領域内でイントラコーディングされなければならないマクロブロックの数が決まる。適応型動き変化の検出を利用すると、動き領域に誤りが発生した場合でも、大きな伝搬誤りを効率よく低減することができる。

【0182】

図8Hは、E-AIRプロセスの一実施の形態を示している。状態802Hにおいて、Frame(n)の注目している現在のマクロブロックに対する1つ又は複数の動きベクトルを受信する。状態804Hにおいて、この動きベクトルを使用して、現在のマクロブロックを決定する際に使用される前のFrame(n-1)からのマクロブロックを見つける。状態806Hにおいて、領域又はピクセル数に関して、Frame(n-1)内の見つかったマクロブロックのそれぞれをどれだけ使用して現在のマクロブロックを生成するかを決定する。状態808Hにおいて、オーバーラップ重み付け

10

【数25】

$$\left( w(i,j) \frac{1}{1+\gamma_{i,j}} \right)$$

を含んで、誤差分散 ( $\sigma_{Ec}^2$ 、 $\sigma_{Ech/2}^2$ 、 $\sigma_{Ecv/2}^2$ 、 $\sigma_{Echv/2}^2$ ) を計算する。状態808Hにおいて、誤差分散に基づいて伝搬強度遷移量を計算する。状態812Hにおいて、Frame(n)のイントラ誤り歪み  $D_{INTRA}$  を計算する。状態814Hにおいて、Frame(n)のインター誤り歪み  $D_{INTER}$  を再帰的に計算する。上記のように、再帰的計算は、誤りがFrame(n)に伝搬する、Frame(n-1)、Frame(n-2)、...などの、前のフレームからの誤り歪み及び量子化歪みを含む可能性がある。繰り返しは所定の回数フレーム生成に制限され、全て又は所定量のフレームバッファが使用されるか、又はイントラフレームに達したときに繰り返しが停止する。

20

【0183】

状態816において、 $D_{INTRA}$  と  $D_{INTER}$  との差を求めるか、又は別の手段で、 $D_{INTRA}$  と  $D_{INTER}$  とを比較することによって値  $D(\Delta D)$  を計算する。状態818Hにおいて、Frame(n)のイントラコーディング及びFrame(n)のインターコーディングを行うための、ビット量又はビットレート  $R_{INTRA}$  及び  $R_{INTER}$  をそれぞれ決定する。状態820Hにおいて、差  $\Delta R$  を計算することによって  $R_{INTRA}$  と  $R_{INTER}$  との比較を行う。状態822Hにおいて、説明した基準を用いて  $R(\Delta R)$ 、 $D(\Delta D)$ 、及び  $\lambda$  (Lambda) に基づいてイントラコーディングするかインターコーディングするかの決定を下す。それとは別に、 $D(\Delta D)$  が設定されているマクロブロックをイントラコーディングするように選択することもできる。例えば、Frame(n)のマクロブロックの全てについて  $D(\Delta D)$  を計算した後、最大の  $D(\Delta D)$  を持つ2つのマクロブロックをイントラコーディングする。マクロブロックのイントラコーディングの選択はさらに、この場合、 $Cost = Rate + \lambda * D$ 、又は  $D + \lambda * R$  としたコスト計算に基づくこともでき、及び最高の  $N (= AIR \text{レート})$  を選択できる。

30

【0184】

図8Cは、循環イントラリフレッシュ、トレース802Cの使用と、すぐ上で説明した拡張されているAIR方法、トレース804Cの使用とを比較した実験結果をまとめたものである。全体的な利得は、PSNRでほぼ1dBである。一実施の形態では、付加的な計算負荷は約10%である。

40

【0185】

AIRをより効果的に利用するために、従来の循環イントラリフレッシュ(CIR)をAIRと組み合わせる。VOP内のイントラリフレッシュマクロブロックの数は、 $AIR\_refresh\_rate$  と  $CIR\_refresh\_rate$  の和として定義される。 $AIR\_refresh\_rate$  マクロブロックは、AIRモードで符号化され、 $CIR\_refresh\_rate$  マクロブロックは、従来のCIRモードで符号化される。

50

これらの値は、エンコーダパラメータファイル内でユーザーが定義することができる。チャンネル品質が低下する場合、高いC I R及びA I Rレートを割り当てるのが好ましい。さらに、Iフレーム間の距離が大きい場合、高いC I R及びA I Rレートを割り当てるのが好ましい。これらのレートは、誤り耐性とコーディング効率とのトレードオフの関係を改善するために、チャンネル条件ならびに誤りコーディングパラメータに応じて変えるのが好ましい。

【0186】

エンコーダパラメータファイルは、上記のパラメータなど、さまざまな符号化パラメータを指定する。エンコーダパラメータファイルは、上記の前処理プロセスによって決定される符号化タイプを指定するフレームタイプファイルを読み込むことにより、前処理モジュールの出力とともに使用することができる。例えば、エンコーダパラメータファイルは、A I R、C I R、及びS M Cの有効化/無効化、A I R及びC I Rリフレッシュレートの指定、各々のシーン、G O V又はG O Pの先頭に2つのIフレーム又はを含むことを有効又は無効にするために使用されるフラグを設定するフィールドを含む。

10

【0187】

エンコーダパラメータファイルは、次のパラメータ又はフィールドを含んでいる。

【0188】

【表2】

バージョン	バージョン番号および/または名前	
Source.Width	フレーム幅を指定する	
Source.Height	フレーム高さを指定する	
Source.FirstFrame	符号化する最初のフレームを指定する(0から数える)	
Source.LastFrame	符号化する最後のフレームを指定する	
Source.Directory	後ろに「\」を付けないシーケンスの元のソースの格納先ディレクトリ	
Source.SamplingRate	このサンプリングレートパラメータに基づいて元のソースのサブサンプリングを行うことができる	
Output.Directory.Bitstream	出力ビットストリームディレクトリ	10
Output.Directory.DecodedFrames	エンコーダからの再構成されたフレームの出力先ディレクトリ(エンコーダはさらに復号化も実行する)	
RateControl.Type[0]	レート制御の種類-「None」(一定のQPを維持)、「MP4」(IMP4用)、「TM5」(Test Model 5用)のうちの1つ	
RateControl.BitsPerVOP[0]	シーケンス全体に対するビット割当値	
Quant.Type[0]	「H263」、「MPEG」のうちの1つ。低ビットレート通信では、H263が好ましい。	
GOV.Enable[0]	GOVヘッダの有無	
GOV.Period[0]	GOVヘッダ間のVOPの数	
Texture.QuantStep.IVOP[0]	I-VOPに対する量子化パラメータ(QP)。レート制御の影響を受けない。	
Texture.QuantStep.PVOP[0]	レート制御が無効になっている場合のP-VOPに対するQP	20
Texture.QuantStep.BVOP[0]	レート制御が無効になっている場合のB-VOPに対するQP	
Motion.PBetweenICount[0]	複数のシーンがあり、フレームタイプファイルが存在すれば、エンコーダはこのパラメータを無視する。そうでなければ、GOPの長さはソースのサブサンプリングの前に指定される。負の値は、シーケンス全体で1つGOPがあることを意味する。	
Motion.ReadWriteMVs[0]	「Off」、「Read」、「Write」の1つ	
Motion.ReadWriteMVs.FileName[0]	ファイルからのMVの読み込み/ファイルへの書き込みのためのファイル名	
ErrorResil.RVLC.Enable[0]	RVLCを有効または無効にする-0:無効、1:有効	
ErrorResil.DataPartition.Enable[0]	データ分割を有効または無効にする	
ErrorResil.VideoPacket.Enable[0]	ビデオパケット(VP)内の再同期マーカの導入を有効または無効にする	
ErrorResil.VideoPacket.Length[0]	VideoPacketが有効な場合、ビット単位のVPのサイズターゲットビットレートに基づいて選択する	30
ErrorResil.SMC.Enable	第2の動き補正(SMC)の有効/無効を設定する。1であれば1つのみのPVOPに関して、2であれば全てのPVOPに関して。	
ErrorResil.AIR.Enable	適応型イントラリフレッシュ(AIR)の有効/無効を設定する。	
ErrorResil.CIR.Enable	循環イントラリフレッシュ(CIR)の有効/無効を設定する。	
ErrorResil.AIR.Rate	AIRレートの指定のため追加。例えば、AIRレートとして2を使用できる。	
ErrorResil.CIR.Rate	CIRを有効にしたとき(上を参照)、CIRレート(マクロブロック数/VOP)を指定する。例えば、CIRレートとして2を使用できる。	
ErrorResil.2I.Enable	誤り耐性を高めるための各GOV内の二重フレームコーディングを有効/無効にするために追加。	40

表1 エンコーダパラメータ仕様

誤り耐性をさらに高めるために、エンコーダモジュール106Bではヘッダ拡張コード(HEC)を、連続するビデオパケットの全てのパケット又は全てのビデオパケットに挿入し、従来のエンコーダの場合のようにVOPヘッダに続く第1のビデオパケット上だけで挿入するのではない。この方法だと、パケットが失われたり破損したりしても、後続のパケットを復号化して使用できるのでより好ましい。さらに、代表的な従来のデコーダであっても、追加のHECの使用はMPEG-4ビットストリーム構文と互換性があるため、HECの拡張使用する場合にも対応できる。シーケンス情報を含むヘッダを全てのパケットに追加することで、オーバーヘッドは1パケット当たり約40ビット、又は約0.2%

しか増えないが、復号化の改善は顕著である。

【0189】

さらに、誤り耐性を高めるために、オプションとして二次動き補正(SMC)が提供される。SMCプロセスは、エンコーダモジュール106Bで実行され、補助動きベクトルを生成し、それぞれの予測フレームを2つの先行するフレームから別々に予測することができる。図5のシーケンス502は、SMCプロセスを示しており、k番目のフレームは(k-1)番目のフレーム及び(k-2)番目のフレームの両方の動きベクトルを持つ。したがって、(k-1)番目のフレームの動きベクトルが破損している場合、又は(k-1)番目のフレーム自体が破損している場合であっても、k番目のフレームを(k-2)番目のフレームから、対応する動きベクトルを使用して予測することができる。したがって、(k-2)番目のフレームからk番目のフレームまでの二次動きベクトルとも呼ばれる冗長動きベクトルを挿入することによって、デコーダ側のシーンイメージ品質が送信誤りから適切に保護される。例えば、送信中にk番目のフレームの全ての情報が破損した場合でも、SMCを使用して、図5にシーケンス504として示されているように、後の予測で使用されないようにk番目のフレームを除外することによって誤り伝搬を効果的に抑制することができる。

10

【0190】

SMCを実行するために、フレームバッファをエンコーダモジュール106Bに含め、時刻(t-2)のときにすでに復号化されているフレームを格納する。このすでに復号化されているフレームを使って、二次動きベクトルを計算する。一実施の形態では、これらの冗長動きベクトルは、残差を生成するためにエンコーダでは使用されない。デコーダは、送信時にビットストリームが破損しているときに、及び一次動きベクトル又は対応するフレームが破損しているときに、二次動きベクトルを使用する。そうでない場合、デコーダで二次動きベクトルを使用する必要はない。オプションとして、完全な無制限の動き検索を実行して、これらの二次動きベクトルを決定することができる。フレーム(t-2)と(t-1)との間及びフレーム(t-1)と(t)との間の動きに関する情報を組み合わせ、これらの二次動きベクトルを推定することができる。このような理由から、後述するように、シーン変化後の最初のP-VOP又はそれぞれのP-VOPに関する「ユーザーデータメカニズム」を介して、SMCデータをオプションとして含めることができる。SMCを最初のP-VOPのみに用意する利点は、誤りがないときに帯域幅が無駄にならず、コーディング効率が向上するという点である。しかし、全てのP-VOP又は多数のP-VOPに対してSMCを用意すると、特に重大な誤り状態の場合に、デコーダの堅牢性と復号化能力が高まる。

20

30

【0191】

P-VOP毎に「ユーザーデータ」又は「SMC」ビデオパケットと呼ばれる、追加のビデオパケットを使用して、これらの二次動きベクトルを送信する。このパケットには、標準動きベクトルの場合と同じ予測方法で、同じ可変長コードを使用して、現在のP-VOPの各マクロブロック又は選択されたマクロブロックの動きベクトルが含まれる。HECがこの特別なSMCビデオパケットに含まれており、このP-VOPに対する他のパケットが失われた場合でもSMCビデオパケットを復号化することができる。一実施の形態では、このパケットは、ビットストリーム内の各P-VOPの最後に配置されている。ユーザー側で、エンコーダパラメータファイル内の対応するオプションを1又は0に設定することによって、SMCの使用を有効又は無効にすることができる。図6は、ビットストリーム内のパケットの相対位置を示す、SMCパケット602を含むパケット化されたビットストリームの例を示している。

40

【0192】

オプションとして、SMCパケットをMPEG-4構文に適合させるために、将来MPEG規格委員会などによって割り当てられる他の一意的識別子コードを含む、いわゆる「ユーザーデータ開始コード」(16進数コードB2)などをHEC及び動きベクトル情報の前に付ける。ユーザーデータ開始コードが入っていると、二次動きベクトルを使用できな

50

い標準デコーダは、このコードの後からビットストリーム内の次の開始コード、即ちこの場合VOP開始コードまでの全てのビットを無視する。一実施の形態では、エンコーダは、SMCユーザーデータ拡張と他の人々が同じ規則に従ってビットストリームに含めることに決めたデータとを混同しないように、一意的な16ビット識別子を含んでいる。

【0193】

誤り耐性をさらに高めるために、図7に示されているように、連続する2つのフレームのうち第2のものが、シーン変化フレームでなく、通常であればインターコーディングされる十分に低い相対動きを持っていても、1つのシーン変化で、2つの連続するIフレームが挿入される。即ち、シーン変化フレーム702がイントラコーディングされた後、次のフレーム704が自動的にイントラコーディングされる。したがって、1つのIフレームが失われたとしても、デコーダは、フレーム702ならびにフレーム704から予測される予測フレーム706、708、710、712を復号化する動作を中止しない。第2のIフレームの後のフレームは、Pフレーム又はBフレームなどのイントラコーディングフレームとして符号化されていてもよい。

10

【0194】

2つの連続するIフレームを使用すると、SMCの効率を落とすことなく、他のシーンのシーン内容を使用して現在のシーン内のフレームを予測するという事態をきちんと防止することができる。シーン内の最初の2つの連続するフレームはイントラコーディングされるので、一次動きベクトルも二次動きベクトルもIフレームには挿入されない。

【0195】

連続するIフレームを挿入する動作は、シーン変化フレーム及び次のフレームの両方をイントラコーディングフレームとして指定できる前処理モジュール102Bの制御下に置くことができる。その代わりに、エンコーダモジュール106Bは、前処理モジュール102Bによってイントラコーディングフレームとして指定されているフレームの後のフレームを自動的にイントラコーディングすることもできる。

20

【0196】

2つの連続するIフレームを挿入するとビット割当値が増大し、ある送信ビットレートではコーディング効率が低下するが、誤りの発生しやすい環境では、そうすることで誤り耐性が高まるので、この非効率性は補ってあまりある。「連続Iフレーム」フラグがエンコーダパラメータファイルに用意されており、これがSMCフラグとは独立のものであってもよい。都合のよいことに、SMCモードがオンでなくても、又はシーン変化の後の(二重)Iフレームに続く最初のP-VOPだけについてオンであっても、2つの連続するIフレームが各シーンの先頭に存在することを利用して、デコーダが送信誤りをより効率的に補正するようにできる。

30

【0197】

適応型イントラリフレッシュ(AIR)は、オプションとして、エンコーダモジュール106B側でSMCの副産物としてサポートすることができる。このモードは、最初のP-VOPだけ、又は全てのP-VOPについてSMCを選択したときに有効になり、イントラモードで、フレーム(t-1)及び(t-2)からの予測マクロブロックとして、MAD距離測定基準で測定された場合に著しく異なる2つのマクロブロックを持つマクロブロックを符号化する。例えば、しきい値は20である。よって、現在のフレーム内の与えられたマクロブロックについて2つの予測マクロブロックの間のMADが20よりも大きければ、このマクロブロックはイントラコーディングされる。

40

【0198】

エンコーダモジュール106Bはさらに、動き検出、残差計算などの一般的なエンコーダ機能も実行する。エンコーダ出力は、後で送信するために格納しておくことも、又は実質的にリアルタイムで、適切なデコーダを備える携帯電話などの受信端末に送信することもできる。

【0199】

誤り耐性を高めるために、一実施の形態では、Intra\_dc\_vlc\_thrを「0

50

」に設定し、フレーム又はVOP内のDC VLCを使用して、全てのDC係数をコーディングする。さらに、ac\_pred\_flagを全てのイントラマクロブロックに対して無効にすることができる。これらのオプションは両方とも、構文によって許可され、したがって、標準のデコーダによってサポートされ、送信で誤りが発生した場合でも品質を高められる。このような改善は、PSNRで0.1~0.2 dBのオーダーである。特に、データ分割が有効になっている場合、「INTER VLC」テーブルとして知られているものを使用して、イントラマクロブロックの各8x8 DCTブロックのDC係数を63個のAC DCT係数と一緒にコーディングするか、又は「INTRA VLC」テーブルとして知られているものを使用して、別々にエンコーディングすることができる。

【0200】

10

INTRA VLCテーブルを使用すると、次のように、DC係数に関するビットを、残り63個のAC係数に関するビットと分けることができる。

【0201】

マクロブロックを全てイントラコーディングするI-VOPに対しては、DCデータビットはDCマーカ(DCM)の前にヘッダビットと共に配置され、AC係数のデータビットはDCマーカの後に配置される。

【0202】

P-VOPに対しては、DCデータビットは動きマーカ(MM)の直後に、他の不可欠な又は重要な情報とともに配置され、AC係数に関するデータビットが続く。

【0203】

20

DC及びAC情報をこのように分離することで、誤り耐性が高まるが、これは、破損したパケットに関しても、DCM/MMマーカが復号化プロセスで正しくヒットしていれば、DCデータが復号され、信頼できるからである。さらに、P-VOPに関して、リバーシブル可変長コード(RVLC)の順方向/逆方向復号化により、少なくとも1つの適切な最初の部分を明らかにできる。

【0204】

DCデータを一緒にコーディングするか、又はAC係数と別にするかを制御するために、「intra\_dc\_vlc\_thr」というフラグが、QP値に従って各マクロブロックをマッピングするVOP毎に使用される。この値を0に設定することは、QPにかかわらず、全てのマクロブロックがINTRA DCテーブルを使用し、DCデータをAC係数から分離することを意味する。これは、MPEG-4の標準構文要素であり、標準デコーダによってサポートされている。

30

【0205】

ac\_pred\_flagは、INTRAマクロブロックの特定のブロックについて、最上行及び第1列のDCT係数を、隣接するブロックと独立に、又は区別して、コーディングするか否かを示すもう1つのオプションである。誤り耐性を高めるために、ac\_pred\_flagを0に設定するのが好ましい。

【0206】

一実施の形態では、前方誤り訂正(FEC)を使用してソースレベルで誤り訂正をサポートしている。特に、リードソロモン(Reed-Solomon)を含むBose-Chaudhuri-Hocquenghem(BCH)コードがサポートされている。当業者によく知られているように、BCHは、巡回冗長度符号コード(Cyclic Redundancy Code)に基づく誤り検出及び訂正方法である。kを情報ビットの数として、正の整数m、 $m > 3$ 、 $t < 2^{m-1}$ について、ブロック長nが $2^m - 1$ に等しく、 $n - k < mt$ のパリティ検査ビットを持つ2進数のBCHコードがある。BCHコードは、少なくとも $2t + 1$ の最小距離を持つ。それぞれの2進BCHコード(n, k, t)は、最大tビットまでの誤りを訂正でき、したがって、t誤り訂正コードとも呼ばれる。

40

【0207】

ブロックサイズは異なってもよい。一実施の形態では、ブロックサイズとして511が使用される。FECは基本ビデオビットストリーム(elementary video stream)のバ

50



ケタイザ (packetizer) レベルで実行され、これはソースレベル誤り訂正とみなすことができる。対照的に、チャンネルレベル誤り訂正では、多重化後にビットレベルで冗長性が入り込む。

#### 【0208】

FECは、ある程度のビット割当値と引き換えに、著しい誤り耐性を持つ。図11は、前方誤り訂正のオーバーヘッドと平均のBER訂正能力を対比したグラフの例を示す図である。図に示されているように、FEC冗長性と誤り訂正能力との間には密接な関係があり、誤り耐性の強力な指標となっている。少なくとも予想BERの2倍に対応するのが好ましい。

#### 【0209】

本発明の一実施の形態では、FECを全てのパケットデータに適用する代わりに、従来のアプローチに比べてより効率的で、生成される誤り訂正ビットが少ないプロセスを実行するが、それでも、著しい誤り訂正能力を持つ。FECプロセスの一実施の形態では、オプションとして、パケットの選択した部分についてのみFECビットを生成し、特に、デコーダによってフレームシーケンスを再生する目的に関して、より本質的又は重要であると考えられる部分についてFECビットを生成することができる。さらに、FECプロセスは体系的コードを持つ、即ち、FEC訂正又はパリティビットは元のコーディングされていないデータビットと分離される。したがって、全てのFECビットが失われても、パケットの元の選択された部分はまだ潜在的に復号化可能である。さらに、一実施の形態では、FECデータは符号化され、後述するようにMPEG-4に準拠する方法で送信される。したがって、FECデータを処理する機能のないデコーダがFECパケットを受信した場合でも、そのデコーダは、フレーム動き及びテクスチャデータを処理することができる。

#### 【0210】

特に、FECは、動きベクトル、DC係数、及びヘッダ情報などの重要なデータに効率良く適用され、重要でない又はあまり重要でないデータに対してはFECビットを生成しない。このより重要なデータは、パケット再同期フィールドと動きマーカーとの間に配置することができる。特に、指定されたフレーム又はVOPについては、FECコーディングをターゲットとする選択されたビットが、他のフレームパケットのビットと連結され、その連結されたビットに対してFECコードビットが生成される。

#### 【0211】

一実施の形態では、与えられたフレーム又はVOPに対して、フレームデータと同じ1つのパケット又は複数のパケットにFECビットを含めるのではなく、得られたFECビットを通常の1つのフレーム又は複数のVOPパケットの後の追加パケットに入れ、MPEG互換性を確保する。

#### 【0212】

さらに、データパケットが失われるか、又は動きマーカーを欠損している場合に、デコーダがきちんと復旧できるように、標準パケット毎に、FECパケット内にパケット識別子が格納され、このとき、FECビットを生成するために、何ビットを、及び/又はどのビットをどこで使用するかを示す対応する値と共に格納される。

#### 【0213】

参照又は従来のMPEG-4デコーダとの互換性を維持するために、この追加FECパケットは、さらにユーザー定義データを識別するのに使用されるユーザーデータ識別子コード「user\_data\_start\_code」を含んでおり、したがって、FECパケットを処理する機能を持たない従来のデコーダによって無視される。さらに、誤りがない場合、FECパケットは、FECビットを処理する機能を持つデコーダによって使用されない。しかし、誤りが発生した場合、FEC復号化により、重大な誤り状態にあっても、復号化の対象になるデータを復旧できる。

#### 【0214】

そこで、上記のように、誤り耐性及び拡張した圧縮を実現することによって、本発明の実

10

20

30

40

50

施の形態には、低ビットレートでノイズの多い環境であってもビデオ情報を送信できるという利点がある。例えば、本発明の実施の形態を利用すると、セルラーネットワークなどで正常にビデオ送信を実行することができる。

【0215】

本発明は、好ましいいくつかの実施の形態に関して説明されたが、当業者にとって明白な他の実施の形態も本発明の範囲内にある。

【図面の簡単な説明】

【0216】

【図1A】ビデオ配信システムを実装するためのネットワーク接続されたシステム例を示す図である。

10

【図1B】本発明の一実施の形態に係るエンコーダのアーキテクチャ例を示す図である。

【図1C】本発明の一実施形態に係るエンコーダのアーキテクチャ例を示す図である。

【図2A】本発明の一実施の形態と共に使用することができるリフレッシュマップ及びスキャン順序の例を示す図である。

【図2B】本発明の一実施の形態と共に使用することができるリフレッシュマップ及びスキャン順序の例を示す図である。

【図3】シーン変化を見つけるために使用されるビデオシーケンスの分析例を示す図である。

【図4A】本発明の一実施の形態に係る適応型フレームスキップの例を示す図である。

【図4B】本発明の一実施の形態に係る適応型フレームスキップの例を示す図である。

20

【図5】二次動き補正の使用例を示す図である。

【図6】パケット化されたビットストリームの例を示す図である。

【図7】本発明の一実施の形態に係る連続Iフレームの使用例を示す図である。

【図8A】適応型イントラリフレッシュのプロセス例を示す図である。

【図8B】適応型イントラリフレッシュのプロセス例を示す図である。

【図8C】適応型イントラリフレッシュのプロセス例を示す図である。

【図8D】適応型イントラリフレッシュのプロセス例を示す図である。

【図8E】適応型イントラリフレッシュのプロセス例を示す図である。

【図8F】適応型イントラリフレッシュのプロセス例を示す図である。

【図8G】適応型イントラリフレッシュのプロセス例を示す図である。

30

【図8H】適応型イントラリフレッシュのプロセス例を示す図である。

【図9】本発明の一実施の形態に係るレート制御プロセス例を示す図である。

【図10】シーンレベル再帰的ビット割り当てプロセス例を示す図である。

【図11】前方誤り訂正のオーバーヘッドと平均BER訂正能力を対比したグラフの一例を示す図である。

【図 1 A】

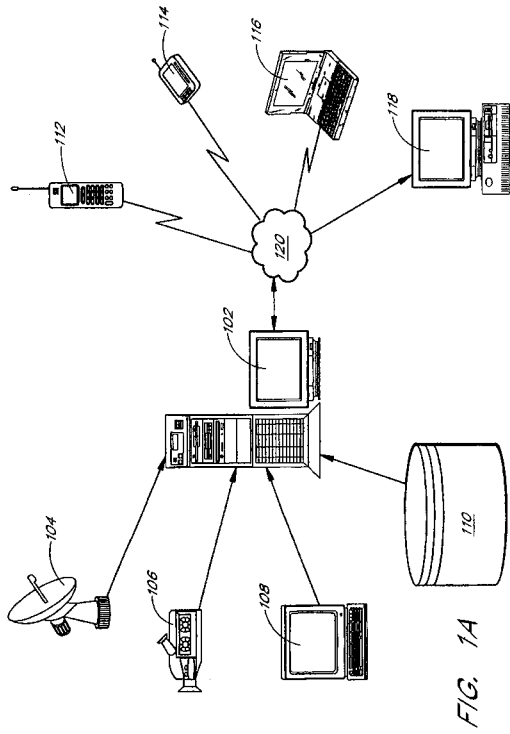


FIG. 1A

【図 1 B】

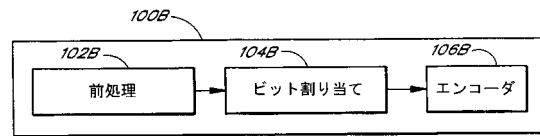


FIG. 1B

【図 1 C】

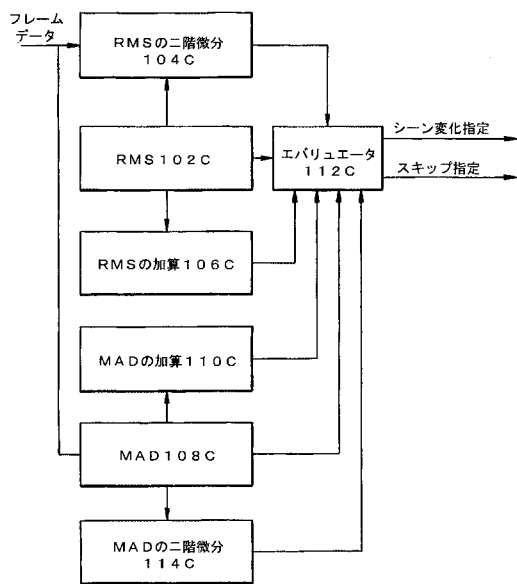


FIG. 1C

【図 2 A】

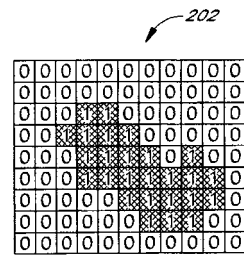


FIG. 2A  
(従来技術)

【図 2 B】

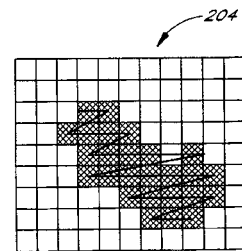
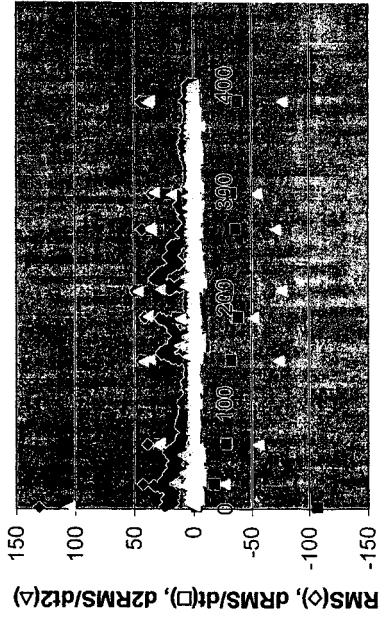


FIG. 2B  
(従来技術)

【 図 3 】



フレーム数  
FIG. 3

【 図 4 A 】

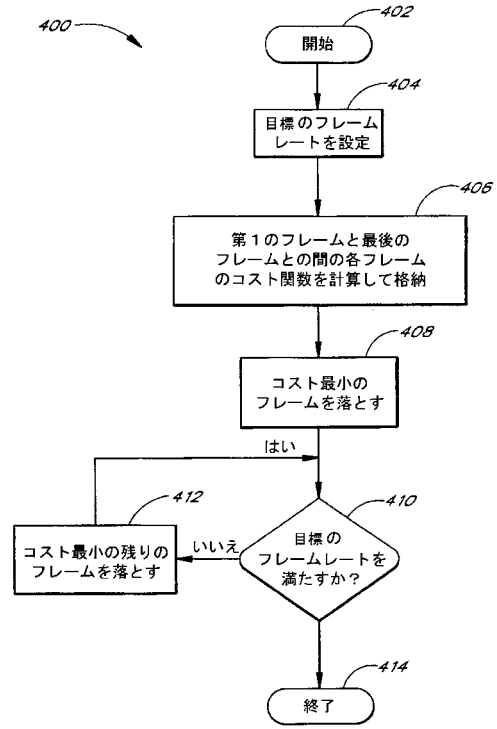


FIG. 4A

【 図 4 B 】

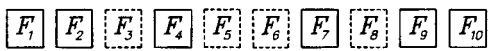


FIG. 4B

【 図 5 】

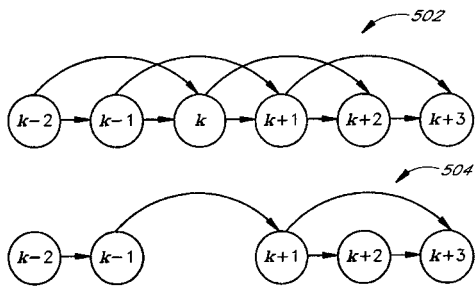


FIG. 5

【 図 6 】

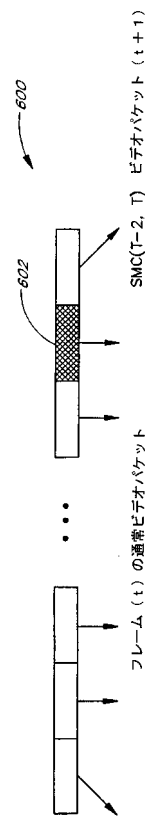


FIG. 6

【図7】

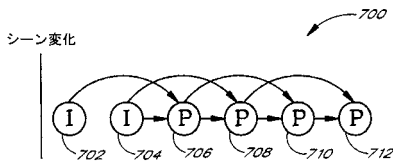


FIG. 7

【図8A】

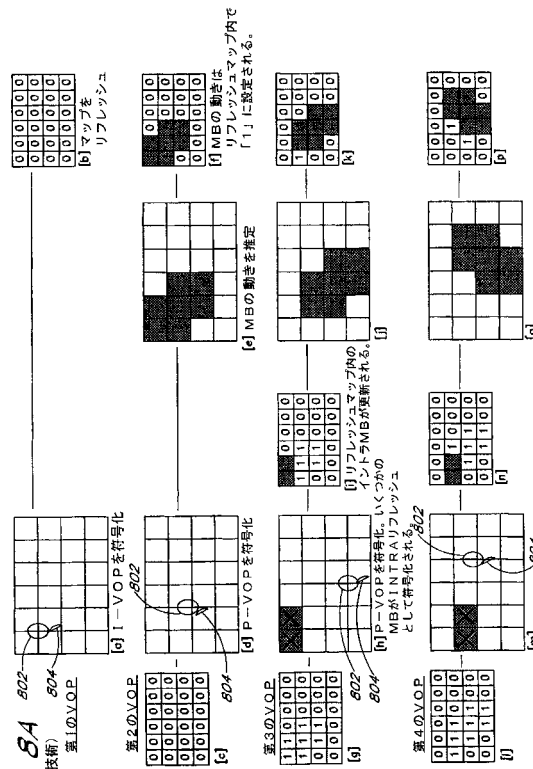


FIG. 8A  
(従来技術)

【図8B】

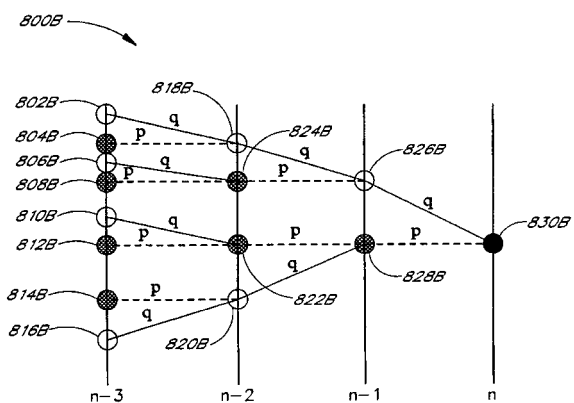


FIG. 8B

【図8C】

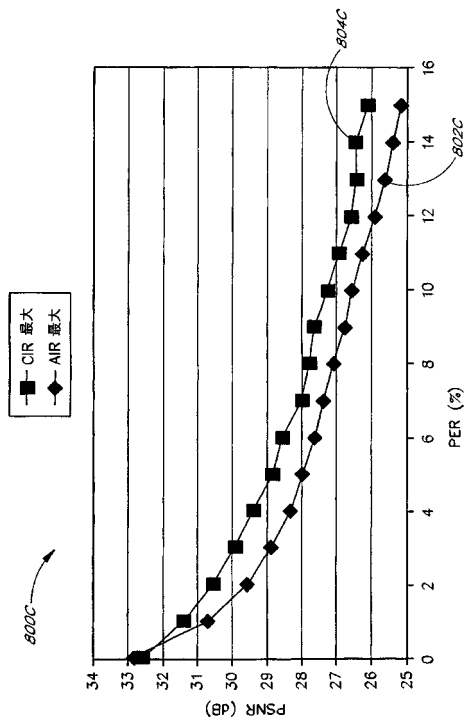


FIG. 8C

【図 8 D】

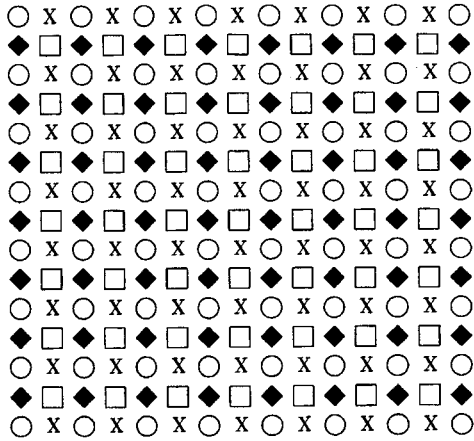


FIG. 8D

【図 8 E】

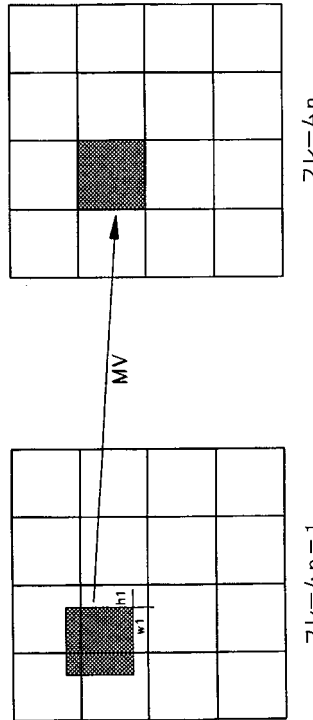


FIG. 8E

【図 8 F】

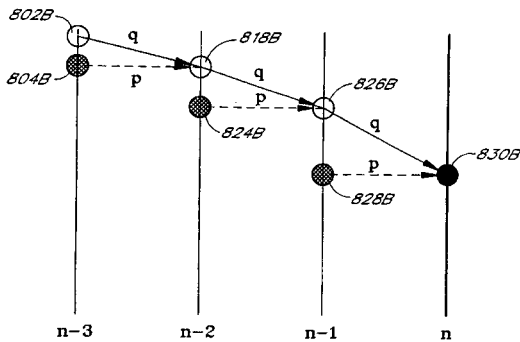


FIG. 8F

【図 8 G】

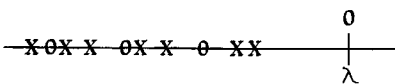


FIG. 8G

【図 8 H】

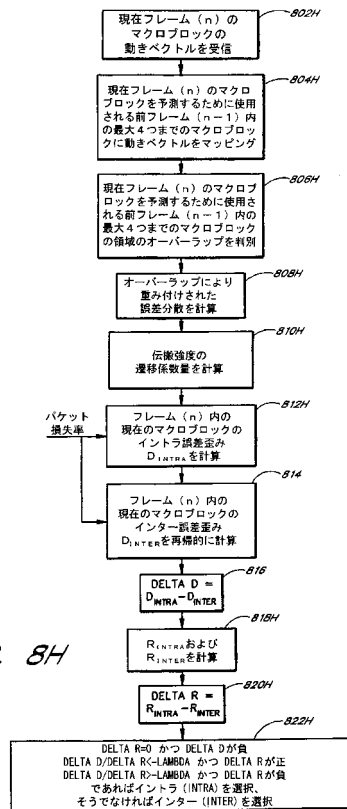


FIG. 8H

【 図 9 】

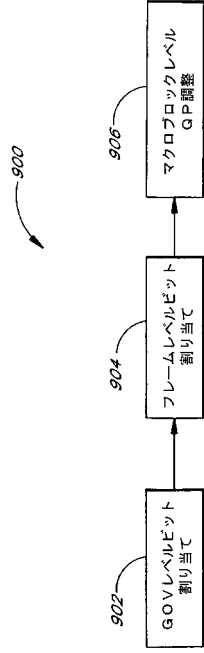


FIG. 9

【 図 10 】

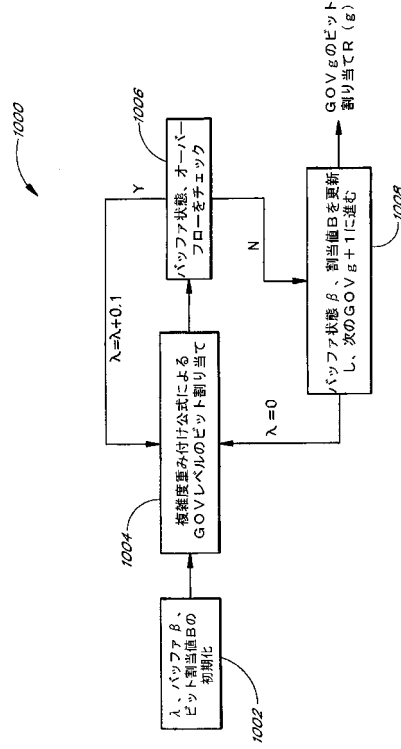
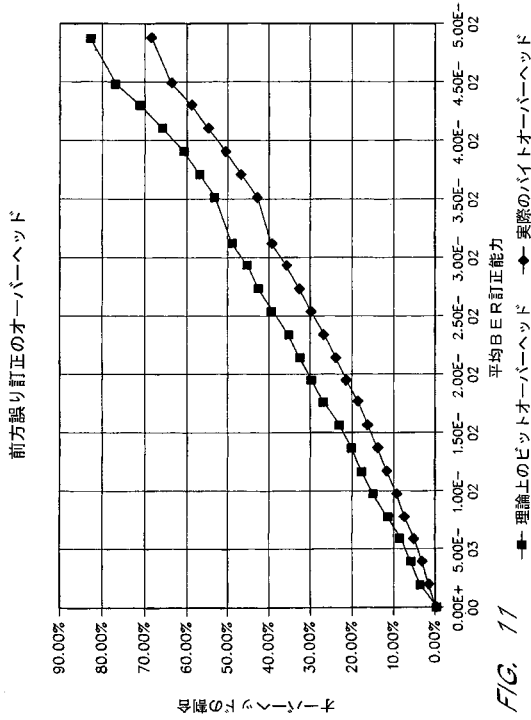


FIG. 10

【 図 11 】



【国際公開パンフレット】

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization  
International Bureau



(43) International Publication Date  
12 September 2002 (12.09.2002)

PCT

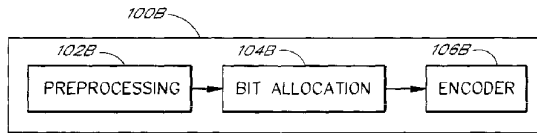
(10) International Publication Number  
WO 02/071639 A1

- (51) International Patent Classification: H04B 1/66
- (21) International Application Number: PCT/US02/06726
- (22) International Filing Date: 5 March 2002 (05.03.2002)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
  - 60/273,443 5 March 2001 (05.03.2001) US
  - 60/275,859 14 March 2001 (14.03.2001) US
  - 60/286,280 25 April 2001 (25.04.2001) US
- (71) Applicant (for all designated States except US): INTER-VIDEO, INC. [US/US]; 47350 Fremont Boulevard, Fremont, CA 94538 (US).
- (81) Designated States (national): AU, AG, AI, AM, AT (utility model), AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ (utility model), DE (utility model), DK (utility model), DM, DZ, EC, EE (utility model), ES, FI (utility model), GB, GD, GL, GR, GM, HR, HU, ID, IL, IN, IS, JP, KH, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MY, NO, NZ, OM, PL, PT, RO, RU, SD, SE, SG, SI, SK (utility model), SL, TI, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZM, ZW.
- (84) Designated States (regional): ARIPO patent (GH, GM, KI, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).



- (72) Inventors: and
  - (75) Inventors/Applicants (for US only): KATSAVOUNDIS, Ioannis [GR/US]; 222 S. El Molino Avenue, Apartment #7, Pasadena, CA 91101 (US), KUO, Chung-Chieh [US/US]; 5634 Halifax Road, Arcadia, CA 91007 (US).
  - (74) Agent: ALTMAN, Daniel, E.; Knobbe, Martens, Olson and Bear, LLP, 620 Newport Center Drive, 16th Floor, Newport Beach, CA 92660 (US).
  - Declaration under Rule 4.17:
    - of inventorship (Rule 4.17(iv)) for US only
  - Published:
    - with international search report
- For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.*

(54) Title: SYSTEMS AND METHODS FOR ERROR RESILIENT ENCODING



WO 02/071639 A1

(57) Abstract: The present invention is related to error resilient video encoding. Embodiments of the present invention provide systems (Figure 1B, 100B) and methods for reducing error propagation in a video data stream, for refreshing macroblocks, for reducing frame rates in a video data stream, for generating error correction information for a media stream, for performing bit rate allocation for a video data stream, and for detecting scene changes in a video data stream.



**SYSTEMS AND METHODS FOR ERROR RESILIENT ENCODING**Copyright Rights

A portion of the disclosure of this patent document contains material that is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by any one of the patent document or the patent disclosure, as it appears in the Patent and Trademark Office patent file or records, but otherwise reserves all copyright rights whatsoever.

Background of the InventionField of the Invention

[0001] The present invention is related to video and image coding and in particular to systems and methods for coding video image information in a compressed and error resilient manner.

Description of the Related Art

[0002] MPEG is an ISO/IEC standard developed by MPEG (Moving Picture Experts Group). There are several versions of the MPEG standard, such as MPEG-1, MPEG-2, MPEG-4, and MPEG-7, and they are intended to standardize certain aspects of image and audio compression. As with other forms of video compression, such as H.261, H.262, H.263, H.263+, H.263+++, H.26L, MPEG compression attempts to eliminate redundant or irrelevant data. For example, an MPEG encoder uses information from selected frames to reduce the overall video data that needs to be transmitted for certain other frames.

[0003] Typically, a video frame can be encoded in one of three ways, as an intraframe, as a predicted frame, and as a bi-directional frame. In addition, a video frame can also be skipped in order to reduce the resulting file size or bit-rate. An intraframe typically contains the complete image data for that frame and so does not rely on image data from other frames. Intraframe encoding provides the least compression. A predicted frame generally contains just enough information to allow a decoder to display the frame based on

WO 02/071639

PCT/US02/06726

a recent preceding intraframe or predicted frame. This means that the predicted frame contains the data that relates to how the image has changed from the previous frame and residual error correction data. A bi-directional frame is generated from information from the surrounding intraframe(s) and/or predicted frames, including residual error correction data. Using data from the surrounding frames, the decoder uses interpolation to calculate the position and color of each pixel.

**[0004]** The MPEG-4 standard was developed for use with both low and high bit rate applications. For example, MPEG-4 has been enhanced for use in interactive video games, videoconferencing, videophones, interactive storage media, multimedia mailing, wireless multimedia and broadcasting applications. MPEG-4 provides for object scalability, improved error robustness and enhanced compression.

**[0005]** The ever-increasing demand for multimedia communications via the wired/wireless Internet faces the challenge of packet loss as well as bandwidth fluctuation. The dependency between image frames makes the compressed video stream vulnerable even to a small number of lost packets. MPEG-4 has therefore been particularly enhanced for use in low bit rate (< 64 kbs), error prone applications, such as mobile, wireless applications, and error-prone ATM (asynchronous transfer mode) network applications. Mobile operation tends to be more susceptible to transmission errors as there is often less data redundancy, in order to reduce bit rates, and greater sources of "noise." For example, wireless channels can be corrupted by environmental noise, and in the case of mobile applications, by burst noise resulting from multipath fading and shadowing caused by buildings and other structures. With respect to ATM network applications, cells can be lost due to network congestion and buffer overflow.

**[0006]** MPEG-4 has enhanced error resiliency as compared to previous versions of MPEG so that video data can be more successfully transmitted over such error prone networks. For example, one error resiliency technique provided for by the MPEG-4 standard is the use of resync markers in the video bit-stream. In particular, MPEG-4 has adopted fixed interval synchronization and specifies that video object plane (VOP) start codes and resynchronization markers (i.e., the start of a video packet) appear only at legal fixed interval locations in the bitstream. This helps to avoid the problems associated with start codes emulations. Through the use of resync markers included by an encoder in the video data, synchronization lost after an error can be regained by a decoder.

WO 02/071639

PCT/US02/06726

[0007] Another error resiliency technique provided for by the MPEG-4 standard is the use of a reversible variable-length code. This code can be decoded even when read backwards, enabling a decoder to use uncorrupted information from a newly found resync marker back to the point in the data where the error occurred. Still another error resiliency technique adopted by MPEG-4 is data partitioning, used to separate motion information from texture information using a second resynchronization marker inserted between motion and texture information. Thus, if there is an error and the texture information is undecodable or lost, the decoder can utilize the motion information to conceal the error by using the motion information to compensate the previous decoded frame or VOP.

[0008] However, despite the use of the enhancements described above, many MPEG-4 encoders and decoders fail to provide sufficient error resiliency as is often desired in error-prone applications, such as in cellular phone applications. Thus, the transmission of MPEG-4 compliant video streams over cellular networks often results in unrecoverable corrupted data and the significant degradation in the quality of the video data seen by a recipient. Such video degradation can make video communication over error-prone networks undesirable for a user's perspective, and disadvantageously impedes the adoption and use of video transmissions over error-prone networks.

[0009] Further, to meet target bit rates, conventional encoders drop frames to reduce the frame rate according to a simple skipping algorithm. For example, a conventional encoder will drop every 4 of 5 frames in a video clip to convert the video clip from a 30 frames per second rate to a 6 frames per second rate. However, this simple form of skipping often has a significant adverse impact on the visual quality when decoded.

#### Summary of the Invention

[0010] The present invention is related to video encoding and in particular to systems and methods for encoding video information for transmission in a compressed manner and/or an error resilient manner. Embodiments of the present invention advantageously enable the transmission of video information even in low-bit rate, high noise environments. For example, embodiments of the present invention enable video transmission to be successfully performed over cellular networks and the like.

[0011] In one embodiment of the present invention, enhanced error resilience is provided by automatically inserting two consecutive intracoded I-frames upon a scene

WO 02/071639

PCT/US02/06726

change. By including two intracoded frames upon a scene change, error propagation is reduced on the decoder-side if one of the two sequential intracoded frames is lost or undecodable.

[0012] By way of example, in one embodiment a method of enhancing error resiliency in an encoded video transmission is provided, the method comprising: receiving an indication that a first frame corresponds to a scene change; intracoding the first frame at least partly in response to receiving the indication that the first frame corresponds to a scene change; automatically intracoding the next consecutive frame; and transmitting the intracoded first frame and the intracoded next consecutive frame.

[0013] In another embodiment, a video coding apparatus is provided, comprising: a first instruction stored in a processor readable memory, the first instruction configured to receive an indication that a first frame is to be intracoded; a second instruction stored in processor readable memory, the second instruction configured to intracode the first frame at least partly in response to the indication; and a third instruction stored in processor readable memory, the third instruction configured to automatically intracode the frame immediately after the first frame as a result of the indication that the first frame is to be intracoded.

[0014] In still another embodiment, a method of encoding video frames is provided, the method comprising: receiving a first instruction to intracode a first frame; intracoding the first frame in response to the first instruction; and intracoding a second frame as a result of the first instruction to intracode the first frame.

[0015] In yet another embodiment, an encoding apparatus is provided, comprising: a means for receiving a first indication to intracode a first frame; a means for intracoding the first frame in response to the first instruction; and a means for intracoding a second frame as a result of the first instruction to intracode the first frame.

[0016] An enhanced Adaptive Intra Refresh (AIR) process provides efficient error resiliency by selectively and adaptively encoding macroblocks in a predicted frame. In one embodiment, an adaptive motion area process is performed to determine which macroblocks are to be intracoded. Adaptive motion change detection can efficiently reduce the large propagation error, including errors occurring in the motion area. An intercode distortion value and an intracode distortion value are calculated, as are an intercode bit quantity and an intracode quantity. Based on a comparison of the calculated intercode

WO 02/071639

PCT/US02/06726

distortion value and the intracode distortion value, and on a comparison of the intercode bit quantity and the intracode bit quantity for each macroblock, a decision is made as to which predicted frame macroblocks are to be intracoded.

[0017] One embodiment of the present invention is a method of performing adaptive intra refresh, the method comprising: receiving a packet loss probability value; receiving a motion vector for a first macroblock in a first frame; mapping the motion vector onto portions of a plurality of other macroblocks in a previous frame; calculating at least a first transition factor value based at least in part on the mapping; calculating an estimated inter distortion value for the first macroblock based on at least the packet loss probability value and the first transition factor value; calculating an estimated intra distortion value for the first macroblock based on at least the packet loss probability value; receiving a first bit quantity value corresponding to a bit quantity used to intracode the first macroblock; receiving a second bit quantity value corresponding to a bit quantity used to intercode the first macroblock; and based at least in part on the estimated inter distortion value, the estimated intra distortion value, the first bit quantity value, and the second bit quantity value, transmitting the first macroblock as one of an intra macroblock and an inter macroblock.

[0018] Another embodiment of the present invention is a method of selectively intracoding macroblocks, the method comprising: receiving a packet loss probability value; receiving a motion vector for a first macroblock in a first frame; based at least in part on the motion vector, determining which portions of macroblocks in a previous frame would be used in predicting the first macroblock; calculating at least a first propagation strength value based at least in part on determining which portions of macroblocks in the previous frame would be used in predicting the first macroblock; calculating an estimated inter distortion value for the first macroblock based on at least the packet loss probability value and the first propagation strength value; calculating an estimated intra distortion value for the first macroblock based on at least the packet loss probability value; calculating a quantization distortion value for the first macroblock; receiving a first bit quantity value corresponding to a bit quantity used to intracode the first macroblock; receiving a second bit quantity value corresponding to a bit quantity used to intercode the first macroblock; and based at least in part on the estimated inter distortion value, the estimated intra distortion value, the quantization distortion value, the first bit quantity value, and the second bit

WO 02/071639

PCT/US02/06726

quantity value, providing for transmission the first macroblock as one of an intra macroblock and an inter macroblock.

[0019] Still another embodiment of the present invention is a method of selectively intracoding macroblocks in a plurality of macroblocks in a first frame, the method comprising: receiving a packet loss probability value; receiving a corresponding motion vector for each macroblock in the plurality of macroblocks; based at least in part on the corresponding motion vector, determining which portions of macroblocks in a previous frame would be used in predicting said each macroblock; calculating for each of said macroblocks at least a first corresponding propagation strength value based at least in part on determining which portions of macroblocks in the previous frame would be used in predicting said each macroblock; calculating for each of said macroblocks an estimated inter distortion value based upon at least the packet loss probability value and the corresponding at least first propagation strength value; calculating for each of said macroblocks an estimated intra distortion value based upon at least the packet loss probability value; calculating for each of said macroblocks a quantization distortion value; and based at least in part on the estimated inter distortion, the estimated intra distortion, and the quantization distortion values for each macroblock in the plurality of macroblocks, designating a subset of the plurality of macroblocks to be intracoded.

[0020] One embodiment of the present invention is a circuit configured to selectively intracode macroblocks, the circuit comprising: a first instruction configured to receive a packet loss probability value; a second instruction configured to receive a motion vector for a first macroblock in a first frame; a third instruction configured to determine, based at least in part on the motion vector, which portions of macroblocks in a previous frame would be used in predicting the first macroblock; a fourth instruction configured to calculate at least a first propagation strength value based at least in part on the determination of which portions of macroblocks in the previous frame would be used in predicting the first macroblock; a fifth instruction configured to calculate an estimated inter distortion value for the first macroblock based on at least the packet loss probability value and the first propagation strength value; a sixth instruction configured to calculate an estimated intra distortion value for the first macroblock based on at least the packet loss probability value; and a seventh instruction configured to selectively provide for transmission the first macroblock as one of an intra macroblock and an inter macroblock

WO 02/071639

PCT/US02/06726

based at least in part on the estimated inter distortion value and the estimated intra distortion value.

[0021] Embodiments of the present invention provide for adaptive skipping of frames in an encoding process based on a defined cost function. The visual quality, particularly in scene change areas, is improved while scenes are efficiently encoded. One embodiment of the skipping process is iterative in that a frame is selectively dropped from the sequence of frames by computing a mean of absolute differences between the frames adjacent to the frame of interest, and by weighting the computation with a temporal parameter, wherein the frame having the least impact on the scene is dropped. This procedure is repeatedly iterated until a target frame rate, which is related to the desired bit rate and frame size, is achieved. Other measurement techniques, such as a sum of the mean absolute differences (SMAD) or sums of root mean squared (SRMS) values, can be used in place of or in addition to the mean of absolute differences technique.

[0022] One embodiment of the present invention is a method of selecting which video sequence frames to skip during an encoding process, the method comprising: receiving a sequence of frames; for each frame in at least a portion of the sequence of frames, determining a corresponding mean of absolute differences value weighted by a temporal parameter between frames bracketing said each frame; and at least partly based on the corresponding mean of absolute differences value weighted by a temporal parameter calculated for each frame, skipping a first frame in the sequence of frames.

[0023] Another embodiment of the present invention is a method of determining which frames to skip in a video sequence, the method comprising: receiving a first frame rate goal; selecting a first potential frame to be skipped, the first potential frame between a first frame and a last frame in a video sequence scene; calculating a first cost value for the first potential frame to be skipped, the first cost value related to the visual impact that would result from skipping the first potential frame; storing the first cost value in computer readable memory; selecting a second potential frame to be skipped, the second potential frame between the first frame and the last frame; calculating a second cost value for the second potential frame to be skipped; storing the second cost value in computer readable memory; selecting a third potential frame to be skipped, the third potential frame between the first frame and the last frame; calculating a third cost value for the third potential frame to be skipped; storing the third cost value in computer readable memory; determining which

WO 02/071639

PCT/US02/06726

one of the first, second, and third cost values has the least cost; skipping the one of the one of the first, second, and third potential frames corresponding to the least cost value; determining if the first frame rate goal is met; and at least partly in response to determining the first frame rate goal is not met, recalculating cost values associated with frames remaining from the first, second and third potential frames, and skipping the one of the remaining frames having the least recalculated cost value.

[0024] Still another embodiment of the present invention is a method of selecting image frames to be skipped during an encoding process, the method comprising: receiving a sequence of frames including a first frame; calculating a first cost function value for the first frame, wherein the cost function includes a first temporal parameter related to a time difference and luminosity differences between frames bounding the first frame; calculating a second cost function value for a second frame, wherein the cost function includes a second temporal parameter related to a time difference between frames and luminosity differences bounding the second frame; and selecting for skipping one of the first frame and the second frame based at least in part on the first cost function value and the second cost function value.

[0025] Yet another embodiment of the present invention is a processor circuit configured to select video frames to skip in a sequence of frames, the processor circuit comprising: a first instruction configured to calculate for each frame in at least a portion of the sequence of frames a corresponding mean of absolute differences value weighted by a temporal parameter between frames bracketing said each frame in the portion of the sequence of frames; and a second instruction configured to designate a first frame to be skipped based at least in part on the corresponding mean of absolute differences value weighted by the temporal parameter.

[0026] One embodiment of the present invention is a frame rate control apparatus, comprising: a first instruction configured to receive a frame rate target; a second instruction configured to calculate a first cost function value for a first frame in a sequence of frames, wherein the first cost function value is based at least in part on a first temporal parameter related to a time difference and luminosity differences between frames in the sequence of frames bounding the first frame; a third instruction configured to calculate a second cost function value for a second frame in the sequence of frames, wherein the second cost function based at least in part on a second temporal parameter related to a time



WO 02/071639

PCT/US02/06726

difference and luminosity differences between frames in the sequence of frames bounding the second frame; a fourth instruction configured to select for skipping one of the first frame and the second frame based at least in part on the first cost function value and the second cost function value; and a fifth instruction configured to determine if the target frame rate is met with the selected one of the first frame and the second frame skipped.

[0027] In one embodiment, error resiliency is enhanced using forward error correction (FEC) information. FEC coding is efficiently and selectively applied in real-time to important data, such as motion vectors, DC coefficients and header information, rather than generating FEC bits for unimportant or less important data. This selected important data may be located in a packet between a packet resync field and a motion marker. In particular, for a given frame or VOP, the selected packet bits targeted for FEC coding are concatenated together and the FEC code bits are generated for the concatenated bits. Optionally, the resulting FEC bits are placed in an additional packet after the regular frame or VOP packets to ensure MPEG compatibility.

[0028] One embodiment of the present invention is a method of providing forward error correction (FEC) on a plurality of frame packets, the method comprising: concatenating selected portions of packet data corresponding to a plurality of frame packets for a first frame; generating forward error correction bits for the concatenated selected portions of packet data; and transmitting the forward error correction bits in a separate packet identified with a user data identifier code or the like, including other unique identifier codes to be assigned in the future by MPEG-standards committee and the like.

[0029] Another embodiment of the present invention is an error correction generation circuit, comprising: a first instruction stored in processor readable memory configured to generate forward error correction data for selected portions of packet data that are to be transmitted in a corresponding plurality of frame packets; a second instruction stored in processor readable memory configured to store the forward error correction data in a first packet separate from the plurality of frame packets; and a third instruction stored in processor readable memory configured to identify the first packet with a first data identifier code.

[0030] Still another embodiment of the present invention is an encoder circuit, comprising: a means for generating forward error correction data for selected portions of packet data from a plurality of frame packets; a means for storing the forward error

WO 02/071639

PCT/US02/06726

correction data in a first packet separate from the plurality of frame packets; and a means for identifying the first packet with a first data identifier code.

[0031] Further, embodiments of the present invention provide for using a Header Extension Code (HEC) in a sequence of video packets or in every video packet, and not just on the first video packet following the VOP header as with conventional encoders. This better ensures that even if a packet is lost or corrupted, subsequent packets can still be decoded and used. Further, even many conventional decoders will be able to handle the inclusion of the enhanced use of HECs.

[0032] Optionally, a Video-Object-Layer (VOL) header has a flag set indicating that a fixed Video Object Plane (VOP) increment is to be used, followed by the fixed time increment value. This will facilitate the decoder's detection of missing frames, that is, frames either skipped by the encoder in order to achieve higher compression or lost during transmission.

[0033] Embodiments of the present invention provide bit allocation on a scene level, on a frame level, and/or on a macroblock level. A bit allocation module determines an appropriate distribution of a fixed bit budget. In one embodiment, a weight is assigned to a scene based on the number of intracoded frames and the number of intercoded frames. Intracoded frames are weighted more heavily than intercoded frames to account for the greater number of bits needed to encode an intraframe. The bit allocation module distributes the fixed bit budget within a scene by comparing the current bit usage and the target bit usage and based on the comparison, adjusts a quantization parameter or step size for the current frame.

[0034] To further increase coding efficiency, an encoding quantization step or quantization parameter (QP) is dynamically adjusted. This provides for fast convergence to a desired bit budget, even for short scenes or Groups of Video Object Planes (GOVs). In addition, after encoding each scene or GOV, the QP for I-frames is dynamically adjusted based on the actual bit usage of the previous scene or GOV. This QP adaptation of I-frame can be very useful for achieving or coming close to achieving the bit budget when there are high motion segments during a scene sequence.

[0035] One embodiment of the present invention is a bit allocation method for use with video sequences, the method comprising: receiving a clip bit budget for at least a first clip; determining a scene quantity for the first clip; determining a quantity of predicted

WO 02/071639

PCT/US02/06726

frames in a first scene in the first clip; and calculating a bit budget for the first scene based at least in part on the scene quantity for the first clip, the first clip bit budget, a quantity of intracoded frames in the first scene, and the quantity of predicted frames in the first scene.

[0036] Another embodiment of the present invention is a bit allocation method for use with video clip scenes, the method comprising : calculating a bit budget for a first scene; calculating bit budgets for corresponding frames, including at least a first frame, within the first scene; and calculating bits budgets corresponding to macroblocks within the first frame.

[0037] Still another embodiment of the present invention is an apparatus for allocating bits, comprising: a first instruction configured to assign a first bit budget for a first scene based at least in part on how many intracoded frames are in the first scene, a complexity determination for the first scene, and how many intercoded frames are in the first scene; and a second instruction configured to allocate bit budgets to frames within the first scene based at least in part on a current bit usage and a target bit usage.

[0038] Yet another embodiment of the present invention is a bit budgeting apparatus, comprising : a first instruction configured to receive a bit budget for at least a first video sequence; a second instruction configured to determine a scene quantity for the first video sequence; a third instruction configured to determine a quantity of predicted frames in a first scene in the first video sequence; and a fourth instruction configured to determine a bit budget for the first scene based at least in part on the scene quantity for the first clip, the first clip bit budget, a quantity of intracoded frames in the first scene, and the quantity of predicted frames in the first scene.

[0039] Embodiments of the present invention provide accurate systems and methods for locating scene changes. Because scene change frames are intracoded, which generally utilizes more bits than intercoding while providing greater error resiliency, the accurate determination of scene change frames is important in ensuring that error resiliency is efficiently performed. In one embodiment, a first root mean squared (RMS) value is calculated for a first frame relative to a second frame. A second temporal derivative RMS value is calculated for the first frame relative to the second frame, and the second frame relative to a third frame. Based at least in part on the second temporal derivative RMS value, the second frame is designated as a scene change frame. Frames are adaptively grouped based on the scene change detection. As discussed below, other criteria, based on

WO 02/071639

PCT/US02/06726

the mean of absolute differences (MAD), and/or locating RMS local maxima, and/or locating MAD local maxima, can be used to detect a scene change.

[0040] One embodiment of the present invention is a method of detecting a scene change in a digital video sequence having a plurality of frames, the method comprising: calculating a first root mean squared (RMS) value for a first frame relative to a second frame and the second frame relative to a third frame; calculating a first mean absolute difference (MAD) value for the first frame relative to the second frame; determining if the first RMS value meets a first criterion; determining if the first MAD value meets a second criterion; and designating the second frame as a scene change frame at least partly in response to determining that both the first RMS value meets the first criterion and the first MAD value meets the second criterion.

[0041] Another embodiment of the present invention is a method of detecting a scene change in a digital video sequence, the method comprising: calculating a second temporal derivative RMS value for a first frame relative to a second frame and the second frame relative to a third frame; and based at least in part on the second derivative value, determining that the second frame is a scene change frame.

[0042] Still embodiment of the present invention is an apparatus for identifying a scene change in a video sequence, the apparatus comprising: a first instruction stored in processor readable memory, the first instruction configured to calculate a first root mean squared (RMS) value for a first portion of a video sequence relative to a second portion of the video sequence; a second instruction stored in processor readable memory, the second instruction configured to calculate a second temporal derivative RMS value; and a third instruction configured to cause the second portion of the video sequence to be intracoded based at least in part on the second derivative RMS value.

[0043] One embodiment of the present invention is method of determining which portions of a video sequence are to be intracoded, the method comprising: calculating a first root mean squared (RMS) value for a first portion of the video sequence; calculating a first mean absolute difference (MAD) value for the first portion of the video sequence; determining if the first RMS value meets a first criterion; determining if the first MAD value meets a second criterion; determining if the first MAD value meets a third criterion; and causing an intracoding operation to be performed at least partly in response to at least two of the first, second and third criteria being met.

WO 02/071639

PCT/US02/06726

[0044] Another embodiment of the present invention is a scene change detection apparatus, comprising: an RMS circuit having a first frame information input and an RMS output, the RMS circuit configured to provide at the RMS output a value corresponding to root mean squared differences information between at least two frames based on frame information received on the first frame information input; a MAD circuit having a second frame information input and a MAD output, the MAD circuit configured to provide at the MAD output a value corresponding to mean absolute differences information between at least two frames based on frame information received on the second frame information input; and an evaluator circuit coupled to the RMS output and the MAD output, the evaluator circuit configured to detect a scene change frame based at least on the value corresponding to root mean squared differences information and the value corresponding to mean absolute differences information, and to provide a scene change designation.

#### Brief Description of the Drawings

[0045] Embodiments of the present invention will now be described with reference to the drawings summarized below. These drawings and the associated description are provided to illustrate example embodiments of the invention, and not to limit the scope of the invention.

[0046] Figure 1A illustrates an example networked system for implementing a video distribution system.

[0047] Figures 1B-C illustrate an example encoder architecture in accordance with an embodiment of the present invention.

[0048] Figures 2A-B illustrate an example refresh map and an example scan order that can be used with an embodiment of the present invention.

[0049] Figure 3 illustrates an example analysis of a video sequence used to locate a scene change.

[0050] Figures 4A-4B illustrate an example of adaptive frame skipping in accordance with an embodiment of the present invention.

[0051] Figure 5 illustrates an example use of second order motion compensation.

[0052] Figure 6 illustrates an example packetized bitstream.

WO 02/071639

PCT/US02/06726

[0053] Figure 7 illustrates an example use of consecutive I-frames in accordance with an embodiment of the present invention

[0054] Figures 8A-H illustrate example processes for adaptive intra refresh.

[0055] Figure 9 illustrates an example rate control process in accordance with an embodiment of the present invention.

[0056] Figure 10 illustrates an example scene level recursive bit allocation process.

[0057] Figure 11 illustrates an example graph of Forward Error Correction overhead vs. average BER correction capability.

Detailed Description of Preferred Embodiments

[0058] The present invention is related to video encoding and in particular to systems and methods for encoding video information for transmission in a compressed and/or an error resilient manner. Thus, embodiments of the present invention advantageously enable the transmission of video information even in low-bit rate, noise, error-prone environments. Embodiments of the present invention can be used with a variety of video compression standards, such as, by way of example, the MPEG-4 standard, as well as MPEG-1, MPEG-2, H.261, H.262, H.263, H.263+, H.263++, and H.26L, and video standards yet to be developed. Aspects of the MPEG-4 standard are defined in "Coding of Audio-Visual Objects: Systems," 14496-1, ISO/IEC JTC1/SC29/WG11 N2501, November 1998, and "Coding of Audio-Visual Objects: Visual," 14496-2, ISO/IEC JTC1/SC29/WG11 N2502, November 1998, and the MPEG-4 Video Verification Model is defined in "MPEG-4 Video Verification Model 17.0", ISO/IEC JTC1/SC29/WG11 N3515, Beijing, China, July 2000.

[0059] Figure 1A illustrates a networked system for implementing a video distribution system in accordance with one embodiment of the invention. An encoding computer 102 receives a video signal, which is to be encoded to a relatively compact and robust format. The encoding computer 102 can correspond to a variety of machine types, including general purpose computers that execute software and to specialized hardware. The encoding computer 102 can receive a video sequence from a wide variety of sources, such as via a satellite receiver 104, a video camera 106, and a video conferencing terminal 108. The video camera 106 can correspond to a variety of camera types, such as video camera recorders, Web cams, cameras built into wireless devices, and the like. Video

WO 02/071639

PCT/US02/06726

sequences can also be stored in a data store 110. The data store 110 can be internal to or external to the encoding computer 102. The data store 110 can include devices such as tapes, hard disks, optical disks, and the like. It will be understood by one of ordinary skill in the art that a data store, such as the data store 110 illustrated in Figure 1A, can store unencoded video, encoded video, or both. In one embodiment, the encoding computer 102 retrieves unencoded video from a data store, such as the data store 110, encodes the unencoded video, and stores the encoded video to a data store, which can be the same data store or another data store. It will be understood that a source for the video can include a source that was originally taken in a film format.

[0060] The encoding computer 102 distributes the encoded video to a receiving device, which decodes the encoded video. The receiving device can correspond to a wide variety of devices that can display video. For example, the receiving devices shown in the illustrated networked system include a cell phone 112, a personal digital assistant (PDA) 114, a laptop computer 116, and a desktop computer 118. The receiving devices can communicate with the encoding computer 102 through a communication network 120, which can correspond to a variety of communication networks including a wireless communication network. It will be understood by one of ordinary skill in the art that a receiving device, such as the cell phone 112, can also be used to transmit a video signal to the encoding computer 102.

[0061] The encoding computer 102, as well as a receiving device or decoder, can correspond to a wide variety of computers. For example, the encoding computer 102 can be a microprocessor or processor (hereinafter referred to as processor) controlled device, including, but not limited to a terminal device, such as a personal computer, a workstation, a server, a client, a mini computer, a main-frame computer, a laptop computer, a network of individual computers, a mobile computer, a palm top computer, a hand held computer, a set top box for a TV, an interactive television, an interactive kiosk, a personal digital assistant, an interactive wireless communications device, a mobile browser, a Web enabled cell phone, a personal digital assistant (PDA) or a combination thereof. By way of example, an encoder computer may also be included in the camera 106, the cell phone 112, the PDA 114, the laptop computer 116, and/or the desktop computer 118. The computer 102 may further possess input devices such as a keyboard, a mouse, a trackball, a touch

WO 02/071639

PCT/US02/06726

pad, or a touch screen and output devices such as a computer screen, printer, speaker, or other input devices now in existence or later developed.

[0062] The encoding computer 102, as well as a decoder computer, can correspond to a uniprocessor or multiprocessor machine. Additionally, the encoder and decoder computers can include an addressable storage medium or computer accessible medium, such as random access memory (RAM), an electronically erasable programmable read-only memory (EEPROM), masked read-only memory, one-time programmable memory, hard disks, floppy disks, laser disk players, digital video devices, Compact Disc ROMs, DVD-ROMs, other optical media, video tapes, audio tapes, magnetic recording tracks, electronic networks, and other techniques to transmit or store electronic content such as, by way of example, programs and data. In one embodiment, the encoding and decoding computers are equipped with a network communication device such as a network interface card, a modem, Infra-Red (IR) port, a wireless network interface, or other network connection device suitable for connecting to a network. Furthermore, the computers execute an appropriate operating system, such as Linux, Unix, Microsoft® Windows® 3.1, Microsoft® Windows® 95, Microsoft® Windows® 98, Microsoft® Windows® NT, Microsoft® Windows® 2000, Microsoft® Windows® Me, Microsoft® Windows® XP, Apple® MacOS®, IBM® OS/2®, Microsoft® Windows® CE, or Palm OS®. As is conventional, the appropriate operating system may advantageously include a communications protocol implementation, which handles all incoming and outgoing message traffic passed over the network, which can include a wireless network. In other embodiments, while the operating system may differ depending on the type of computer, the operating system may continue to provide the appropriate communications protocols necessary to establish communication links with the network.

[0063] Figure 1B illustrates an example encoding system 100B in accordance with an embodiment of the present invention. The term encoding system, as used herein, includes one or more encoders. The encoding system 100B comprises, by way of example, one or more of processors, program logic, or other substrate configurations representing data and instructions, which operate as described herein. In other embodiments, the encoding system 100B can comprise controller circuitry, integrated circuits, gate arrays, application specific circuits, processor circuitry, processors, general purpose single-chip or multi-chip microprocessors, digital signal processors, embedded microprocessors,



WO 02/071639

PCT/US02/06726

microcontrollers and the like, executing software code, including instructions and data stored in computer readable memory. By way of example and not limitation, the encoding system 100B can be housed in one or more leaded, leadless, or ball grid array semiconductor packages, on one or more circuit boards, and/or using one or more hybrid packages. All or portions of the encoding system 100B may be included in a fixed terminal, such as a desktop computer, or in a portable terminal, such as a cellular phone, portable computer, personal digital assistant, video camera, or the like. The encoding system 100B can, in an example embodiment, correspond to the encoding computer 102. By way of further example, an encoding system in accordance with the present invention can be used to conduct video conferencing, to aid in the storage and transmission of movies or other images, and the like.

**[0064]** The encoding system 100B encodes and compresses video information for transmission to a decoder. The encoding system 100B includes a preprocessing module or circuit 102B, a bit allocation module or circuit 104B, and an encoder module or circuit 106B. The preprocessing module or circuit 102B, including a video sequence analyzer, is used to detect when a scene change has taken place and to determine how a given frame, VOP or picture, is to be encoded.

**[0065]** A video object layer contains a sequence of 2D representations of arbitrary shape at different time intervals that is referred to in MPEG-4 as a video object plane (VOP). Each of the VOP regions can be non-rectangular and may correspond to particular image or video content of interest, such as physical objects within a scene. Video object planes (VOPs) are divided into macroblocks of size 16x16. A macroblock is encoded in six blocks, four for luminosity and two for chromaticity, of size 8x8. To obtain a macroblock structure from an arbitrary shaped VOP, the bounding box of the VOP is calculated and extended to multiples of the macroblock size.

**[0066]** However, for most current applications, and in particular for wireless applications using the so-called "simple profile," there is generally only 1 VOP per frame, which is a rectangular VOP. For clarity, the term frame, as used herein, can also include a VOP, such as an MPEG-4 VOP, or a picture. Similarly, the term VOP, as used herein, can also refer to a frame. In MPEG-4, the VOPs can be structured in groups of video object planes (GOV). Using MPEG-2 terminology, frames or pictures can be arranged in groups

WO 02/071639

PCT/US02/06726

of pictures (GOPs). For clarity, the term "scene" as used herein, may also refer to a GOV or a GOP and visa versa.

[0067] A frame or video object may be encoded as an intracoded frame (an "I-frame" or "I-VOP"), as a predicted frame (a "P-frame" or "P-VOP"), or as a bi-directional frame (a "B-frame" or "B-VOP"). MPEG-1 also provides for a D-frame. A D-frame is a frame that has no motion vectors, so that a zero vector is assumed, and has texture DCT data. To exploit spatial redundancy, a Discrete Cosine Transformation (DCT) is performed on the encoded frames and the resulting coefficients are quantized.

[0068] The MPEG-4 simple profile does not support B-frames or B-VOPs. However, the simple profile does support frame skipping. A video frame can be skipped in order to reduce the resulting file size or bit-rate. Because the MPEG-4 simple profile does not support B-frames or D-frames, the following discussions will not focus on such frames. Nonetheless, embodiments of the present invention can be used with B-frames and D-frames in accordance with other profiles and other standards.

[0069] The term frame can correspond to either an interlaced frame or to a non-interlaced frame, i.e., a progressive frame. In an interlaced frame, each frame is made of two separate fields, which are interlaced together to create the frame. Such interlacing is not performed in a non-interlaced or progressive frame. While illustrated in the context of non-interlaced or progressive video, one of ordinary skill in the field will appreciate that the principles and advantages described herein are applicable to both interlaced video and non-interlaced video. In addition, while embodiments of the invention are described in the context of MPEG-4, aspects of the principles and advantages described herein are also applicable to other video standards, including, by way of example, MPEG-1, MPEG-2, H.261, H.262, H.263, H.263+, H.263++, and H.26L, as well as video standards yet to be developed.

[0070] An intracoded I-frame typically only includes information from the image itself and thus an I-frame can be decoded independently of other frames. P and B frames are also referred to as intercoded frames because they are encoded based on data from other frames. The preprocessing module 102 generates a file, referred to as an input frame-type file, containing the frame-type designations corresponding to the frames. In other embodiments, the frame-type information is passed to other portions of the encoding system 100B using variables and the like. While the preprocessing module 102B is

WO 02/071639

PCT/US02/06726

illustrated in Figure 1B as being included in the encoding system 100B, the preprocessing module 102B can be physically separate from the other portions of the encoding system 100B. In such an embodiment, the preprocessing module 102B can produce a text file that includes frame-type designation that is then input by the remainder of the encoding system 100B.

[0071] Many standard MPEG-4 encoders can only handle one scene, that is, one I-frame followed by P-frames or B-frames, or they introduce regular I-frames every k frames, as is commonly done in MPEG-2 encoding. These approaches make implementation of the encoder simpler, however they place the burden on the user to determine how clips of multiple scenes are to be handled. To adequately improve the coding efficiency, the number of I-frames should be reduced or minimized. In the absence of error conditions, I-frames are preferably used in scene-changes only. Thus, it is advantageous to correctly and accurately detect scene changes.

[0072] An example scene change detection process in accordance with an embodiment of the present invention will now be described. In the example embodiment, the process operates on YUV-4:2:0 files and produces a text file as an output. In the example embodiment, YUV-4:2:0 files are header-less files with concatenated frames, where, for each frame, the (luminosity) Y-pixel values are provided first, followed by the (Chromaticity-blue) Cb-values, and then the (Chromaticity-red) Cr-values. The term "4:2:0" indicates that chromaticity values are subsampled by a factor 4 with respect to luminosity. In particular, if the size of a frame (in pixels) is  $W \times H$  ( $W$ : width,  $H$ : height), there are  $W \times H$  Y-values (1 for each frame pixel),  $(W/2) \times (H/2)$  Cb-values and  $(W/2) \times (H/2)$  Cr-values for each frame. That gives a total of  $3 \times W \times H/2$  bytes as the frame buffer size needed to store a frame of size  $W \times H$ . The subsampling for chromaticity components is achieved by subsampling along the vertical and horizontal dimensions by a factor of 2. Thus, a  $2 \times 2$  block has 4 luminosity values, and 1 chromaticity-blue and 1 chromaticity-red. In other embodiments, other formats for storing image data can be used.

[0073] The preprocessing module 102B will now be described in greater detail. The preprocessing module 102B performs frame evaluation and encoding designation. As will be described below, each frame is designated by the preprocessing module 102B as an I-frame, a P-frame, or as a skipped frame. In other embodiments, the preprocessing module 102B may also designate frames as B-frames or D-frames. B-frame encoding may be

WO 02/071639

PCT/US02/06726

performed if there is sufficient computational power, available bandwidth (B-frames take much more bandwidth than skipped frames), and if allowed by the corresponding standard. For example, the MPEG-4 simple-profile syntax, used in wireless networks, does not allow for B-frames. The example file format generated by the preprocessing module 102B includes a line per input frame, with a frame-type designation character on each line: 0, 1 or 2. A "0" indicates an I-frame, a "1" indicates a P-frame, and a "2" indicates a skipped frame. In other embodiments, designations can be provided for a bidirectional frame and a D-frame.

[0074] As previously discussed, scene change frames are generally intracoded. To locate the scene change frames, the preprocessing module's scene change analysis performs a color-weighted Root Mean Squared (RMS) calculation and a Mean Absolute Differences (MAD) calculation between the  $i^{\text{th}}$  frame  $F_i$  and the  $k^{\text{th}}$  frame  $F_k$ . The RMS can be defined as:

$$\begin{aligned} RMS(F_i, F_k) = & \frac{\alpha}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^w \sum_{y=1}^h |Y_i(x, y) - Y_k(x, y)|^2} + \\ & \frac{2\beta}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^w \sum_{y=1}^h |U_i(x, y) - U_k(x, y)|^2} + \\ & \frac{2\gamma}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^w \sum_{y=1}^h |V_i(x, y) - V_k(x, y)|^2} \end{aligned} \quad \text{Equation 1}$$

[0075] where  $F(x, y)$  denotes the  $(x, y)^{\text{th}}$  pixel in frame  $F$ , and  $w$  and  $h$  are the width and height of the frame, respectively.  $Y(x, y)$  indicates the luminance value, while  $U(x, y)$  and  $V(x, y)$  are the two chromaticity components. The coefficients  $\alpha$ ,  $\beta$  and  $\gamma$  are weighting coefficients for the luminosity, chromaticity-blue and chromaticity-red components correspondingly. To ease computations, the weighting coefficients can be fixed. For example, the weighting coefficients can be set as follows:  $\alpha = \beta = \gamma = 1$ .

[0076] The Mean Absolute Difference (MAD) measure can be defined as follows:

$$MAD(F_i, F_k) = \frac{1}{w \times h} \sum_{x=1}^w \sum_{y=1}^h |Y_i(x, y) - Y_k(x, y)| \quad \text{Equation 2}$$

[0077] In this example, the MAD does not need to include the two chromaticity components.

WO 02/071639

PCT/US02/06726

[0078] If  $MAD(F_i, F_k)$  and/or  $RMS(F_i, F_k)$  are large or greater than a selected criteria, this indicates that the content of  $F_i$  is substantially different from  $F_k$ . Thus in one embodiment, if the MAD between consecutive frames,  $MAD(F_{i-1}, F_i)$ , is larger than a pre-specified threshold,  $F_i$  is designated a scene change frame. An example threshold value for designating a scene change frame is approximately 25.

[0079] Optionally, a second temporal derivative of the RMS can be used to determine if a frame is scene change frame, as follows:

$$\frac{d^2(RMS)}{dt^2}(i) = RMS(F_{i-1}, F_i) - 2RMS(F_i, F_{i+1}) + RMS(F_{i+1}, F_{i+2})$$

Equation 3

[0080] As defined by Equation 3, the second temporal derivative of the RMS is based on the RMS value for the previous frame  $F_{i-1}$  relative to the current frame  $F_i$ , the RMS value of the current frame  $F_i$  relative to the next frame  $F_{i+1}$ , and the RMS value of the next frame  $F_{i+1}$  to the subsequent frame  $F_{i+2}$ .

[0081] The second temporal derivative of the RMS value will be negative with relatively high amplitude when  $F_i$  is a scene-change frame, as illustrated in Figure 3. Thus, if the absolute value of the second temporal derivative of the RMS value is larger than a pre-specified threshold,  $F_i$  is designated a scene change frame. As illustrated in Figure 3, there is a correlation between the RMS values, indicated by the diamonds, and the second derivative of RMS, indicated by the triangles. Thus, both the RMS values and the values of the second derivative of RMS generally provide a correct indication of a scene change. An example second derivative of RMS threshold value for determining a scene change is -6.5.

[0082] While the second derivative of the RMS is a good peak detector, it is somewhat sensitive to noise. To better increase the accuracy of the scene change determination, in one embodiment only if both the temporal activity measures of MAD and the second-order derivative of RMS indicate that the corresponding thresholds are met or passed, then a frame is designated a scene change frame. As will be described below, scene changes frames will be intracoded as I-frames or I-VOPs.

[0083] In particular, in one example a frame is designated as a scene change, and thus will be coded in INTRA mode, when its MAD is greater than 20 and the second derivative of RMS is negative and has an absolute value of greater than 4. In another

WO 02/071639

PCT/US02/06726

example, a frame is designated as a scene change, and thus will be coded in INTRA mode, when its RMS is greater than 40 and/or when the second derivative of RMS is negative and has an absolute value of greater than 8. In other embodiments other thresholds can be used. Alternatively or in addition, a second derivative of MAD can be used, as similarly described above with respect to the second derivative of RMS, as a further indication of whether a frame corresponds to a scene change or not.

[0084] An additional criterion can be used to determine when a scene change has occurred. For example, in one embodiment, a determination is made as to whether the MAD value is a local maximum, that is, has increased from a previous frame to the frame at issue, and then decreased from the frame at issue to the next frame. If so, this indicates that it is likely the frame at issue is a scene change frame and should be intracoded. In addition, a similar determination may be made for the RMS value. For example, a determination is made as to whether the RMS value is a local maximum, that is, has increased from a previous frame to the frame at issue, and then decreased from the frame at issue to the next frame. If so, this too indicates that it is likely the frame at issue is a scene change frame and should be intracoded.

[0085] Optionally, a voting process can be used, wherein if at least two of the RMS, the second derivative of the RMS, and the MAD, meet corresponding criteria, then a frame is designated as a scene change that is to be intracoded. In another embodiment, if the RMS and second derivative of the RMS meet the corresponding criteria, and if the MAD is a local maximum, then the frame is designated as a scene change frame. In still another embodiment, if the RMS and MAD meet the corresponding criteria, and if the MAD is a local maximum, then the frame is designated as a scene change frame.

[0086] To further improve coding efficiency and to meet target bit rates, the number of frames that needs to be encoded per second is preferably reduced as much as acceptable. One technique used to reduce the number of frames encoded per second is to skip-frames in the encoding process. Two example frame-skipping techniques are fixed frame skipping and adaptive frame skipping. Conventional encoders drop frames to reduce the frame rate according to a simple skipping algorithm. For example, a conventional encoder will drop every 4 of 5 frames in a video clip to convert the video clip from a 30 frames per second rate to a 6 frames per second rate.

WO 02/071639

PCT/US02/06726

[0087] As will be discussed below, fixed frame skipping tends to provide better error resiliency in noisy environments, while adaptive frame skipping tends to provide better visual results in low noise environments. Optionally, the preprocessing module 102B calculates, based on the bit-rate/frame-rate formula defined in Equation 7 below, the target encoding frame-rate and then switches between adaptive and fixed skipping in order to meet a target encoding frame rate.

[0088] In fixed frame skipping, the input video frame sequence is subsampled along the time axis, by keeping 1 in every k frames, where k is the subsampling factor. For example, if:

$$k = 5$$

and the original video sequence has a frame rate = 25 frames-per-second (fps); then the subsampled sequence has a frame rate = 5 fps.

[0089] In adaptive frame skipping, as in fixed frame skipping, the input video frame sequence is subsampled along the time axis in order to achieve a desired or predetermined average frame rate. However, rather than skip-frames in a fixed manner, using adaptive frame skipping the rate of frame skipping can be irregular and can vary along the sequence length. Preferably, low activity frames are identified and skipped, and scene-change frames are kept and intracoded. Non-scene changes having some activity frame are intercoded. Because the skipped frames are intelligently selected based on changes on visual activity, the visual result when reproduced by the decoder will be better than with fixed frame skipping, assuming no or relatively few errors occur.

[0090] In one embodiment, the preprocessing module 102B codes skipped frames using a "not\_coded" bit-flag or indicator set in the video object plane (VOP) header in an MPEG bit stream. An MPEG-4 video packet starts with the VOP header or the video packet header, followed by motion\_shape\_texture(), and ends with next\_resync\_marker() or next\_start\_code(). Thus, a VOP specifies particular image sequence content and is coded into a separate video object layer by coding contour, motion and texture information.

[0091] In another embodiment, a skipped frame is skipped altogether, without inserting VOP-header information in the bitstream. A skipped frame may be recreated by a decoder using interpolation or by repeating a previous frame. The decoder may perform interpolation by pixel averaging between a preceding frame and a subsequent decoded frame, weighted by their time difference.

WO 02/071639

PCT/US02/06726

[0092] The information regarding the video sequence frame rate is normally carried in the Video-Object-Layer (VOL) header. In particular, a parameter, referred to as `vop_time_increment_resolution`, determines the number of time units for each encoding cycle. The `vop_time_increment` value in the Video-Object-Plane (VOP) header carries the time stamp for each frame.

[0093] The `vop_time_increment_resolution` may be, by way of example, a 16 bit unsigned integer value. For example, for 25 frames per second (fps), `vop_time_increment_resolution = 25`, and `vop_time_increment` cycles through the values 0...24. For 7.5 fps, `vop_time_increment_resolution=75`, and `vop_time_increment` cycles through the values 0, 10, 20, 30, 40, 50, 60, 70, 5, 15, 25, 35, 45, 55, 65. Thus, successful decoding of the first 2 frames of a sequence can yield the exact frame rate. But, given the error-prone environment of wireless communication or other error-prone environments, the successful reception and decoding of any 2 consecutive frames cannot be guaranteed. Thus, the decoder may incorrectly determine the frame rate of the entire sequence. For that reason, in one embodiment, the "fixed\_vop\_rate" flag is set by the encoder module 106B in the VOL header, which then provides (via the value of `fixed_vop_time_increment`) the default frame rate. This technique better ensures the successful decoding or determination of the frame rate upon the successful decoding of the VOL header. The `fixed_vop_time_increment` value can later be stored in a global variable of the decoder, which will use the value to determine whether certain frames need to be interpolated or not. The frames to be interpolated can either be frames skipped by the encoder, or lost during transmission. Thus, the error-resilience performance of the MPEG-4 decoder will be enhanced because it will decode the correct number of frames, thereby avoiding loss-of-sync problems with the audio stream.

[0094] As discussed above, for error-prone environments, error resiliency can be enhanced by utilizing fixed frame skipping rather than adaptive frame skipping. Fixed frame skipping enables the decoder to better determine when a frame has been dropped or skipped. Another approach to enhancing error resiliency is to use adaptive frame skipping, but provide a VOP-header with the `not_coded` flag set for a skipped frame. One drawback of this approach is that it results in a slight increase in bit rate due to the more frequent VOP-headers.



WO 02/071639

PCT/US02/06726

[0095] Figure 4A illustrates one example process 400 of adaptive frame skipping. The process is iterative in that a frame is selectively dropped from the sequence of frames by computing a mean of absolute differences between the frames adjacent to the frame of interest, and by weighting the computation with a temporal parameter, wherein the frame having the least impact on the scene is dropped. This procedure is repeatedly iterated until a target frame rate, which is related to the desired bit rate and frame size, is achieved.

[0096] Beginning at start state 402, the process 400 proceeds to state 404. The desired frame rate is set or specified. The desired frame rate may be user specified or may be dynamically determined. Proceeding to state 406, the cost function, or adverse impact, that would result from dropping a particular frame is calculated for each frame between the first and last frame in a scene. As described in greater detail below, the cost function can be based at least in part on the mean absolute differences (MAD) between frames closely or most closely bracketing or bounding the particular frame of interest, or on the sum of the mean absolute differences (SMAD). Alternatively or in addition, the cost function can be based on sums of RMS (SRMS) for frames bracketing the particular frame of interest.

[0097] Proceeding to state 408, the frame associated with the lowest cost, that is, having the least adverse impact on visual quality, is skipped or dropped. At state 410 a determination is made as to whether the remaining frames will allow the target frame rate to be met. If the target frame rate can now be met, the adaptive frame rate skipping process 400 proceeds to the end state 414. Otherwise, the process 400 proceeds to state 412, and a remaining frame having the lowest cost will be dropped. The cost of all the frames remaining between the first and last frames may be recalculated at state 412 based on the frames that are currently remaining, and the frame with the lowest cost will be dropped. The process 400 repeats states 410 and 412 until the target frame rate has been met, or the number of already consecutively skipped frames is at a specified maximum. While the process 400 is described with respect to selecting which frames to skip, the process 400 can be similarly used to decide what frames should be bi-directionally encoded.

[0098] One example embodiment of an adaptive frame skipping selection process using the weighted sum of mean absolute differences will now be described in greater detail. Given an input sequence of a frame rate  $r_{orig}$  and the desired frame rate  $r_{des}$  (where  $r_{orig} > r_{des}$ ), the video sequence analyzer skips frames in a "greedy" way, that is, until  $r_{des}$  is met. The weighted sum of MAD and the time difference between two frames

WO 02/071639

PCT/US02/06726

are used in specifying a cost function for identifying the next frame to be skipped. The frame whose cost function meets predetermined criteria or is minimum among remaining candidates for skipping is skipped.

[0099] By way of example, referring to Figure 4B, if using the greedy approach  $F_3$ ,  $F_5$ ,  $F_6$ , and  $F_8$  have already been skipped,  $F_4$  is now considered as a candidate for the next skipped frame. In order to determine if  $F_4$  is to be skipped, a cost function is calculated assuming that  $F_4$  has been skipped. Upon skipping  $F_4$ ,  $F_2$  and  $F_7$  will be the left and right frames bounding the skipped segment of  $F_3$ - $F_6$ . The cost function is then defined as:

$$[0100] \quad \text{Cost} = \text{MAD}(F_2, F_7) + \lambda \frac{29.97}{r_{\text{orig}}} TD \quad \text{Equation}$$

5

[0101] Where the term

$$\frac{29.97}{r_{\text{orig}}}$$

is used to normalize the original frame rate  $r_{\text{orig}}$  with respect to the NTSC frame rate of 29.97 frames/second and where  $TD$  denotes the time difference measure. Of course other or different normalizations may be used as well.

[0102] Thus, in this example  $TD$  is 5 ( $=7-2$ ), and  $\lambda$  is a weighting coefficient. In this example, the weighting coefficient value  $\lambda = 5.0$ , determined experimentally, provides a suitable result. Alternatively, the weighting coefficient value  $\lambda$  may be dynamically determined. At the current state, the frame whose cost function is minimum among the candidates is skipped. This process is iteratively performed until the desired frame rate is met. The time difference measure  $TD$  can be modified such that no more than  $n$  consecutive frames are skipped. For example, if a predetermined maximum number of consecutive frames that can be skipped simultaneously is set to 4,  $TD$  can be modified as follows

$$TD = \begin{cases} TD & \text{if } TD \leq 5 \\ \infty & \text{otherwise} \end{cases}$$

where  $\infty$  is infinity

WO 02/071639

PCT/US02/06726

[0103] Optionally, the complexity for the MAD calculation can be reduced, though the accuracy will be reduced, if only even (or only odd) coordinate pixels are used in the calculation. For example:

$$MAD(F_i, F_k) = \frac{1}{w/2 \times h/2} \sum_{x=1}^{w/2} \sum_{y=1}^{h/2} |Y_i(2x, 2y) - Y_k(2x, 2y)| \quad \text{Equation 6a}$$

[0104] Another example process to adaptively determine which frame to skip estimates the total induced distortion, both spatial and temporal, for each candidate frame for skipping, and then skips the frame whose absence would result in the least distortion. As described below, the process utilizes sums of MAD (SMAD) or sums of RMS (SRMS). Advantageously, all the MADs do not have to be recalculated. Instead, the already calculated appropriate MADs are summed differently, depending on which frame is being considered for skipping.

[0105] In one embodiment, the process is performed as follows. Assume that  $F_i$  is the frame currently being considered for skipping. For example, assume  $F_i$  is frame  $F_4$  in Figure 4B. Then let  $F_j$  indicate the previous non-skipped frame,  $F_2$  in this example, and let frame  $F_k$  indicate the next non-skipped frame,  $F_6$  in our example. Then the cost function can be defined as follows:

$$\text{Cost} = |SMAD_{est}(F_j, F_k) - SMAD_{est}(F_j, F_i) - SMAD_{est}(F_i, F_k)| + \lambda \frac{29.97}{r_{orig}} TD \quad \text{Equation 6b}$$

[0106] where  $SMAD_{est}(F_i, F_k)$  is the estimated spatial distortion when skipping frames  $(i+1), \dots, (k-1)$ .

[0107] As can be seen from Equation 6b, the cost function subtracts out the contribution from previously skipped frames. The distortion is calculated as follows:

$$SMAD_{est}(F_i, F_k) = \varphi(k-i-1) \cdot \frac{k-i-1}{k-i} \sum_{j=i+1}^k MAD(F_{j-1}, F_j) \quad \text{Equation 6c}$$

[0108] where  $\varphi(n)$  is a coefficient that depends on the number ( $n$ ) of consecutive skipped frames and takes into account how much, on the average or based on a statistical sampling, interpolated frames at the decoder are different from the original

WO 02/071639

PCT/US02/06726

frames. The following table provides example coefficient values determined experimentally using typical video sequences:

$n$	0	1	2	3	4	5	6	$\geq 7$
$\phi(n)$	0.0	0.746	0.886	1.010	1.198	1.276	1.348	1.500

Adaptive Frame Skipping Coefficient Table

[0109] As can be seen,  $\phi(n)$  increases as the number of skipped frames increases.

[0110] To reduce the computational complexity and resources needed, optionally only the MAD between consecutive frames ( $j-1$  and  $j$ ) is used to estimate the distortion of a complete segment of skipped frames, as in Equation 6b above. The additional distortion when skipping a certain frame is calculated and the temporal component is added.

[0111] In one embodiment, a user specifies the desired encoding frame rate. The desired frame rate can be based on the video sequence statistics, such as temporal and spatial complexity, frame size, frame rate and target bit rate or compression ratio. In another embodiment, a heuristic can be used to select the desired frame rate. The following is an example heuristic equation for calculating the frame rate, normalized with respect to the Quarter Common Intermediate Format (QCIF) frame size, containing 144 lines and 176 pixels per line:

$$\text{Encoding frame rate} = \frac{\text{target bit rate in kbps}}{8 \sqrt{\frac{176 * 144}{w * h}}} \quad \text{Equation 7}$$

[0112] where  $w$  and  $h$  are the frame dimensions. In order for the encoding frame rate to remain within reasonable bounds, the encoding frame rate should preferably be in the range of :

[1 up to, and including, the source frame rate]

[0113] In addition, in order to make adaptive frame skipping independent of the temporal complexity of a specific scene, the weighting coefficient  $\lambda$  is optionally set equal to the average RMS or MAD of the designated entire sequence.

WO 02/071639

PCT/US02/06726

[0114] Thus, to determine which frames are scene change frames, and to adaptively select which frames are to be skipped, the video sequence analyzer is provided with the video sequence, the frame width, frame height, the source frame rate, the target bit rate, and the setting of the error resilience flag in the following format:

<input file> <width> <height> <source frame rate> <target bit rate> <error resilience flag>

[0115] The error resilience flag is set by the user to switch between adaptive frame skipping, which has less error resiliency but a better visual result when there are no or few errors, and fixed frame skipping, which provides better error resiliency with a lesser visual result.

[0116] In one embodiment, as illustrated in Figure 1C, an RMS circuit 102C is used to calculate RMS values as described above, a Second Derivative of RMS circuit 104C is used to calculate the second derivative of RMS as described above, a MAD circuit 108C is used to calculate the MAD values as described above, a SUM OF MAD circuit 110C is used to calculate the SUM of MAD values as described above, and a Second Derivative of MAD circuit 114C is used to calculate the Second Derivative of MAD as described above. An Evaluator circuit 112C coupled to the outputs of the RMS circuit 102C, the Second Derivative of RMS circuit 104C, the MAD circuit 108C, and the SUM OF MAD circuit 110C, and the Second Derivative of MAD circuit 114C, is used to determine when a scene change has occurred and what frames to skip, based on one or more of the outputs, as discussed above. Of course different embodiments need not include all or any portion of the circuits illustrated in Figure 1C.

[0117] The bit allocation module or circuit 104B will now be described. The bit allocation module or circuit 104B provides for bit allocation on a scene, frame, and/or macroblock level. The bit allocation module 104B reads the file or otherwise receives the information generated by the preprocessing module 102B, including the frame-type designations, and calculates a bit budget for each scene, GOV or GOP based on the coded frames. The bit allocation module determines an appropriate distribution of a fixed bit budget.

[0118] As will be described below with respect to an example implementation, a first intracoded frame defines a beginning of a scene. A weight is assigned to the scene based on the number of intracoded frames and the number of intercoded frames, where

WO 02/071639

PCT/US02/06726

intracoded frames are weighted more heavily than intercoded frames to account for the greater number of bits needed to encode an intraframe. The bit allocation module distributes the fixed bit budget within a scene by comparing the current bit usage and target bit usage, and based on the comparison, adjusts a quantization parameter or step size for the current frame.

[0119] In particular, the bit-allocation module 104B first parses the input frame-type file from the preprocessing module 102B. The number or quantity of GOVs is then calculated. Based on the calculated bit budget, the encoder module 106B then encodes each GOV using the novel rate control process in accordance with an encoder parameter file, discussed below.

[0120] For a given GOV or scene, the number of coded frames and the first and end frame are known. Using the following definitions, the bit budget for a GOV or scene is calculated using Equation 8 below:

[0121]  $N_{ci}$  = number of coded P-VOPs (predicted, intercoded VOPs) in scene; or GOVi.

$B_i$  = bit budget for scene i.

$B$  = bit budget for a clip including one or more scenes

$N_c$  = number of coded frames for the clip

$N_s$  = number of scenes in clip. Usually, in the absence of the inclusion of consecutive I-frames for error resilience purposes,  $N_s$  = the number of I-VOPs (intracoded VOPs)

$T_c$  = equivalent total number of VOPs in clip

[0122] An example method of determining the bit-budget for a scene or GOV is as follows.  $B_i$  bits are allocated for each GOV (i). In this example allocation, an assumption is made that the bit usage for one I-frame or I-VOP is close to or equal to the bit-usage of ten P-frames or P-VOPs (Ratio\_Of\_I\_to\_P = 10). However, the method is not limited to using the 1:10 ratio.  $B_i$  and  $T_c$  are then determined as follows:

$$B_i = B * (N_{ci} + \text{Ratio\_Of\_I\_to\_P}) / T_c \quad \text{Equation 8}$$

and

$$T_c = N_c + (\text{Ratio\_Of\_I\_to\_P} - 1) * N_s \quad \text{Equation 9}$$

WO 02/071639

PCT/US02/06726

[0123] As defined by Equation 8, the bit allocation for a given scene is based on the total number of frames in the scene, wherein an intracoded frame is normalized to be the equivalent of several predicted frames. In order to reduce the complexity and computational overhead, this example bit allocation formula does not take into account the spatial and temporal complexity of each GOV or GOP. In other embodiments, if sufficient computational and time resources are available, the bit allocation formula takes the temporal and spatial complexity into consideration to provide a still more intelligent bit allocation for each GOV.

[0124] For example, in one embodiment a two-pass encoding process is used that takes into account spatial and temporal complexity. The first pass detects scene changes and collects frame complexity. The second pass performs the actual encoding using complexity guided bit allocation.

[0125] The first pass process will now be described in greater detail. A new GOV or GOP is started from a scene change instance. The two-pass rate control process provides substantially uniform quality for each temporally segmented GOV so that quality variation is better limited to GOV or GOP boundaries. This approach is taken because minimizing quality variation, as measured by the Peak Signal to Noise Ratio (PSNR), the Root Mean Square Error or other image fidelity metric, among different scenes provides less benefit with respect to human visual perception.

[0126] In characterizing the relative frame complexity, the complexity measure in accordance with one embodiment of the present invention is relatively invariant with the quantization parameter (QP) used. In particular, generally the bit count for non-texture information, such as frame headers/syntax and motion vectors, denoted by  $H_i$ , is constant or almost constant in terms of quantization parameter (QP) change. This is in contrast to the bit count of the texture information, which does vary with a change in QP. By way of example, based on the MPEG-4 VM R-Q model [MPEG4VM], if the total number of bits used for coding the current frame  $i$  is  $R_i$ , then the texture bits  $T_i=R_i-H_i$  can be represented as:

$$\frac{R_i - H_i}{M_i} = \frac{a_1}{Q_i} + \frac{a_2}{Q_i^2} \quad \text{Equation 10}$$

WO 02/071639

PCT/US02/06726

[0127] where  $M_i$  is the MAD computed with a motion-compensated residual that is substantially invariant with respect to the QP (i.e., Q), and  $a_1$  and  $a_2$  are Taylor expansion coefficients of texture bits  $T_i$  over QP. The coefficients  $a_1$  and  $a_2$  are normally of the same order, that is, have similar values. As can be seen, the lower the QP, the greater then number of texture bits needed to encode a given frame.

[0128] The complexity measure  $C_{g,i}$  addresses both the motion and texture bit count, and is substantially QP invariant. In one embodiment,  $C_{g,i}$  is defined by the ratio of the texture bit count for a given frame to the average texture bit count, and the ratio of the motion vector bit count for the given frame to the average motion vector bit count, as follows:

$$C_{g,i} = (R_{g,i} - H_{g,i}) / \overline{R_g - H_g} + MV_{(g,i)} / \overline{MV_g} \quad \text{Equation 11}$$

[0129] where the  $MV_{(g,i)}$  is the motion vector bit count for frame(g,i),  $\overline{MV_g}$  is the average motion vector bit count, and  $\overline{R_g - H_g}$  is the average texture bit count. Because the obtained complexity measure  $C_{g,i}$  is substantially QP invariant, the frame complexity can be generated in one pass, that is, with one QP. The calculated frame complexity will be utilized in the second encoding pass as will now be described.

[0130] As illustrated in Figure 9, the rate control process 900 in the second pass consists of a three level hierarchy, scene (GOV or GOP)-level bit allocation 902, frame-level bit allocation 904, and macroblock-level QP adjustment 906 which utilizes the obtained frame complexity values  $C_{g,i}$ .

[0131] Complexity-guided scene, GOV, or GOP, level bit allocation will now be described with reference to Figure 10. To assign bits to each scene, the complexity measure is defined for each GOV or GOP by calculating the average spatial complexity  $\overline{C}^{(g)}$  as follows:

$$\overline{C}^{(g)} = \sum_i C_{g,i} / N_g \quad \text{Equation 12}$$

[0132] Then, the GOV or GOP level recursive bit allocation process 1000 is applied, as illustrated in Figure 10. At state 1002, an initialization process is performed, with the following assignments:

$$\lambda = 0$$



WO 02/071639

PCT/US02/06726

Bit budget  $B_g$  (bit allocation for a given time window corresponding to a certain number of GOVs or GOPs) =  $B$

The initial transmit buffer fullness  $\beta_1 = T_d \times R$

Start from GOV or GOP of index 1, that is, the first GOV or GOP.

[0133] At state 1004, bits are assigned to the scene (GOV or GOP) of index  $g$  according to the following formula:

$$B_{(g)} = \lambda \times (R/F) \times N_{(g)} + (1-\lambda) \times \frac{C_g \times N_g}{\sum_i C_i \times N_i} \times B_r \quad \text{Equation 13}$$

[0134] Where:

$R$  = the channel rate

$F$  = the selected frame rate

$N_{(g)}$  = the number of frames in GOV or GOP of index  $g$

$\lambda$  = the weighting factor between the buffer variation and complexity demands.

and

$$\sum_i C_i \times N_i$$

defines the total scene complexity for the given time window corresponding to the GOVs or GOPs under consideration

[0135] The case of  $\lambda=0$  represents the bit allocation scheme that directly follows the frame complexity, which is preferred if the transmit buffer constraints can be met. The assignment  $\lambda=1.0$  represents the case where the bit budget is evenly distributed without considering the frame complexity. In this case, little pre-loading and a small capacity of the decoder buffer are needed as only the first frame needs to be pre-fetched. The case with  $0 < \lambda < 1.0$  represents a bit-allocation tradeoff between the buffer and the quality constraints.

[0136] At state 1006, the buffer status is inspected with the tentatively assigned bit budget  $B_{(g)}$ , if

$$\beta_{g-1} + B_{(g)} \times (R/F) \times N_{(g)} < \text{MarginFactor1} \times \beta_{\max}$$

where, by way of example,  $\text{MarginFactor1} = 0.8$ , which provides a safe margin (0.8 of the maximum buffer size) for buffer regulation.

[0137] then the allocation is accepted, and the process 1000 proceeds to state 1008. Otherwise, the value of  $\lambda$  is adjusted upwards by a factor, such as 0.1, and the process 1000 proceeds back to state 1004.

WO 02/071639

PCT/US02/06726

[0138] At state 1008 the buffer status is updated as follows:

$$\beta_g = \beta_{g-1} + B_{t(g)} - (R/F) \times N_{t(g)},$$

and the remaining budget  $B_r$  is adjusted as follows:  
 $B_r - B_{t(g)}$ .

[0139] The process 1008 then proceeds back to state 1002 and the value of  $\lambda$  is set equal to 0. The allocation for the next GOV or GOP of index  $g+1$  is then performed.

[0140] The GOV or GOP scene-level bit allocation advantageously suitably allocates the bit budget to each GOV or GOP while meeting both the buffer and the quality constraints. However, to obtain a constant or consistent quality within each GOV or GOP, it is preferable to allocate the bit budget according to frames within each GOV or GOP based on frame complexity and while still meeting buffer constraints. The frame level bit allocation process 904 illustrated in Figure 9 performs such a frame bit allocation process, and is similar to the process 1000 for the GOV or GOP level bit allocation. However, rather than using variables corresponding to a GOP or a GOV, the variables corresponding to frames are utilized. Thus, for frames

$$B_f = \lambda \times (R/F) + (1-\lambda) \times C_{t(g,i)} \times B_g' / (C_g) \quad \text{Equation 14}$$

where:

$B_g'$  = running bits = (Bit allocation for current GOP - used bits for frames already coded)

$C_g'$  = running complexity = (complexity for current GOP - complexity of frames already coded)

$R$  = target bit rate

$F$  = frame rate

[0141] In one embodiment, the macroblock level QP adjustment 906 can be alternatively performed in accordance with the following description. In the scene and the frame level bit allocation processes described above, a safe margin (0.8 of the maximal buffer by way of example) is provided for buffer regulation. To reduce computational complexity, in one alternative embodiment, all the macroblocks are quantized with the same quantization parameter (QP) using a one-pass rate control. The QP value is determined as part of the frame level rate control using the following iterative process:

[0142] If  $B_{actual}(i) > 1.15 * B_f(i)$ , then  $QP_{i+1} = QP_i + 1$ .

If  $B_{actual}(i) \leq 1.15 * B_f(i)$   
 then if  $B_{actual}(i) < 0.85 * B_f(i)$   
 $QP_{i+1} = QP_i - 1$   
 -34-

WO 02/071639

PCT/US02/06726

else  $QP_{i+1} = QP_i$ .  
 //To ensure that  $QP_{i+1}$  is within the valid QP range of 1-31, the following clipping operation is performed  
 $QP_{i+1} = \max(QP_{i+1}, 1)$ ;  
 $QP_{i+1} = \min(QP_{i+1}, 31)$ .

[0143] However, if the buffer is relatively small, then the macroblock level rate control process as described below can be used.

[0144] Suppose  $N_{MB}$  is the number of macroblocks in one frame,  $MAD_k$  is the mean absolute difference of MB  $k$ , and  $QP_{k-1}$  is the QP for a previous macroblock. Then the QP for the current MB  $k$  can be in the range of  $[QP_{k-1}-2, QP_{k-1}+2]$ , as determined in accordance with the following rules:

$$QP_k = \begin{cases} QP_k - 2 & \text{if } R_{k-1} > 1.5 \times B_{k-1} \times (MAD_{k-1} / \sum_{k=1}^{N_{MB}} MAD_n) \text{ else} \\ QP_k - 1 & \text{if } R_{k-1} > 1.25 \times B_{k-1} \times (MAD_{k-1} / \sum_{k=1}^{N_{MB}} MAD_n) \text{ else} \\ QP_k + 2 & \text{if } R_{k-1} < 0.67 \times B_{k-1} \times (MAD_{k-1} / \sum_{k=1}^{N_{MB}} MAD_n) \text{ else} \\ QP_k + 1 & \text{if } R_{k-1} < 0.8 \times B_{k-1} \times (MAD_{k-1} / \sum_{k=1}^{N_{MB}} MAD_n) \text{ else} \\ & QP_k \end{cases} \quad \text{Equation 15}$$

[0145] Optionally, in high noise environment, the macroblock bit allocation process can be disabled or not used so that the decoder can assume that the QP is the same for each macroblock. This helps prevent the decoder from using the wrong QP when portions of a frame have been corrupted or lost.

[0146] After determining the bit budget for each GOV or GOP, a novel rate control process, also referred to as IMP4, is used to meet or substantially meet the calculated bit budget. Conventional MPEG-4 rate control does provide adequate performance for many applications. For example, many conventional rate control processes do not explicitly support multiple scenes. Instead, these conventional rate control processes assume that an entire sequence comprises a single scene, and therefore fail to provide for satisfactory rate control. By contrast, an embodiment of the present invention takes into account when scene changes occur, and so provides enhanced rate control.

[0147] In one embodiment of the present invention, a self-converging rate control process is used to meet the bit budget of each GOV by adjusting the quantization parameter QP of each frame, where QP is equal to half the quantization step size. In

WO 02/071639

PCT/US02/06726

MPEG-4, the quantizer parameter QP can have 31 values [1-31]. The rate control process determines the QP based on past bit usage, the number of un-coded frame and the rest bits for a given GOV. Thus, if the current bit usage exceeds the assigned bit budget by more than a certain amount or percentage, the quantization parameter, and therefore the quantization step size, are increased. If, instead, the current bit usage falls beneath the assigned bit budget by more than a certain amount or percentage, the quantization parameter, and therefore the quantization step size, are decreased. The process can be performed in accordance with the following example pseudo-code:

```

if (current bit usage > assigned bit budget *Margin1)
    QPnext = min (31, INT(QPcur+StepUpPrcnt* QPcur));
Else if (current bit usage < assigned bit budget*Margin2)
    QPnext = max (1, INT(QPcur-StepDwnPrcnt* QPcur));
End if.

```

Where:

Margin1 is a constant that allows the current bit usage to exceed the assigned bit budget, so that the system has an opportunity to stabilize. For example, Margin1 can be set equal to 1.15, allowing the current bit usage to exceed the assigned bit budget by 15%.

Margin2 is a constant that allows the current bit usage to under-run the assigned bit budget, so that the system has an opportunity to stabilize. For example, Margin2 can be set equal to 0.85, allowing the current bit usage to under-run the assigned bit budget by 15%.

StepUpPrcnt is constant related to how much the quantization parameter is to be increased. For example, StepUpPrcnt may be set equal to 0.1.

StepDwnPrcnt is constant related to how much the quantization parameter is to be increased. For example, StepDwnPrcnt may be set equal to 0.1.

[0148] After encoding each GOV, the actual bit usage  $B_{act}$  is compared with the pre-assigned bit budget  $B_{ass}$ , and if the actual bit usage varies by more than a certain amount or percentage (Margin3, Margin 4) from the budgeted bit usage, the quantization parameter QP for an I-frame (QPI) is adjusted up or down by a certain amount (StepUp, StepDown) or percentage as needed. The quantization parameter adjustment process can be performed in accordance with the following example pseudo-code:

WO 02/071639

PCT/US02/06726

```
If (Bact > Margin3 * Bass)
    QPI = QPI + StepUp;
Else if (Bact < Margin4 * Bass)
    QPI = QPI - StepDown;
End if
```

Where, by way of example:  
Margin3 = 1.2  
Margin4 = 0.8  
StepUp = 2  
StepDown = -2

[0149] The initial value of QPI may be set to 10, while the initial value of QP for a P-VOP may be set to 12. Note that when the quantization parameter QPI for I-VOP is changed, the QP assignment of the following P-frames may also change.

[0150] In one embodiment, the encoder module 106B then performs adaptive motion change detection to efficiently reduce large propagation errors. In particular, adaptive intra refresh (AIR) is used to reduce error propagation in an MPEG data stream by the selective intra-coding of macroblocks in p-frames. Thus, AIR is used to help determine how many macroblocks should be intra-encoded in the detected motion region of a frame. While the performance of macroblock intra-refresh increases error resilience of the compressed bitstream, increasing the number of intra-coded macroblocks correspondingly increase the number of bits used to encode these macroblocks. Further, if there is a fixed bit rate, the quantization error has to increase for the other, non-intracoded, macroblocks. Thus, preferably, bandwidth and the bit error probability (BER) are taken into account to determine the percentage or number of macroblocks that are to be intracoded.

[0151] In addition, the encoder module 106B optionally used Cyclic Intra Refresh (CIR) to encode a predetermined number of macroblocks (MBs) in each frame. Thus, CIR provides periodic intra-refresh to constrain possible error propagation.

[0152] In one embodiment, Cyclic Intra Refresh (CIR) and Adaptive Intra Refresh (AIR) are performed as follows. The number of Intra macroblocks in a VOP is specified by the user in the encoder parameter file. The number of Intra macroblocks in a VOP depends on target bit rate, frame rate, bit buffer usage, channel noise feedback, and other transmission related parameters. The encoder module 106B estimates the amount of motion for each macroblock and selects heavy motion area to be encoded in INTRA mode

WO 02/071639

PCT/US02/06726

to enhance error resiliency. The results of the estimation are recorded a refresh map at the macroblock level.

[0153] An example conventional refresh map 202 is illustrated in Figure 2A. The encoder module 106B refers to the refresh map and selectively determines whether to encode a given macroblock of the current VOP in INTRA mode or not. The estimation of motion is performed by comparing SAD (Sum of the Absolute Difference) and  $SAD_{th}$ . SAD refers to the Sum of the Absolute Differences value between the current macroblock and the macroblock in the same location of the previous VOP. Advantageously, the SAD is already calculated when performing motion estimation. Therefore, the SAD calculation does not have to be repeated as part of the AIR process.  $SAD_{th}$  is used as a threshold value in determining whether a given macroblock is a motion area. If the SAD of the current macroblock is larger than  $SAD_{th}$ , this macroblock is regarded as motion area.

[0154] Once a macroblock is regarded as a motion area, it remains as a candidate motion area until it is encoded in Intra mode a predetermined number of times. In the example refresh map 202 illustrated in Figure 2A, the value for this "predetermined number of times" is set "1," in other embodiments, the predetermined number of times can be set equal to 2, or a higher value. Horizontal scanning is used to resolve among macroblocks that are candidates to be encoded in Intra mode within the moving area as illustrated in map 204 in Figure 2B.

[0155] With reference to Figure 8A, conventional AIR processing, as applied to four sample sequential VOPs, will now be explained in greater detail. The AIR refresh rate, that is, the fixed number of Intra macroblocks in a VOP is preferably determined in advance. In this example, the number of Intra macroblocks in a VOP is set to "2".

[0156] [1] 1st VOP - Figure 8A [a] and [b]

[0157] The first VOP is a scene change frame containing elements 802, 804. Therefore, all macroblocks in the 1st VOP are encoded in Intra mode, as illustrated in Figure 8A [a]. As illustrated in Figure 8A [b], the refresh map is set to "0", where a 0 indicates that an Intra refresh is not to be performed and a 1 indicates that an Intra refresh is to be performed, because the 1st VOP is encoded without reference to a previous VOP.

[0158] [2] 2nd VOP - Figure 8A [c] - [f]

WO 02/071639

PCT/US02/06726

[0159] The 2nd VOP is intercoded as a P-VOP. Elements 802, 804 have moved down one macroblock and to the right by one macroblock. Intra refresh is not performed in this VOP, because all values in the refresh map are still zero, as illustrated in Figure 8A [c]. The encoder module 106B estimates motion of each macroblock. If the SAD for a given macroblock is larger than  $SAD_{th}$ , the given macroblock is regarded as motion area, illustrated by the hatched area in Figure 8A [e]; thus, the refresh map is updated as illustrated in Figure 8A [f], where the refresh map entry corresponding to a motion macroblock is set to 1.

[0160] [3] 3rd VOP - Figure 8A [g] - [k]

[0161] Elements 802, 804 have moved down by an additional macroblock and to the right by an additional macroblock. When the 3rd VOP is encoded, the encoder module 106B refers to the Refresh Map illustrated in Figure 8A [g]. If the refresh map indicates that a macroblock is to be Intra refreshed, the macroblock is encoded in Intra mode, as illustrated by the macroblocks containing an "X" in Figure 8A [h]. The corresponding refresh map value for an intracoded macroblock is decreased by 1 as illustrated in Figure 8A [i].

[0162] If the decreased value is 0, the corresponding macroblock is not regarded as a motion area. Proceeding, the processing is substantially the same as that for the 2nd VOP as illustrated in Figure 8A [j]-[k], where if the SAD for a given macroblock is larger than  $SAD_{th}$ , the given macroblock is regarded as motion area. The refresh map is updated as illustrated in Figure 8A [k], where the refresh map entry corresponding to a motion macroblock is set to 1.

[0163] [4] 4th VOP - Figure 8A [l] - [p]

[0164] The processing is substantially the same as for the 3rd VOP. If a current macroblock has a 1 associated with it in the refresh map, it is encoded in Intra mode as illustrated by the macroblocks containing an "X" in Figure 8A [m]. The corresponding refresh map value for an intracoded macroblock is decreased by 1 as illustrated in Figure 8A [n].

WO 02/071639

PCT/US02/06726

[0165] If the decreased value is 0, the corresponding macroblock is not regarded as a motion area. If the SAD for a given macroblock is larger than  $SAD_{th}$ , the given macroblock is regarded as motion area. The refresh map is updated as illustrated in Figure 8A [p].

[0166] In another embodiment, a novel enhanced AIR process is performed as follows to select which macroblocks are to be intracoded in a predicted frame. An intercode distortion value and an intracode distortion value are calculated, as are an intercode bit rate and an intracode bit rate. Based on a comparison of the calculated intercode distortion value and the intracode distortion value, and on a comparison of the intercode bit rate and the intracode bit rate for each macroblock, a decision is made as to which macroblocks are to be intracoded. The enhanced AIR process will now be described in greater detail.

[0167] In order to select which prediction frame macroblock is to be intracoded, the expected distortion that would result if the macroblock were lost or corrupted is estimated. For a predicted or intracoded macroblock, the distortion can be reduced if the reference macroblock in the prediction is intracoded.

[0168] With reference to Figure 8B, recursive tracking in conjunction with the prediction path can be used to determine the expected distortion of the macroblock. The dashed lines 804B to 818B, 806B to 820B, 820B to 826B, 812B to 822B, 814B to 824B, 822B to 828B, and 828B to 830B, indicate motion vectors (MV) that are part of the encoded bitstream from a macroblock in the previous frame to a macroblock in the current frame. The angled solid lines, such as those from 802B to 818B, 806B to 820B, 810B to 822B, 816B to 824B, 818B to 826B, 824B to 828B, and 826B to 830B, indicate a zero-motion vector, where a lost motion vector is set to zero. Zero motion vectors are used by a decoder in an error condition, wherein the decoder, for error concealment, replaces an unrecoverable macroblock with a corresponding macroblock from a previous frame. Note that this is just one of the available error concealment strategies, which is termed "basic concealment." Other concealment strategies, such as temporal concealment or supplemental motion vector concealment can be performed as well. Optionally, the effects of these other error concealment strategies are considered separately, in parallel, when



WO 02/071639

PCT/US02/06726

performing the recursive tracking. Referring to Figure 8B, "p" is the packet loss probability or rate, and  $q = (1-p)$ .

[0169] While encoding a given current macroblock, the encoder module 106B performs a motion search on the previous frame and locates a macroblock that most closely matches the current macroblock or is otherwise determined to be a good prediction frame. This located macroblock from the previous frame, depicted by a non-hatched circle, such as macroblocks 802B, 806B, 810B, 816B, 818B, 820B, 826B, is called a prediction macroblock. After the motion search is performed, a residual error is calculated and further encoded using the Discrete Cosine Transform (DCT), then quantized using a selected quantization step or quantization parameter (QP), and entropy coded using variable length coding (VLC). The encoded bitstream consists of motion vector information, entropy coded quantized DCT coefficients for the residual error, and corresponding header information.

[0170] When the decoder receives the encoded bitstream, the decoder processes the coded information and reconstructs the macroblocks. When information for a macroblock is missing, which may be due to packet loss or other error conditions, the decoder preferably conceals the corresponding macroblock using one or more error-concealment strategies, such as the basic concealment discussed above. As discussed above, when a macroblock is missing, basic concealment copies a macroblock at the same spatial location from the previous frame. This is equivalent to receiving a zero-motion vector and zero DCT coefficients.

[0171] In order to determine which macroblocks should be intracoded, in one embodiment the encoder system 100 includes corresponding decoder circuitry so that it can mimic the decoder process and reconstruct what the decoder will reconstruct both in the absence of errors, and in the presence of one or more errors, such as a single error affecting just the current macroblock ("MBC"). By way of example, the difference between the error-free reconstruction and the reconstruction assuming one error is termed "concealment error" or EC. EC is defined as follows:

$$[0172] \quad EC = MBQ - MBC \quad \text{Equation 16}$$

[0173] Where MBQ is the error free reconstruction, and MBC is a single error reconstruction

WO 02/071639

PCT/US02/06726

[0174] When a given macroblock is used as a prediction macroblock for the next frame, an error present on the given macroblock will propagate to those macroblocks in the next frame that use the given macroblock for prediction purposes, even when there is no further error in motion vectors and DCT coefficients for those next-frame macroblocks. The mechanism with which error propagates from a macroblock in a given frame to other macroblocks in the next frame is termed "the error propagation model."

[0175] Error attenuation occurs when half-pixel accuracy is used for prediction either in the vertical or horizontal direction or in both the vertical and the horizontal directions. Error attenuation, comparable to a low pass filter, occurs as a result of the low-pass frequency characteristic of the pixel averaging operation applied when half-pixel motion is used. Thus, given the concealment error EC calculated at the encoder system 100B, the propagated error via half-pixel motion in the horizontal direction  $ECh/2$ , the propagated error via half-pixel motion in the vertical direction  $ECv/2$ , and the propagated error via half-pixel motion in the horizontal and vertical direction  $EChv/2$ , can be determined.

[0176] Half pixel interpolation is illustrated in Figure 8D, showing integer pixel locations, half-pixel locations in the horizontal direction, half-pixel locations in the vertical direction, and half-pixel locations in the horizontal and vertical dimension.

[0177] The half-pixel averaging filter that is normally applied to pixel values can be applied to the concealment error, EC, to define four types of propagated error arrays:

[0178]  $EC0 = EC$

[0179]  $ECh/2$  = error through horizontal half-pixel motion  
(value calculated on crosses "x" in Figure 8D)

[0180]  $ECv/2$  = error through vertical half-pixel motion  
(value calculated on diamonds in Figure 8D)

[0181]  $EChv/2$  = error through horizontal and vertical half-pixel motion  
(value calculated on squares in Figure 8D)

[0182] For each of the four error arrays, the corresponding energy, which approximates the error variance under the hypothesis of zero mean, is calculated.

[0183] The four error variances for these four cases can correspondingly be defined as:

(Equation 17)

WO 02/071639

PCT/US02/06726

 $\sigma_{Ec}^2$ ,  $\sigma_{Ech/2}^2$ ,  $\sigma_{Ecv/2}^2$  and  $\sigma_{Echv/2}^2$ 

[0184] The following four transition or strength factors can then be defined as:

$$\gamma_{Ec} = \frac{\sigma_{Ec}^2}{\sigma_{Ec}^2} - 1 = 0, \quad \gamma_{h/2} = \frac{\sigma_{Ech/2}^2}{\sigma_{Ec}^2} - 1, \quad \gamma_{v/2} = \frac{\sigma_{Ecv/2}^2}{\sigma_{Ec}^2} - 1 \quad \text{and} \quad \gamma_{hv/2} = \frac{\sigma_{Echv/2}^2}{\sigma_{Ec}^2} - 1 \quad \text{Equation 18}$$

18

which correspond to the four possible cases of motion for the current macroblock. These quantities are saved, together with the motion vector that is used to encode the current macroblock,  $(m_x, m_y)$ , the initial error energy  $\sigma_{Ec}^2$ , and the coding mode (Intra/Inter), in a table, file or other record.

[0185] The half-pixel horizontal and vertical propagation strength can be approximated as follows:

$$\gamma_{hv/2} = \gamma_{h/2} + \gamma_{v/2} + \gamma_{h/2}\gamma_{v/2} \quad \text{Equation 19}$$

[0186] using the transition factors of half-pixel horizontal and half-pixel vertical motion, thereby reducing the computation time and resources needed to calculate half-pixel horizontal and vertical propagation strength or transition factor. In addition, the propagation strengths should be positive. Therefore, a negative propagation strength result will be rounded or set to zero.

[0187] As illustrated in Figure 8E, a motion vector MV can map macroblocks in the current frame  $\text{Frame}_n$ , aligned with a grid of 16-pixel rows and columns, into 16x16 pixels in the predicted frame  $\text{Frame}_{n-1}$  that are not necessarily aligned on the same grid. Indeed, as illustrated in figure 8E, a macroblock in  $\text{Frame}_n$  can map to portions of up to four macroblocks in the predicted frame  $\text{Frame}_{n-1}$ .

[0188] An error present on one or more of the four possible macroblocks from the previous frame used for prediction for a macroblock in the current frame will be reflected in the macroblock in the current frame. The error relationship can be proportional to the overlap area. For example the error relationship can be proportional or based on the number of pixels that they overlap. Thus, for each macroblock in the current frame, the up to four prediction macroblocks are identified that would be used when encoding the macroblock in Inter mode. Using the corresponding motion vector information, the overlapping area is determined, and a weighting factor equal or related to that area is used

WO 02/071639

PCT/US02/06726

to normalize the overlap area to the total macroblock area, 256 (=16x16) for example, as defined by the following equation:

$$\sigma_v^2(i, j) = w(i, j) \frac{\sigma_v^2(i)}{1 + \gamma_{i,j}}$$

that estimates the expecting distortion on macroblock  $j$  in current frame due to macroblock  $i$  in previous frame. Note that  $\gamma_{i,j}$  is one of the transition factors  $\gamma_{EC}$ ,  $\gamma_{h/2}$ ,  $\gamma_{v/2}$  and  $\gamma_{hv/2}$  defined previously, depending on the type of motion (half or integer pixel motion vector) along horizontal, vertical or both directions and  $w(i, j) = \frac{w1 \cdot h1}{256}$  is the weighting factor relating the area of overlap ( $w1 \times h1$ ) between macroblock  $j$  and macroblock  $i$ . The term  $\sigma_v^2(i)$  is the concealment error  $\sigma_{EC}^2$  for macroblock  $i$ .

**[0189]** With reference to Figure 8B, beginning with the macroblock 830B in current Frame<sub>*n*</sub>, there are two macroblocks in Frame<sub>*n-1*</sub> that may be used by a decoder to recreate macroblock 830B, a macroblock 826B used for normal decoding, and a macroblock 828B used for concealment. Each of the macroblocks 826B, 828B in Frame<sub>*n-1*</sub> may correspond to up to 4 aligned macroblocks, as discussed above. The same "decode or conceal" strategy can be recursively applied for the two macroblocks 826B, 828B in Frame<sub>*n-1*</sub> to locate 4 macroblocks 818B, 824B, 822B, 829B in Frame  $n-2$ , and then reach Frame<sub>*n-3*</sub> with 8 macroblocks 802B, 804B, 806B, 808B, 810B, 812B, 814B, 816B, and so on. Each of the 8 macroblocks in Frame<sub>*n-3*</sub> has a probability of appearing in Frame<sub>*n*</sub> at the current macroblock, if a certain series of errors/packet loss occurs during transmission. The probability of each one of these 8 paths is the product of the corresponding branch probability ( $p/q$ ), where  $p$  is the packet loss probability and  $q = (1-p)$ . The probability of a particular path to the macroblock 830B occurring can be determined by multiplying the  $p$  and  $q$  values along the path. Thus, there exist paths that have probability  $p^2$ , such as those where two packet losses in a row occur, and a path defined by 812B-830B with probability  $p^3$ .

**[0190]** Assuming a relatively small probability of error (e.g. 0.1), the higher-order (with respect to  $p$ ) paths, such as those with a probability of  $p^2$  or  $p^3$ , can be neglected, and Figure 8B can thereby be simplified to the paths illustrated in Figure 8F. The reductions in paths in Figure 8B is based on an assumption that a macroblock that is to be used for concealment is not itself corrupted, that is, the probability is neglected of

WO 02/071639

PCT/US02/06726

multiple error/packet loss on a certain path between two macroblocks. While this assumption may not always be true, it will most often be true.

[0191] Based on this simplified macroblock relationship, the expected distortion for the current macroblock in Frame<sub>n</sub> can be estimated using the propagation model described above. The expected distortion is defined using the following equation:

$$D(n) = p\sigma_{EC}^2(n) + qD'(n-1) \quad \text{Equation 20}$$

[0192] where  $D'(n-1)$  is the expected distortion for the reference macroblocks in Frame<sub>n-1</sub>, as modified by the transition factors to take into account the possible half-pixel motion from Frame<sub>n-1</sub> to Frame<sub>n</sub>. Expanding this formula for the reference macroblock in Frame<sub>n-1</sub>, expected distortion is defined as follows:

$$D(n) = p\sigma_{EC}^2(n) + q\left(p\frac{\sigma_{EC}^2(n-1)}{1+\gamma_{(n-1,n)}} + qD''(n-2)\right) \quad \text{Equation 21}$$

$\gamma_{(n-1,n)}$  is one of the 4 transition factors ( $\gamma_{EC}$ ,  $\gamma_{h/2}$ ,  $\gamma_{v/2}$  and  $\gamma_{hv/2}$ ) for the reference macroblock in Frame<sub>n-1</sub>, depending on the motion vector from Frame<sub>n-1</sub> to Frame<sub>n</sub>. As similarly described above,  $D''(n-2)$  is the expected distortion for the reference macroblocks in Frame<sub>n-2</sub> as modified by the transition factors to take into account the possible half-pixel motion from Frame<sub>n-2</sub> to Frame<sub>n-1</sub> and from Frame<sub>n-1</sub> to Frame<sub>n</sub>. Expanding this term further, the expected distortion is defined as:

$$D(n) = p\sigma_{EC}^2(n) + q\left(p\frac{\sigma_{EC}^2(n-1)}{1+\gamma_{(n-1,n)}} + q\left(p\frac{\sigma_{EC}^2(n-2)}{1+\gamma_{(n-2,n-1)}+\gamma_{(n-1,n)}} + qD'''(n-3)\right)\right) \quad \text{Equation 22}$$

[0193] If Frame<sub>n-3</sub> is an I-frame or if the frame buffer is limited or restricted to 3 frame, then  $D'''(n-3)$  is equal to zero. Otherwise, the same procedure is recursively applied to previous frame macroblocks. Similarly, if an Intra macroblock is encountered during the recursive processes, it is assumed that the distortion is equal to  $p\sigma_{EC}^2$ , because there is no motion vector, and thus no error-propagation term.

[0194] For the above distortion equations, contributions from each one of a maximum of 4 prediction macroblocks in each frame are summed and multiplied by the corresponding weighting factor that relates the area of overlap between each one of these macroblocks with the target macroblock in Frame<sub>n</sub>.

[0195] The information stored for each macroblock of previous frames can be utilized to calculate the expected distortion for each macroblock for the current Frame<sub>n</sub>. Note that this expected distortion is due to errors in transmission and is not correlated to the

WO 02/071639

PCT/US02/06726

distortion due to quantization for each macroblock. Therefore, the expected distortion term needs to be added to the quantization error to determine the total distortion for each macroblock. This total distortion is referred to as "total Inter-mode distortion," or  $D_{INTER}$ , as it relates to Inter mode encoding of macroblocks.

[0196] For Intra mode, the expected distortion reduces to the first term as follows:

$$D(n) = p\sigma_{ec}^2(n) \quad \text{Equation 23}$$

[0197] reflecting the expected error distortion that needs to be added to the corresponding Intra-mode quantization distortion in order to get the "total Intra-mode distortion" for the macroblock, also referred to as  $D_{INTRA}$ .

[0198] A certain number of bits are needed for the Inter mode encoding and the Intra mode encoding, respectively referred to as  $R_{INTER}$  and  $R_{INTRA}$ . The difference between these bit rates,  $\Delta R = R_{INTRA} - R_{INTER}$ , together with the difference in total distortion,  $\Delta D = D_{INTRA} - D_{INTER}$  can be used to select the best coding mode.

[0199] If the encoder has sufficient resources and capability, a full Rate-Distortion optimization can be performed that involves determination of an optimal weighting factor  $\lambda$  to be used for evaluating a cost function for each macroblock, given by:

$$C = D + \lambda R \quad \text{Equation 24}$$

and thus obtain the following Intra/Inter decision rule:

[0200] Choose Intra mode, when

- $\Delta D < 0$ , if  $\Delta R = 0$
- $\frac{\Delta D}{\Delta R} < -\lambda$ , if  $\Delta R > 0$
- $\frac{\Delta D}{\Delta R} > -\lambda$ , if  $\Delta R < 0$

Else select Inter mode

[0201] Note that determination of the optimal  $\lambda$  parameter is optionally achieved by trying all possible QP and  $\lambda$  combinations. The particular combination that results in the least distortion among all combinations that produce bitstreams below or optionally equal to the desired bit rate is then selected.

WO 02/071639

PCT/US02/06726

[0202] Alternatively, the encoder system 100 can first choose macroblocks that satisfy the first of the above listed cases ( $\Delta D < 0$ , if  $\Delta R = 0$ ), since it is applicable to all values of  $\lambda$ , and also all macroblocks that satisfy the condition  $\frac{\Delta D}{\Delta R} \geq 0$ , if  $\Delta R < 0$ , which automatically applies to the third case. Then macroblocks with  $\Delta R > 0$  are grouped together and ordered in increasing order with respect to the ratio  $\frac{\Delta D}{\Delta R}$ . Similarly, the macroblocks with  $\Delta R < 0$  are grouped together and ordered in decreasing order with respect to the same ratio,  $\frac{\Delta D}{\Delta R}$ .

[0203] This is shown by the graph illustrated in Figure 8G, representing the value of the ratio  $\frac{\Delta D}{\Delta R}$  for each macroblock, where "x"s indicate those macroblock with  $\Delta R > 0$  and "o"s indicate those with  $\Delta R < 0$ .

[0204] The "x"s that have the most negative values, or negative values that meet corresponding defined criteria, and the "o"s that have the least negative values, or negative values that meet corresponding defined criteria, are selected as candidates for Intra Refresh. Note that "o"s with a positive value have already been chosen for Intra coding, and "x"s with positive value are excluded altogether as they are automatically intracoded. Experimental results indicates that macroblocks with  $\Delta R > 0$ , indicated by an "x", are the most common ones, because generally Intra mode costs more, as measured in bits, compared to Inter mode. Thus, optionally only macroblocks with  $\Delta R > 0$  will be considered for Intra refresh. The so-called refresh rate, specifying how many additional macroblocks are to be encoded in intra mode, then dictates how many of the candidate macroblocks are eventually chosen.

[0205] A less resource intensive process is to calculate the expected distortion for each macroblock due to transmission error, while ignoring or excluding quantization error. Then, the differential between the expected distortion for Intra and Inter mode can be used as the criterion for selecting macroblocks to be Intra coded by ordering them according to this criterion.

[0206] Thus, Adaptive Intra Refresh (AIR) can be used to help determine how many macroblocks should be intra-encoded in the detected motion region of a frame. AIR can be enabled and disabled in the encoder parameter file using an AIR bit set by a user and

WO 02/071639

PCT/US02/06726

read by the encoder module 106B. When AIR is enabled, the user also specifies another parameter, the AIR refresh rate. The AIR refresh rate determines how many macroblocks should be intra-coded in the detected motion region of one frame. Adaptive motion change detection can efficiently reduce the large propagation error, even when the error occurs in the motion region.

[0207] Figure 8H illustrates an embodiment of the E-AIR process. At state 802H, the motion vector or vectors for the current macroblock of interest in Frame (n) are received. At state 804H the motion vector is used to locate which macroblocks from a previous Frame (n-1) are to be used in predicating the current macroblock. At state 806H a determination is made as to how much, in terms of area or pixels, of each of the located macroblocks in Frame (n-1) will be used in generating the current macroblock. At state 808H, the error variances ( $\sigma_{Ec}^2$ ,  $\sigma_{Ech/2}^2$ ,  $\sigma_{Ecv/2}^2$ ,  $\sigma_{Echw/2}^2$ ) are calculated, including the overlap weighting ( $w(i, j) = \frac{1}{1 + \gamma_{i,j}}$ ). At state 808H the propagation strength transition quantities are calculated based on the error variances. At state 812H, the Intra Error Distortion  $D_{INTRA}$  for Frame (n) is calculated. At state 814H, the Inter Error Distortion  $D_{INTER}$  for Frame (n) is recursively calculated. As discussed above, the recursive calculation can include the error distortion and quantization distortion from previous frames, such as Frame (n-1), Frame (n-2), and so on, whose errors may propagate to Frame (n). The recursion may be limited to a predetermined number of frame generations, until all or a predetermined amount of the frame buffer is being used, or the recursion may stop when an Intra frame is reached.

[0208] At state 816, the value DeltaD is calculated by taking the difference between  $D_{INTRA}$  and  $D_{INTER}$ , or by otherwise comparing  $D_{INTRA}$  and  $D_{INTER}$ . At state 818H the bit quantity or bit rate  $R_{INTRA}$  and  $R_{INTER}$  for intracoding Frame (n) and for intercoding Frame (n) respectively are determined. At state 820H a comparison of  $R_{INTRA}$  and  $R_{INTER}$  is made by calculating the difference DeltaR. At state 822H the decision to intracode or to intercode is made based on DeltaR, DeltaD and Lambda using the illustrated criteria. Alternatively, those macroblocks having a DeltaD may be chosen for intracoding. For example, after DeltaD is calculated for all of Frame (n)'s macroblocks, the two macroblocks having the largest DeltaD are intracoded. The macroblock intracode selection may also be based on a cost calculation where the Cost = Rate + lambda\*D, or D + lambda\*R, and choose the highest N (=AIR rate)



WO 02/071639

PCT/US02/06726

[0209] Figure 8C illustrates experimental results comparing the use of Cyclic Intra Refresh, trace 802C, with the use of the enhanced AIR method described immediately above, trace 804C. The overall gain is approximately 1 dB in the PSNR. In one embodiment, the additional computational load is approximately 10%.

[0210] In order to utilize AIR more effectively, conventional Cyclic Intra Refresh (CIR) is combined with AIR. The number of the Intra Refresh macroblocks in a VOP is defined as the summation of the AIR\_refresh\_rate and the CIR\_refresh\_rate. AIR\_refresh\_rate macroblocks are encoded in AIR mode and CIR\_refresh\_rate macroblock are encoded in the conventional CIR mode. These values are user definable in the encoder parameter file. When the channel degrades, higher CIR and AIR rates should preferably be assigned. In addition, when the distance between I-frames is large, higher CIR and AIR rates should preferably be assigned. These rates are preferably varied adaptively with changing channel conditions as well as with the coding parameters to improve the tradeoff between error resilience and coding efficiency.

[0211] The encoder parameter file specifies many different encoding parameters, including those discussed above. The encoder parameter file can be used in conjunction with the preprocessing module output by reading the frame-type file, which specifies the encoding type are determined by preprocessing process described above. By way of example, the encoder parameter file includes fields to enable/disable AIR, CIR, and SMC, to specify the AIR and CIR refresh rates, and a flag used to enable or disable the inclusion of two I-frames at the beginning of each scene, GOV or GOP.

WO 02/071639

PCT/US02/06726

The encoder parameter file has the following parameters or fields:

Version	Version number and/or name
Source.Width	specifies frame width
Source.Height	specifies frame height
Source.FirstFrame	specifies the first frame to be encoded (counting from 0)
Source.LastFrame	specifies the last frame to be encoded
Source.Directory	Directory to store the original source of sequence without trailing "\"
Source.SamplingRate	Allows sub-sampling the original source based on this sampling rate parameter
Output.Directory.Bitstream	Output bitstream directory
Output.Directory.DecodedFrames	Directory to put the reconstructed frames from the encoder (encoder also performs decoding)
RateControl.Type [0]	What type of rate control- one of "None" (maintain constant QP), "MP4" (for IMP4), "TMS" (for Test Model 5);
RateControl.BitsPerVOP [0]:	bit budget for the entire sequence
Quant.Type [0]	One of "H263", "MPEG". For low bit rate communications, H263 is preferred.
GOV.Enable [0]	GOV header present or not
GOV.Period [0]	Number of VOPs between GOV headers
Texture.QuantStep.IVOP [0]	Quantization Parameter (QP) for I-VOP; not affected by rate control
Texture.QuantStep.PVOP [0]	QP for P-VOP if rate control is disabled
Texture.QuantStep.BVOP [0]	QP for B-VOP if rate control is disabled
Motion.PBetweenCount [0]	In case of multiple scenes, and in the presence of a frame-type file, the encoder ignores this parameter. Else, the length of a GOP is specified before source subsampling. A negative value means one GOP for the whole sequence.
Motion.ReadWriteMVs[0]	One of "Off", "Read", "Write"
Motion.ReadWriteMVs.FileName [0]	Filename for Read/write MV from/to files
ErrorResil.RVLC.Enable [0]	Enable or disable RVLC - 0: disable; 1: enable
ErrorResil.DataPartition.Enable[0]	Enable or disable data partitioning
ErrorResil.VideoPacket.Enable[0]	Enable or disable introduction of resync markers in video packet (VP)
ErrorResil.VideoPacket.Length[0]	If VideoPacket enables, the size of VP in bits - select based on the target bit rate
ErrorResil.SMC.Enable	Enable/disable second motion compensation (SMC). When =1, only 1 PVOP; when =2, for all PVOP
ErrorResil.AIR.Enable	Enable/disable adaptive intra refresh (AIR)
ErrorResil.CIR.Enable	Enable/disable cyclic intra refresh (CIR)
ErrorResil.AIR.Rate	Added for specifying the AIR rate. an AIR rate of 2, for example, may be used.
ErrorResil.CIR.Rate	When enabling CIR (see above), specifies the CIR rate (Macroblocks per VOP). A CIR rate of 2, for example, may be used.
ErrorResil.2I.Enable	Added to enable/disable double I-frame coding in each GOV for enhanced error-resilience

Table 1 Encoder parameter specification.

[0212] To further increase error resiliency, a Header Extension Code (HEC) is included by the encoder module 106B in every packet in a sequence of video packets or in every video packet, and not just on the first video packet following the VOP header as with conventional encoders. This better ensures that even if a packet is lost or corrupted,

WO 02/071639

PCT/US02/06726

subsequent packets can still be decoded and used. Further, even typical conventional decoders will be able to handle the inclusion of the enhanced use of HECs as the use of additional HECs is compatible with the MPEG-4 bitstream syntax. Adding a header, including sequence information, to all packets increases overhead by only about 40 bits per packet, or about 0.2%, but results in a noticeable improvement in decoding.

[0213] Further, Second-order Motion Compensation (SMC) is optionally provided to enhance error resiliency. The SMC process is performed by the encoder module 106B and generates supplemental motion vectors so that each predicted frame can be predicted separately from two preceding frames. Sequence 502 of Figure 5 illustrates the SMC process, in which  $k_{th}$  frame has motion vectors from both from the  $(k-1)_{th}$  frame and the  $(k-2)_{th}$  frame. Therefore, even if the motion vectors from the  $(k-1)_{th}$  frame are corrupted, or the  $(k-1)_{th}$  frame is itself corrupted, the  $k_{th}$  frame can still be predicted from the  $(k-2)_{th}$  frame using the corresponding motion vectors. Thus, by inserting the redundant motion vectors, also termed second-order motion vectors, from the  $(k-2)_{th}$  frame to  $k_{th}$  frame, the scene image quality at the decoder-side will be better protected from transmission errors. For example, even if all the information for  $k_{th}$  frame is corrupted during transmission, the use of SMC can effectively suppress error propagation by excluding  $k_{th}$  frame from being used in any later prediction as illustrated in Figure 5 by sequence 504.

[0214] To perform SMC, a frame buffer is included in the encoder module 106B to store the previously decoded frame at time  $(t-2)$ . This previously decoded frame is used to calculate the second order motion vectors. In one embodiment, these redundant motion vectors are not used in the encoder to produce residuals. The decoder uses the second order motion vectors when the bitstream is corrupted during the transmission and the first order motion vectors or corresponding frame is corrupted. Otherwise, the second order motion vectors need not be used in the decoder. Optionally, full, unrestricted motion search can be performed in order to determine these second-order motion vectors. The information regarding motion between frames  $(t-2)$  and  $(t-1)$  and between frames  $(t-1)$  and  $(t)$  can be combined in order to estimate these second-order motion vectors. For this reason, the SMC data can optionally be included, via the "user data mechanism", as explained below, for the first P-VOP following a scene change or for each P-VOP. The advantage of having SMC on only the first P-VOP is that bandwidth is not wasted when

WO 02/071639

PCT/US02/06726

there is no error, thereby providing better coding efficiency. However, providing SMC for every P-VOP or for many P-VOPs enhances the robustness and decoding ability of the decoder, especially for cases of severe error conditions.

**[0215]** An additional video packet, referred to as "User data" or an "SMC" video packet, for each P-VOP is used to transmit these second-order motion vectors. This packet contains, in the same predictive fashion and using the same variable-length codes as in the standard motion vectors, a motion vector for each macroblock or selected macroblocks of the current P-VOP. An HEC is included in this special SMC video packet, which allows the SMC video packet to be decoded even if other packets for this P-VOP are lost. In one embodiment, this packet is positioned in the bitstream at the end of each P-VOP. A user can enable or disable the use of SMC by setting to 1 or 0 the corresponding option in the encoder parameter file. Figure 6 illustrates an example packetized bitstream showing the relative position of packets in the bitstream, including the SMC packet 602.

**[0216]** Optionally, in order to make SMC packets compliant with the MPEG-4 syntax, a so-called "User data start code" (hex code B2) or the like, including other unique identifier codes to be assigned in the future by MPEG-standards committee and the like, precedes the HEC and motion vector information. The user data start code signals standard decoders not capable of using the second order motion vectors to ignore all bits following it until the next start code in the bitstream, which in this will be a VOP start code. In one embodiment, the encoder includes a unique 16-bit identifier in order not to confuse the SMC user data extensions with data that other people may decide to include in the bitstream following the same convention.

**[0217]** To further enhance error resiliency, two consecutive I-frames are inserted upon a scene change, even if the second of the two frames is not a scene change frame and has low enough relative motion that it would normally be intercoded, as illustrated in Figure 7. That is, the scene change frame 702 is intracoded, and the next frame 704 is then automatically intracoded. Thus, the loss of one I-frame will not prevent the decoder from decoding the predicted frames 706, 706, 710, 712 that are predicted from frame 702 as well as frame 704. Frames subsequent to the second I-frame may be encoded as intracoded frames, such as P-frames or B-frames.

**[0218]** The use of two consecutive I-frames advantageously prevents the prediction of a frame in the current scene using scene content from other scenes, without

WO 02/071639

PCT/US02/06726

degrading the performance of the SMC. Because the first two consecutive frames in a scene are intracoded, neither first nor second-order motion vectors are inserted into the I-frames.

[0219] The inclusion of the consecutive I-frames can be under the control of the preprocessing module 102B which can designate both a scene change frame and the next frame as intracoded frames. Alternatively, the encoder module 106B can automatically intracode a frame following a frame designated as an intracoded frame by the preprocessing module 102B.

[0220] While the insertion of two consecutive I-frames increases the bit-budget and thus decreases coding efficiency for a certain transmission bit-rate, in error-prone environments this inefficiency is more than compensated for by the additional error-resilience thereby provided. A "consecutive I-frame" flag is provided in the encoder parameter file that can be independent of the SMC flag. Advantageously, the presence of two consecutive I-frames at the beginning of each scene can be used for the decoder to conceal transmission errors more efficiently, even if the SMC mode is not turned on, or when it is turned on for just the first P-VOP following the (double) I-frame after a scene change.

[0221] Adaptive Intra Refresh (AIR) is optionally supported by the encoder module 106B as a by-product of SMC. This mode, enabled when selecting SMC for just the first P-VOP or for every P-VOP, encodes in INTRA mode those macroblocks that have as prediction macroblocks from frames (t-1) and (t-2) two significantly different macroblocks, as measured by the MAD distance measure. An example threshold is 20. Thus, if the MAD between the two prediction macroblocks for a given macroblock in the current frame is greater than 20, this macroblock is intracoded.

[0222] The encoder module 106B also performs general encoder functions, such as motion estimation, residual calculation, and the like. The encoder output can be stored for later transmission or can be transmitted in substantially real-time to a receiving terminal, such as a cellular phone, containing an appropriate decoder.

[0223] To increase error resilience, in one embodiment `Intra_dc_vlc_thr` is set to "0", so that all DC coefficients are coded using DC VLC in a frame or VOP. In addition, the `ac_pred_flag` may be disabled for all Intra macroblocks. Both these options are permitted by the syntax and so are supported by standard decoders, and can result in higher

WO 02/071639

PCT/US02/06726

quality for the case of error in transmission. This improvement can be on the order of 0.1-0.2dB in PSNR. In particular, when data partitioning is enabled, the DC coefficient of each 8x8 DCT block of an INTRA macroblock can either be coded together with the 63 AC DCT coefficients, using what is known as an "INTER VLC" table, or separately, using what is known as an "INTRA VLC" table.

[0224] Using the INTRA VLC table results in separating the corresponding bits for the DC coefficient from those of the rest 63 AC coefficients as follows:

[0225] For an I-VOP, where the macroblocks are all intracoded, the DC data bits are located before the DC marker (DCM), together with the header bits, while the data bits for AC coefficients are placed after the DC marker.

[0226] For a P-VOP, the DC data bits are placed immediately after the Motion Marker (MM), together with other crucial or important information, and the data bits for AC coefficients follow.

[0227] This separation of DC and AC information enhances error resilience, since the DC data can be decoded and trusted even for corrupted packets, if the DCM/MM markers are correctly hitting during the decode process. Further, with respect to P-VOPs, Reversible Variable Length Code (RVLC) forward/backward decoding can reveal at least one good first part.

[0228] To control whether the DC data is coded together or separate from AC coefficients, the flag called "intra\_dc\_vlc\_thr" is used for each VOP that maps, according to the QP value, each macroblock. Setting this value = 0 means that all macroblocks, regardless of their QP should use an INTRA DC table and thus separate DC data from AC coefficients. This is a standard syntactic element in MPEG-4 and thus supported by standard decoders.

[0229] The ac\_pred\_flag is another option that indicates whether for a specific block of an INTRA macroblock, the top row and first column DCT coefficients, are coded independently or differentially with respect to the neighboring blocks. To enhance error resilience it is preferable to set ac\_pred\_flag to 0.

[0230] In one embodiment, error-correction is supported at the source level by using Forward Error Correction (FEC). In particular, Bose-Chaudhuri-Hocquenghem (BCH) codes, including Reed-Solomon, are supported. As is well known in the art, BCH is an error detection and correction technique based on Cyclic Redundancy Code. For any

WO 02/071639

PCT/US02/06726

positive integers  $m$ ,  $m > 3$ , and  $t < 2^{m-1}$ , there is a binary BCH code with a block length  $n$  equal to  $2^m - 1$  and  $n - k < mt$  parity check bits, where  $k$  is the number of information bits. The BCH code has a minimum distance of at least  $2t + 1$ . Each binary BCH code  $(n, k, t)$  can correct up to  $t$  bit errors, and thus it is also referred to as a  $t$ -error-correcting code.

[0231] Different block sizes may be used. In one embodiment, a block size of 511 is used. FEC is performed at a packetizer level of the elementary video bitstream, which can be considered as source-level error correction. By contrast, channel level error-correction introduces redundancy at the bit-level after multiplexing.

[0232] FEC can provide significant error resilience, at the cost of some bit budget. Figure 11 illustrates an example graph of Forward Error Correction overhead vs. average BER correction capability. As illustrated, there is a close relation between FEC redundancy and error correcting capability, which is a strong indicator of error resilience. Preferably, at least double the expected BER is provided for.

[0233] In one embodiment of the present invention, rather than apply FEC to all packet data, a more efficient process is performed that reduces the number of error correction bits generated as compared to conventional approaches, while still providing significant error correcting capability. One embodiment of the FEC process optionally generates FEC bits only for selected portions of the packets, and in particular, for those portions that are considered more essential or important for purposes of reproducing a frame sequence by the decoder. In addition, the FEC process provides a systematic code, that is, the FEC correction or parity bits are separate from the original uncoded data bits. Thus, even if all the FEC bits are lost, the original selected portions of the packet are still potentially decodable. Further, in one embodiment the FEC data is encoded and transmitted in an MPEG-4 compliant manner as explained below. Thus, even if a decoder that is not equipped to process the FEC data receives the FEC packet, the decoder will still be able to process the frame motion and texture data.

[0234] In particular, FEC is efficiently applied to important data, such as motion vectors, DC coefficients and header information, and FEC bits are not generated for unimportant or less important data. This more important data may be located in a packet between a packet resync field and a motion marker. In particular, for a given frame or VOP, the selected bits targeted for FEC coding are concatenated together with those from other frame packets and the FEC code bits are generated for the concatenated bits.

WO 02/071639

PCT/US02/06726

[0235] In one embodiment, rather than including the FEC bits in the same packet or packets as the frame data, for a given frame or VOP, the resulting FEC bits are placed in an additional packet after the regular frame or VOP packets to ensure MPEG compatibility.

[0236] In addition, to better allow the decoder to recover in cases where a data packet is lost or has a motion marker missing, for each standard packet, in the FEC packet a packet identifier is stored in association with a corresponding value indicating how many bits and/or which bits were used to generate FEC bits.

[0237] In order to maintain compatibility with reference or conventional MPEG-4 decoders, this additional FEC packet further includes a user data identifier code, "user\_data\_start\_code," used to identify user defined data, and as such will be ignored by conventional decoders not equipped to process the FEC packet. In addition, where there is no error, the FEC packet will not be used by decoders equipped to handle the FEC bits. But when errors do occur, FEC decoding will help recover data that will allow for decoding even under severe error conditions.

[0238] Thus, as described above, by providing error resiliency and enhanced compression, embodiments of the present invention advantageously enable the transmission of video information even in low-bit rate, high noise environments. For example, embodiments of the present invention enable video transmission to be successfully performed over cellular networks and the like.

[0239] Although this invention has been described in terms of certain preferred embodiments, other embodiments that are apparent to those of ordinary skill in the art are also within the scope of this invention.



WO 02/071639

PCT/US02/06726

WHAT IS CLAIMED IS:

1. A method of enhancing error resiliency in an encoded video transmission, the method comprising:
  - receiving an indication that a first frame corresponds to a scene change;
  - intracoding the first frame at least partly in response to receiving the indication that the first frame corresponds to a scene change;
  - automatically intracoding the next consecutive frame; and
  - transmitting the intracoded first frame and the intracoded next consecutive frame.
2. The method as defined in Claim 1, wherein the transmission of the intracoded first frame and the intracoded next consecutive frame is MPEG-4 compliant.
3. The method as defined in Claim 1, wherein the first frame is designated as an I-frame.
4. The method as defined in Claim 1, wherein the next consecutive frame is designated as an I-frame.
5. The method as defined in Claim 1, wherein the next consecutive frame is not a scene change frame.
6. The method as defined in Claim 1, wherein a frame immediately subsequent to the next consecutive frame is intercoded.
7. The method as defined in Claim 1, wherein the indication that that first frame corresponds to a scene change is read from a file.
8. A video coding apparatus, comprising:
  - a first instruction stored in a processor readable memory, the first instruction configured to receive an indication that a first frame is to be intracoded;
  - a second instruction stored in processor readable memory, the second instruction configured to intracode the first frame at least partly in response to the indication; and
  - a third instruction stored in processor readable memory, the third instruction configured to automatically intracode the frame immediately after the first frame as a result of the indication that the first frame is to be intracoded.
9. The video coding apparatus as defined in Claim 8, wherein the video coding apparatus is an integrated circuit.

WO 02/071639

PCT/US02/06726

10. The video coding apparatus as defined in Claim 8, wherein the video coding apparatus is a cellular phone.
11. The video coding apparatus as defined in Claim 8, wherein the video coding apparatus is a desktop computer.
12. The video coding apparatus as defined in Claim 8, wherein the video coding apparatus is a personal digital assistant.
13. The video coding apparatus as defined in Claim 8, wherein the first frame is a scene change frame.
14. The video coding apparatus as defined in Claim 8, further comprising a file used to store the indication that the first frame is to be intracoded.
15. A method of encoding video frames, the method comprising:
  - receiving a first instruction to intracode a first frame;
  - intracoding the first frame in response to the first instruction; and
  - intracoding a second frame as a result of the first instruction to intracode the first frame.
16. The method as defined in Claim 15, further comprising transmitting the intracoded first frame and the intracoded next frame.
17. The method as defined in Claim 15, wherein the first frame is a scene change frame.
18. The method as defined in Claim 15, wherein the second frame has an activity level that would cause it to be intercoded in the absence of the first instruction.
19. The method as defined in Claim 15, further comprising coding a third frame based at least in part on its relative visual motion, the third frame immediately subsequent to the second frame.
20. An encoding apparatus, comprising:
  - a means for receiving a first indication to intracode a first frame;
  - a means for intracoding the first frame in response to the first instruction;
  - and
  - a means for intracoding a second frame as a result of the first instruction to intracode the first frame.
21. The encoding apparatus as defined in Claim 20, further comprising a means for providing the first intracoded frame and the second intracoded frame.

WO 02/071639

PCT/US02/06726

22. The encoding apparatus as defined in Claim 20, wherein the encoding apparatus is included in an integrated circuit.

23. A method of detecting a scene change in a digital video sequence having a plurality of frames, the method comprising:

calculating a first root mean squared (RMS) value for a first frame relative to a second frame and the second frame relative to a third frame;

calculating a first mean absolute difference (MAD) value for the first frame relative to the second frame;

determining if the first RMS value meets a first criterion;

determining if the first MAD value meets a second criterion; and

designating the second frame as a scene change frame at least partly in response to determining that both the first RMS value meets the first criterion and the first MAD value meets the second criterion.

24. The method as defined in Claim 23, wherein the first RMS value is color weighted.

25. The method as defined in Claim 23, further comprising storing an I-frame designation in a file for the second frame and storing a P-frame designation for a third frame in the file.

26. The method as defined in Claim 23, wherein the first RMS value is based at least in part on pixel luminance values and chrominance values of the first and second frames.

27. The method as defined in Claim 23, wherein the first RMS value is defined as

$$RMS(F_i, F_k) = \frac{\alpha}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^w \sum_{y=1}^h |Y_i(x, y) - Y_k(x, y)|^2} +$$

$$\frac{2\beta}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^{w/2} \sum_{y=1}^{h/2} |U_i(x, y) - U_k(x, y)|^2} +$$

$$\frac{2\gamma}{\alpha + \beta + \gamma} \sqrt{\frac{1}{w \times h} \sum_{x=1}^{w/2} \sum_{y=1}^{h/2} |V_i(x, y) - V_k(x, y)|^2}$$

where  $F_i$  is the first frame,  $F_k$  is the second frame,  $F(x, y)$  denotes the  $(x, y)^{th}$  pixel in frame  $F$ ,  $w$  is a frame width and  $h$  is a frame height,  $Y(x, y)$  corresponds to a pixel luminance value, and  $U(x, y)$  and  $V(x, y)$  corresponds to

WO 02/071639

PCT/US02/06726

chromaticity components, and  $\alpha$ ,  $\beta$  and  $\gamma$  are weighting coefficients for luminosity, chromaticity-blue and chromaticity-red components correspondingly.

28. The method as defined in Claim 27, wherein  $\alpha = \beta = \gamma = 1$ .

29. The method as defined in Claim 23, wherein the first MAD value is calculated using luminance value and excluding chromaticity components.

30. The method as defined in Claim 23, wherein the first criterion is a first threshold and the second criterion is a second threshold.

31. A method of detecting a scene change in a digital video sequence, the method comprising:

calculating a second temporal derivative RMS value for a first frame relative to a second frame and the second frame relative to a third frame; and

based at least in part on the second derivative value, determining that the second frame is a scene change frame.

32. The method as defined in Claim 31, wherein the determination that the second frame is a scene change frame is further based upon a mean absolute difference value calculated using at least luminosity information for the first and the second frames.

33. The method as defined in Claim 31, wherein the determination that the second frame is a scene change frame is further based upon both an RMS value meeting a first criterion and the second temporal derivative RMS value meeting a second criterion.

34. The method as defined in Claim 31, wherein the second temporal derivative RMS value is greater than or equal to a first threshold.

35. The method as defined in Claim 31, wherein the second frame is designated as a scene change frame when the second temporal derivative RMS value is negative and has a greater absolute value than a first value.

36. The method as defined in Claim 31, further comprising calculating a first RMS value, wherein the first RMS value is color weighted and the second temporal derivative RMS value is based only on temporal components.

37. The method as defined in Claim 31, wherein the second temporal derivative RMS value is equal to  $(RMS(F_{i-1}, F_i) - 2RMS(F_i, F_{i+1}) + RMS(F_{i+1}, F_{i+2}))$ , where  $F_{i-1}$  is the first frame,  $F_i$  is the second frame,  $F_{i+1}$  is a third frame, and  $F_{i+2}$  is a fourth frame.

38. An apparatus for identifying a scene change in a video sequence, the apparatus comprising:

WO 02/071639

PCT/US02/06726

a first instruction stored in processor readable memory, the first instruction configured to calculate a first root mean squared (RMS) value for a first portion of a video sequence relative to a second portion of the video sequence;

a second instruction stored in processor readable memory, the second instruction configured to calculate a second temporal derivative RMS value; and

a third instruction configured to cause the second portion of the video sequence to be intracoded based at least in part on the second derivative RMS value.

39. The apparatus as defined in Claim 31, wherein the third instruction is further configured to cause the second portion of the video sequence to be intracoded based upon a mean absolute difference value calculated using at least luminosity information for first portion and the second portion.

40. The apparatus as defined in Claim 31, wherein the third instruction is further configured to cause the second portion of the video sequence to be intracoded at least partly in response to both the RMS value meeting a first criterion and the second temporal derivative RMS value meeting a second criterion.

41. The apparatus as defined in Claim 31, wherein the third instruction is further configured to store in processor readable memory a flag indicating that the second portion is to be coded as an I-frame.

42. The apparatus as defined in Claim 31, wherein the first portion is a first frame and the second portion is a second frame.

43. The method as defined in Claim 31, wherein the third instruction identifies the first portion as including a scene change when the second derivative value is negative and has a greater absolute value than a first value.

44. The method as defined in Claim 31, wherein the first RMS value is color weighted.

45. The apparatus as defined in Claim 31, wherein the apparatus is an integrated circuit.

46. A method of determining which portions of a video sequence are to be intracoded, the method comprising:

calculating a first root mean squared (RMS) value for a first portion of the video sequence;

WO 02/071639

PCT/US02/06726

calculating a first mean absolute difference (MAD) value for the first portion of the video sequence;

determining if the first RMS value meets a first criterion;

determining if the first MAD value meets a second criterion;

determining if the first MAD value meets a third criterion; and

causing an intracoding operation to be performed at least partly in response to at least two of the first, second and third criteria being met.

47. The method as defined in Claim 46, wherein the third criterion is that the MAD value is a local maximum.

48. The method as defined in Claim 46, wherein the first portion of the video sequence includes a first frame.

49. The method as defined in Claim 46, wherein the first portion of the video sequence includes a first GOV.

50. The method as defined in Claim 46, wherein the first portion of the video sequence includes a first GOP.

51. A scene change detection apparatus, comprising:

an RMS circuit having a first frame information input and an RMS output, the RMS circuit configured to provide at the RMS output a value corresponding to root mean squared differences information between at least two frames based on frame information received on the first frame information input;

a MAD circuit having a second frame information input and a MAD output, the MAD circuit configured to provide at the MAD output a value corresponding to mean absolute differences information between at least two frames based on frame information received on the second frame information input; and

an evaluator circuit coupled to the RMS output and the MAD output, the evaluator circuit configured to detect a scene change frame based at least on the value corresponding to root mean squared differences information and the value corresponding to mean absolute differences information, and to provide a scene change designation.

52. The apparatus as defined in Claim 50, further comprising a second derivative RMS circuit having a third frame information input and a second derivative RMS output, the second derivative RMS circuit configured to provide at the second derivative RMS

WO 02/071639

PCT/US02/06726

output a value corresponding second derivative root mean squared differences information based on frame information received on the third frame information input, wherein the second derivative RMS output is coupled to the evaluator circuit.

53. A bit allocation method for use with video sequences, the method comprising :  
receiving a clip bit budget for at least a first clip;  
determining a scene quantity for the first clip;  
determining a quantity of predicted frames in a first scene in the first clip;

and

calculating a bit budget for the first scene based at least in part on the scene quantity for the first clip, the first clip bit budget, a quantity of intracoded frames in the first scene, and the quantity of predicted frames in the first scene.

54. The bit allocation method as defined in Claim 53, wherein each scene in the first clip begins with an intracoded frame.

55. The bit allocation method as defined in Claim 53, wherein the bit budget is calculated based at least in part by multiplying the clip bit budget by a sum of the number of predicted frames and a first constant, and dividing by a sum of the quantity of intracoded and predicted frames in the clip and the number of clip scenes multiplied by a second constant.

56. The bit allocation method as defined in Claim 53, wherein the first scene is a GOV.

57. The bit allocation method as defined in Claim 53, wherein the first scene is a GOP.

58. The bit allocation method as defined in Claim 53, wherein the predicted frames are P-VOPs.

59. The bit allocation method as defined in Claim 53, further comprising adjusting a quantization parameter for a first predicted frame in the first scene based on the bit budget for the first scene and current bit usage.

60. The bit allocation method as defined in Claim 59, wherein the adjustment of the quantization parameter is limited to a first range.

61. The bit allocation method as defined in Claim 53, further comprising adjusting a quantization parameter for a first predicted frame in the first scene upwards at least partly in

WO 02/071639

PCT/US02/06726

response to determining that current bit usage is greater than a value related to the bit budget for the first scene.

62. The bit allocation method as defined in Claim 53, further comprising adjusting a quantization parameter for a first predicted frame in the first scene downwards at least partly in response to determining that current bit usage is less than a value related to the bit budget for the first scene.

63. The bit allocation method as defined in Claim 53, further comprising adjusting a quantization parameter for a macroblock based at least in part on a channel rate.

64. A bit allocation method for use with video clip scenes, the method comprising :  
calculating a bit budget for a first scene;  
calculating bit budgets for corresponding frames, including at least a first frame, within the first scene; and  
calculating bits budgets corresponding to macroblocks within the first frame.

65. The bit allocation method as defined in Claim 64, wherein the first scene is a GOP.

66. The bit allocation method as defined in Claim 64, wherein the first scene is a GOV.

67. The bit allocation method as defined in Claim 64, wherein the bit budget for the first scene is based at least in part on a complexity determination for the first scene.

68. The bit allocation method as defined in Claim 64, wherein the bit budget for the first scene is based at least in part on a complexity determination for the first scene and an average complexity of a plurality of scenes.

69. The bit allocation method as defined in Claim 64, wherein the bit budget for the first scene is based at least in part on a buffer status.

70. The bit allocation method as defined in Claim 64, wherein the bit budget for the first frame is based on quantization parameter-invariant criteria.

71. The bit allocation method as defined in Claim 64, wherein the bit budget for the first frame is based at least in part on how many texture bits are in the first frame and how many motion vector bits there are for the first frame.

72. The bit allocation method as defined in Claim 71, wherein the bit budget for the first frame is further based on a mean absolute difference value corresponding to the first frame relative to a second frame.



WO 02/071639

PCT/US02/06726

73. The bit allocation method as defined in Claim 64, wherein the bit budget for the first frame is based at least in part on the bit budget for the first scene, a quantity of bits used for already coded frames in the first scene, a complexity of the first scene, and complexities of already coded frames in the first scene.

74. The bit allocation method as defined in Claim 64, wherein all the macroblocks within the first frame are quantized using one quantization parameter value.

75. The bit allocation method as defined in Claim 64, wherein a quantization parameter is varied for the first frame macroblocks based at least in part on current bit usage and budgeted bit usage.

76. An apparatus for allocating bits, comprising:

a first instruction configured to assign a first bit budget for a first scene based at least in part on how many intracoded frames are in the first scene, a complexity determination for the first scene, and how many intercoded frames are in the first scene; and

a second instruction configured to allocate bit budgets to frames within the first scene based at least in part on a current bit usage and a target bit usage.

77. The apparatus as defined in Claim 76, further comprising varying quantization of macroblocks within the frames based at least in part of the current bit usage and the target bit usage.

78. The apparatus as defined in Claim 76, wherein the first frame in the first scene is intracoded.

79. The apparatus as defined in Claim 76, wherein the first instruction assigns a greater weight to the intracoded frames than to the intercoded frames.

80. The apparatus as defined in Claim 76, wherein the apparatus is an integrated circuit.

81. A bit budgeting apparatus, comprising :

a first instruction configured to receive a bit budget for at least a first video sequence;

a second instruction configured to determine a scene quantity for the first video sequence;

a third instruction configured to determine a quantity of predicted frames in a first scene in the first video sequence; and

WO 02/071639

PCT/US02/06726

a fourth instruction configured to determine a bit budget for the first scene based at least in part on the scene quantity for the first clip, the first clip bit budget, a quantity of intracoded frames in the first scene, and the quantity of predicted frames in the first scene.

82. The apparatus as defined in Claim 81, wherein the first scene is a GOV.

83. The apparatus as defined in Claim 81, wherein the bit budget for the first scene is based at least in part on a buffer status.

84. The apparatus as defined in Claim 81, further comprising a fifth instruction configured to determine a bit budget for a first frame based at least in part on how many texture bits are in the first frame and how many motion vector bits there are for the first frame.

85. The apparatus as defined in Claim 81, further comprising a fifth instruction configured to determine quantization parameters for macroblocks in a first frame in the first scene.

86. The apparatus as defined in Claim 81, wherein the apparatus is an integrated circuit.

87. A method of selecting which video sequence frames to skip during an encoding process, the method comprising:

receiving a sequence of frames;

for each frame in at least a portion of the sequence of frames, determining a corresponding mean of absolute differences value weighted by a temporal parameter between frames bracketing said each frame; and

at least partly based on the corresponding mean of absolute differences value weighted by a temporal parameter calculated for each frame, skipping a first frame in the sequence of frames.

88. The method as defined in Claim 87, further comprising:

receiving a target frame rate; and

determining if the target frame rate is met with the first frame skipped.

89. The method as defined in Claim 88, wherein at least partly in response to determining that the target frame rate is not met with the first frame skipped, selecting a second frame for skipping based at least in part on a second corresponding mean of

WO 02/071639

PCT/US02/06726

absolute differences value weighted by a second temporal parameter between frames bracketing the second frame.

90. The method as defined in Claim 87, further comprising limiting to a first amount how many consecutive frames can be skipped.

91. The method as defined in Claim 87, wherein a first intracoded frame is excluded from being skipped.

92. The method as defined in Claim 87, wherein a last frame in the sequence of frames is excluded from being skipped.

93. The method as defined in Claim 87, further comprising providing a header for the first skipped frame and setting a flag indicating that the first frame is not coded.

94. A method of determining which frames to skip in a video sequence, the method comprising:

- receiving a first frame rate goal;
- selecting a first potential frame to be skipped, the first potential frame between a first frame and a last frame in a video sequence scene;
- calculating a first cost value for the first potential frame to be skipped, the first cost value related to the visual impact that would result from skipping the first potential frame;
- storing the first cost value in computer readable memory;
- selecting a second potential frame to be skipped, the second potential frame between the first frame and the last frame;
- calculating a second cost value for the second potential frame to be skipped;
- storing the second cost value in computer readable memory;
- selecting a third potential frame to be skipped, the third potential frame between the first frame and the last frame;
- calculating a third cost value for the third potential frame to be skipped;
- storing the third cost value in computer readable memory;
- determining which one of the first, second, and third cost values has the least cost;
- skipping the one of the one of the first, second, and third potential frames corresponding to the least cost value;
- determining if the first frame rate goal is met; and

WO 02/071639

PCT/US02/06726

at least partly in response to determining the first frame rate goal is not met, recalculating cost values associated with frames remaining from the first, second and third potential frames, and skipping the one of the remaining frames having the least recalculated cost value.

95. The method as defined in Claim 94, wherein the first cost value is based at least in part on a mean average difference (MAD) value and a time difference between the first frame and the second potential frame to be skipped.

96. The method as defined in Claim 95, wherein only a portion of pixels in the first frame and the second potential frame to be skipped are evaluated in determining the MAD value.

97. The method as defined in Claim 95, wherein the time difference is normalized to an NTSC frame rate.

98. The method as defined in Claim 94, wherein the number of frames that are allowed to be consecutively skipped is limited to a first number.

99. The method as defined in Claim 94, further comprising providing a header for each skipped frame.

100. The method as defined in Claim 94, further comprising transmitting a Header Extension Code (HEC), including sequence information, in every packet in a sequence of video packets.

101. A method of selecting image frames to be skipped during an encoding process, the method comprising:

receiving a sequence of frames including a first frame;

calculating a first cost function value for the first frame, wherein the cost function includes a first temporal parameter related to a time difference and luminosity differences between frames bounding the first frame;

calculating a second cost function value for a second frame, wherein the cost function includes a second temporal parameter related to a time difference between frames and luminosity differences bounding the second frame; and

selecting for skipping one of the first frame and the second frame based at least in part on the first cost function value and the second cost function value.

102. The method as defined in Claim 101, further comprising:

receiving a target frame rate; and

WO 02/071639

PCT/US02/06726

determining if the target frame rate is met with one of the first frame and the second frame being skipped.

103. The method as defined in Claim 101, wherein the luminosity differences between frames bounding the first frame are defined using a mean of average difference function.

104. The method as defined in Claim 101, further comprising limiting to a first amount how many consecutive frames can be skipped.

105. The method as defined in Claim 101, wherein the first temporal parameter is normalized to a first frame rate.

106. A processor circuit configured to select video frames to skip in a sequence of frames, the processor circuit comprising:

a first instruction configured to calculate for each frame in at least a portion of the sequence of frames a corresponding mean of absolute differences value weighted by a temporal parameter between frames bracketing said each frame in the portion of the sequence of frames; and

a second instruction configured to designate a first frame to be skipped based at least in part on the corresponding mean of absolute differences value weighted by the temporal parameter.

107. The processor circuit as defined in Claim 106, further comprising a third instruction configured to determine if a target frame rate is met assuming the first frame is skipped.

108. The processor circuit as defined in Claim 106, further comprising a third instruction configured to limit to a first amount how many consecutive frames can be skipped.

109. A frame rate control apparatus, comprising:

a first instruction configured to receive a frame rate target;

a second instruction configured to calculate a first cost function value for a first frame in a sequence of frames, wherein the first cost function value is based at least in part on a first temporal parameter related to a time difference and luminosity differences between frames in the sequence of frames bounding the first frame;

a third instruction configured to calculate a second cost function value for a second frame in the sequence of frames, wherein the second cost function based at

WO 02/071639

PCT/US02/06726

least in part on a second temporal parameter related to a time difference and luminosity differences between frames in the sequence of frames bounding the second frame;

a fourth instruction configured to select for skipping one of the first frame and the second frame based at least in part on the first cost function value and the second cost function value; and

a fifth instruction configured to determine if the target frame rate is met with the selected one of the first frame and the second frame skipped.

110. The frame rate control apparatus as defined in Claim 109, wherein the frame rate control apparatus is an integrated circuit.

111. The frame rate control apparatus as defined in Claim 109, wherein the luminosity differences between frames bounding the first frame are defined using a mean of average difference function.

112. A method of providing forward error correction (FEC) on a plurality of frame packets, the method comprising:

concatenating selected portions of packet data corresponding to a plurality of frame packets for a first frame;

generating forward error correction bits for the concatenated selected portions of packet data; and

transmitting the forward error correction bits in a separate packet identified with a user data identifier code.

113. The method as defined in Claim 112, wherein the transmission of the forward error correction bits in the separate packet is MPEG-4 compliant.

114. The method as defined in Claim 112, wherein the separate packet is transmitted after the plurality of frame packets.

115. The method as defined in Claim 112, wherein the forward error correction bits are generated using a BCH code.

116. The method as defined in Claim 112, wherein the forward error correction bits are generated using a systematic code.

117. The method as defined in Claim 112, wherein the selected portions of packet data includes motion vector data and DCT data.

WO 02/071639

PCT/US02/06726

118. The method as defined in Claim 112, wherein the selected portions of packet data includes only header data, motion vector data and DCT data.
119. The method as defined in Claim 112, wherein the selected portions of packet data corresponds to packet data located between a resync field and a motion marker.
120. The method as defined in Claim 112, further comprising:  
setting a flag indicating that a fixed Video Object Plane (VOP) increment is to be used; and  
providing a corresponding fixed time increment value.
121. The method as defined in Claim 112, further comprising transmitting in the separate packet a value for at least a first of the plurality of frame packets related to a quantity of bits within the first packet for which forward error correction bits were generated.
122. An error correction generation circuit, comprising:  
a first instruction stored in processor readable memory configured to generate forward error correction data for selected portions of packet data that are to be transmitted in a corresponding plurality of frame packets;  
a second instruction stored in processor readable memory configured to store the forward error correction data in a first packet separate from the plurality of frame packets; and  
a third instruction stored in processor readable memory configured to identify the first packet with a first data identifier code.
123. The error correction generation circuit as defined in Claim 122, further comprising a fourth instruction configured to concatenate selected portions of packet data before the first instruction generates the forward error correction data.
124. The error correction generation circuit as defined in Claim 122, further comprising a fourth instruction configured to set a flag indicating that a fixed Video Object Plane (VOP) increment is to be used and to provide a corresponding fixed time increment value.
125. The error correction generation circuit as defined in Claim 122, further comprising a fourth instruction configured to provide a Header Extension Code (HEC) in a every packet in a first sequence of packets.

WO 02/071639

PCT/US02/06726

126. The error correction generation circuit as defined in Claim 122, wherein the error correction generation circuit is an integrated circuit.
127. The error correction generation circuit as defined in Claim 122, wherein the first packet is MPEG-4 compliant.
128. The error correction generation circuit as defined in Claim 122, wherein the forward error correction data is generated using a BCH code.
129. The error correction generation circuit as defined in Claim 122, wherein the forward error correction data is generated using a systematic code.
130. The error correction generation circuit as defined in Claim 122, wherein the selected portions of packet data includes motion vector data and DCT data.
131. The error correction generation circuit as defined in Claim 122, wherein the selected portions of packet data includes only header data, motion vector data and DCT data.
132. The error correction generation circuit as defined in Claim 122, wherein the selected portions of packet data corresponds to packet data located between a resync field and a motion marker.
133. An encoder circuit, comprising:  
a means for generating forward error correction data for selected portions of packet data from a plurality of frame packets;  
a means for storing the forward error correction data in a first packet separate from the plurality of frame packets; and  
a means for identifying the first packet with a first data identifier code.
134. The encoder as defined in Claim 133, further comprising a means for concatenating the selected portions of packet data.
135. The encoder as defined in Claim 133, further comprising a means for transmitting in the first packet at least a first value related to a quantity of bits within the first packet for which forward error correction bits were generated.
136. A method of performing adaptive intra refresh, the method comprising:  
receiving a packet loss probability value;  
receiving a motion vector for a first macroblock in a first frame;  
mapping the motion vector onto portions of a plurality of other macroblocks in a previous frame;



WO 02/071639

PCT/US02/06726

calculating at least a first transition factor value based at least in part on the mapping;

calculating an estimated inter distortion value for the first macroblock based on at least the packet loss probability value and the first transition factor value;

calculating an estimated intra distortion value for the first macroblock based on at least the packet loss probability value;

receiving a first bit quantity value corresponding to a bit quantity used to intracode the first macroblock;

receiving a second bit quantity value corresponding to a bit quantity used to intercode the first macroblock; and

based at least in part on the estimated inter distortion value, the estimated intra distortion value, the first bit quantity value, and the second bit quantity value, transmitting the first macroblock as one of an intra macroblock and an inter macroblock.

137. The method as defined in Claim 136, further comprising transmitting the first macroblock as one of an intra macroblock and an inter macroblock based in part on a quantization distortion value.

138. The method as defined in Claim 136, further comprising determining a first difference value based on a difference between the estimated intra distortion value and the estimated inter distortion value, and a second difference value based on a difference between the first bit quantity value and the second bit quantity value.

139. The method as defined in Claim 137, wherein the first macroblock is transmitted as an intra macroblock at least partly in response to the second difference value being positive, and the first difference value divided by the second difference value being more negative than a first threshold.

140. The method as defined in Claim 137, wherein the first macroblock is transmitted as an intra macroblock at least partly in response to the second difference value being zero, and the first difference value being negative.

141. The method as defined in Claim 137, wherein the first macroblock is transmitted as an intra macroblock at least partly in response to the second difference value being negative, and the first difference value divided by the second difference value being greater than a first threshold.

WO 02/071639

PCT/US02/06726

142. The method as defined in Claim 141, wherein the first threshold is a weighting factor.

143. The method as defined in Claim 137, wherein the first macroblock is transmitted as an inter macroblock at least partly in response to the second difference value being zero, and the first difference value being positive.

144. The method as defined in Claim 137, wherein the first macroblock is transmitted as an intra macroblock at least partly in response to the second difference value being negative, and the first difference value divided by the second difference value being less than a first threshold.

145. The method as defined in Claim 137, wherein the first macroblock is transmitted as an intra macroblock at least partly in response to the second difference value being positive, and the first difference value divided by the second difference value being less than a first threshold.

146. The method as defined in Claim 136, wherein the estimated intra distortion value is further based on an initial error energy.

147. The method as defined in Claim 136, wherein the estimated intra distortion value is further based on an initial error energy squared multiplied by the packet loss probability value.

148. The method as defined in Claim 136, wherein the estimated inter distortion value is recursively calculated to include distortion propagated from a plurality of previous frames.

149. The method as defined in Claim 136, wherein the first transition factor corresponds to a first half-pixel horizontal and vertical propagation strength.

150. The method as defined in Claim 136, wherein the first transition factor corresponds to a first half-pixel horizontal strength.

151. The method as defined in Claim 136, wherein the first transition factor corresponds to a first half-pixel vertical propagation strength.

152. The method as defined in Claim 136, further comprising limiting how many macroblocks can be intracoded in the first frame to a first amount.

153. A method of selectively intracoding macroblocks, the method comprising:  
receiving a packet loss probability value;  
receiving a motion vector for a first macroblock in a first frame;

WO 02/071639

PCT/US02/06726

based at least in part on the motion vector, determining which portions of macroblocks in a previous frame would be used in predicting the first macroblock;

calculating at least a first propagation strength value based at least in part on determining which portions of macroblocks in the previous frame would be used in predicting the first macroblock;

calculating an estimated inter distortion value for the first macroblock based on at least the packet loss probability value and the first propagation strength value;

calculating an estimated intra distortion value for the first macroblock based on at least the packet loss probability value;

calculating a quantization distortion value for the first macroblock;

receiving a first bit quantity value corresponding to a bit quantity used to intracode the first macroblock;

receiving a second bit quantity value corresponding to a bit quantity used to intercode the first macroblock; and

based at least in part on the estimated inter distortion value, the estimated intra distortion value, the quantization distortion value, the first bit quantity value, and the second bit quantity value, providing for transmission the first macroblock as one of an intra macroblock and an inter macroblock.

154. The method as defined in Claim 153, wherein the intra distortion value is further based on an initial error energy.

155. The method as defined in Claim 153, wherein the estimated inter distortion value is recursively calculated to include distortion propagated from a plurality of previous frames.

156. The method as defined in Claim 153, wherein the intra distortion value is further based on an initial error energy squared multiplied by the packet loss probability value.

157. The method as defined in Claim 153, wherein the first macroblock is provided as one of an intra macroblock and an inter macroblock further based upon a weighting factor.

158. The method as defined in Claim 157, wherein the weighting factor is generated at least in part by calculating a plurality of intra distortions values using

WO 02/071639

PCT/US02/06726

corresponding different quantization parameters, and selecting a weighting factor value that results in the lowest distortion value meeting a first bitrate criterion.

159. A method of selectively intracoding macroblocks in a plurality of macroblocks in a first frame, the method comprising:

receiving a packet loss probability value;

receiving a corresponding motion vector for each macroblock in the plurality of macroblocks;

based at least in part on the corresponding motion vector, determining which portions of macroblocks in a previous frame would be used in predicting said each macroblock;

calculating for each of said macroblocks at least a first corresponding propagation strength value based at least in part on determining which portions of macroblocks in the previous frame would be used in predicting said each macroblock;

calculating for each of said macroblocks an estimated inter distortion value based upon at least the packet loss probability value and the corresponding at least first propagation strength value;

calculating for each of said macroblocks an estimated intra distortion value based upon at least the packet loss probability value;

calculating for each of said macroblocks a quantization distortion value; and

based at least in part on the estimated inter distortion, the estimated intra distortion, and the quantization distortion values for each macroblock in the plurality of macroblocks, designating a subset of the plurality of macroblocks to be intracoded.

160. The method as defined in Claim 159, wherein the subset is limited to a predetermined number of macroblocks.

161. The method as defined in Claim 159, further comprising:

for each of said macroblocks, receiving a first bit quantity value corresponding to a bit quantity used to intracode said each macroblock; and

for each of said macroblocks, receiving a second bit quantity value corresponding to a bit quantity used to intercode said each macroblock, wherein the

WO 02/071639

PCT/US02/06726

subset is designated based in part on the first bit quantity values and the second bit quantity values.

162. A circuit configured to selectively intracode macroblocks, the circuit comprising:

- a first instruction configured to receive a packet loss probability value;
- a second instruction configured to receive a motion vector for a first macroblock in a first frame;
- a third instruction configured to determine, based at least in part on the motion vector, which portions of macroblocks in a previous frame would be used in predicting the first macroblock ;
- a fourth instruction configured to calculate at least a first propagation strength value based at least in part on the determination of which portions of macroblocks in the previous frame would be used in predicting the first macroblock;
- a fifth instruction configured to calculate an estimated inter distortion value for the first macroblock based on at least the packet loss probability value and the first propagation strength value;
- a sixth instruction configured to calculate an estimated intra distortion value for the first macroblock based on at least the packet loss probability value; and
- a seventh instruction configured to selectively provide for transmission the first macroblock as one of an intra macroblock and an inter macroblock based at least in part on the estimated inter distortion value and the estimated intra distortion value.

163. The circuit as defined in Claim 162, wherein the intra distortion value is further based upon an initial error energy.

164. The circuit as defined in Claim 162, further comprising an eighth instruction configured to calculate a quantization distortion value for the first macroblock, wherein the seventh instruction is further configured to selectively provide for transmission the first macroblock as one of an intra macroblock and an inter macroblock based at least in part on the quantization distortion value.

165. The circuit as defined in Claim 162, wherein the fifth instruction is further configured to recursively calculate the estimated inter distortion value to thereby include distortion propagated from a plurality of previous frames.

WO 02/071639

PCT/US02/06726

166. The circuit as defined in Claim 162, wherein the intra distortion value is further based on an initial error energy squared multiplied by the packet loss probability value.

167. The circuit as defined in Claim 162, wherein the seventh instruction is further configured to selectively provide for transmission the first macroblock as one of an intra macroblock and an inter macroblock based at least in part on a weighting factor.

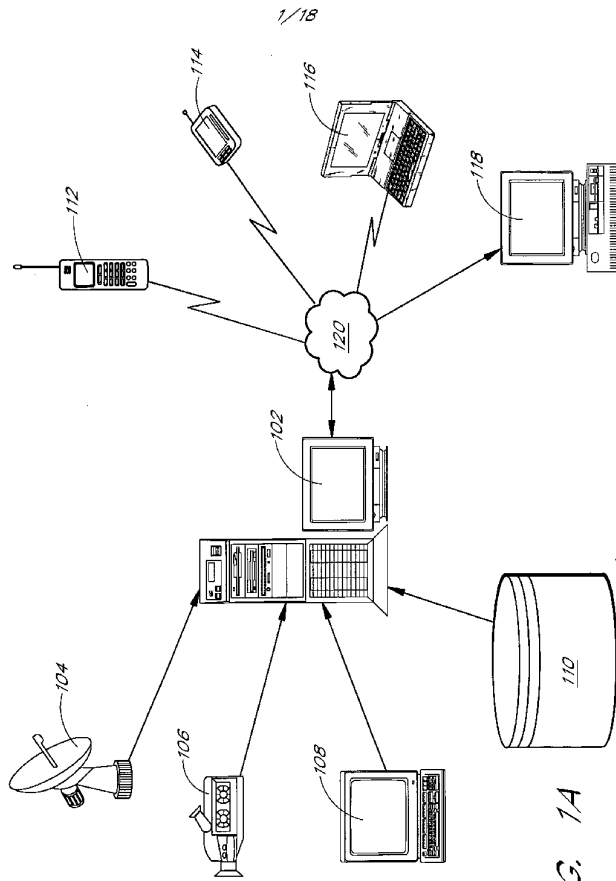


FIG. 1A

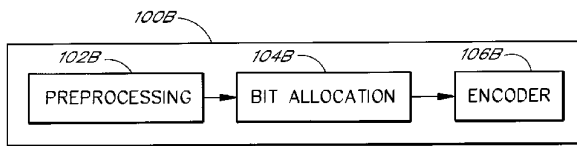


FIG. 1B



3/18

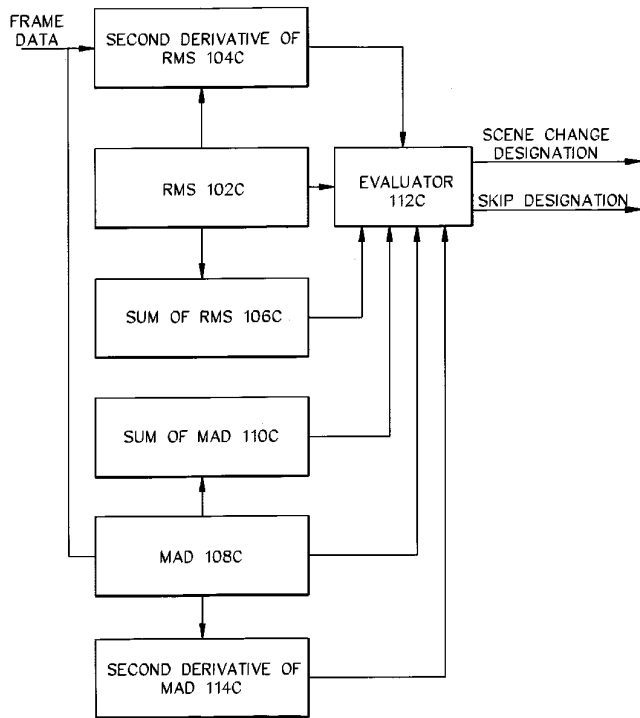


FIG. 1C

4/18

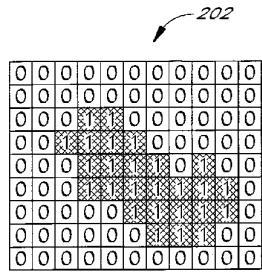


FIG. 2A  
(PRIOR ART)

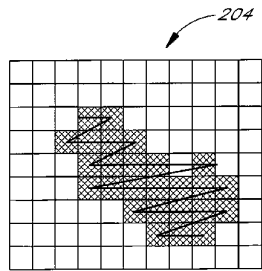
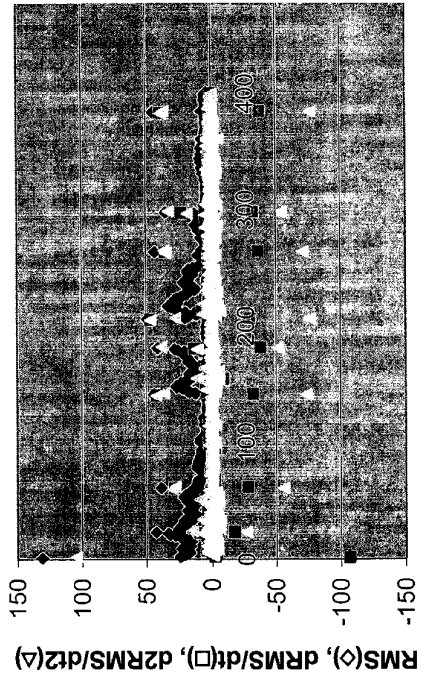


FIG. 2B  
(PRIOR ART)



Frame number

FIG. 3

6/18

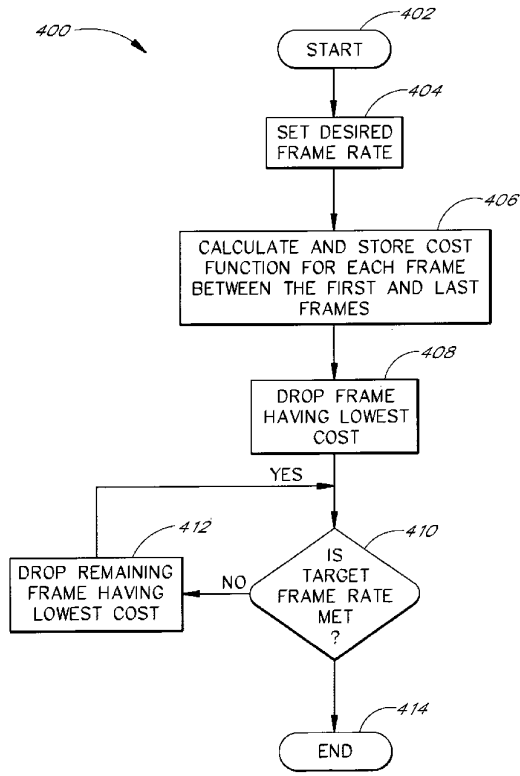


FIG. 4A

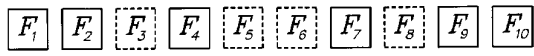


FIG. 4B

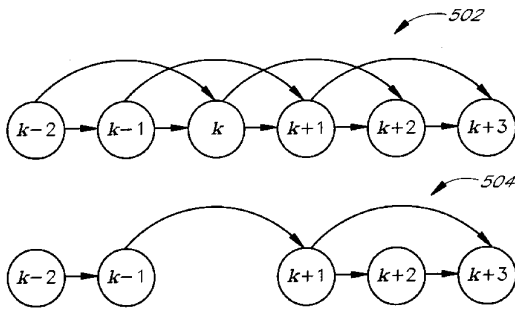


FIG. 5

WO 02/071639

PCT/US02/06726

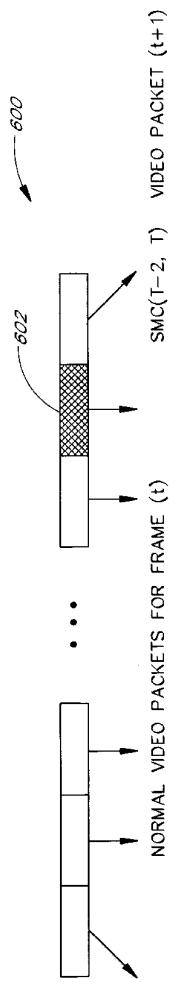


FIG. 6

8/18

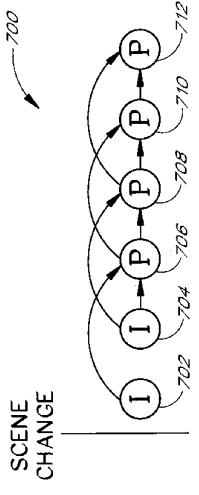
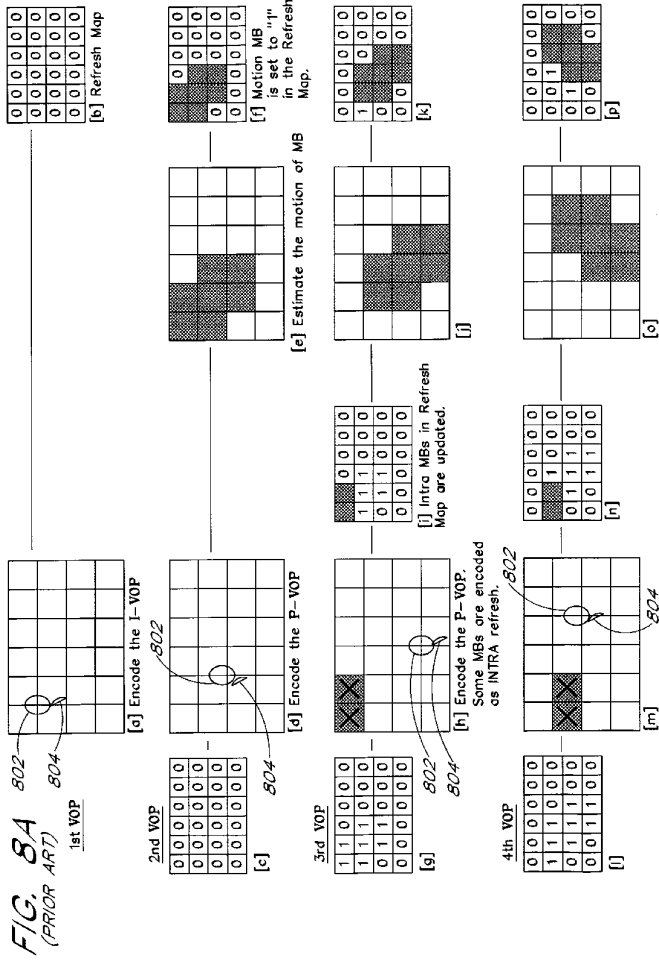


FIG. 7

WO 02/071639

PCT/US02/06726

9/18



10/18

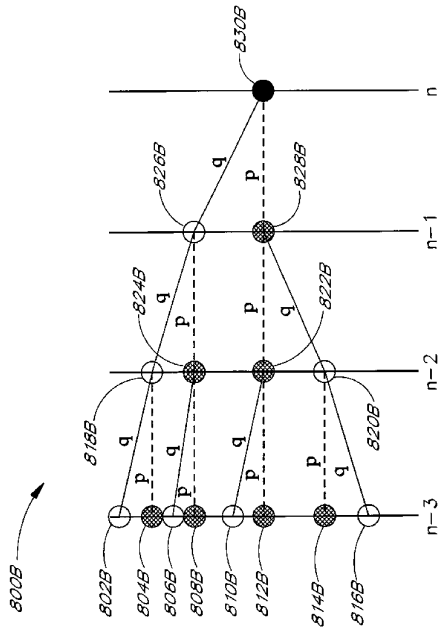


FIG. 8B



11/18

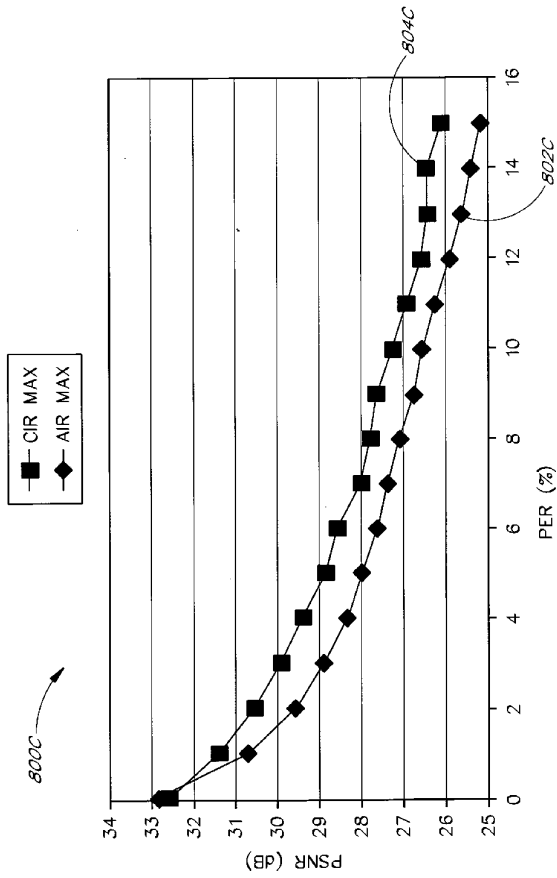


FIG. 8C

12/18

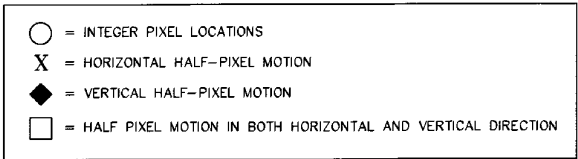
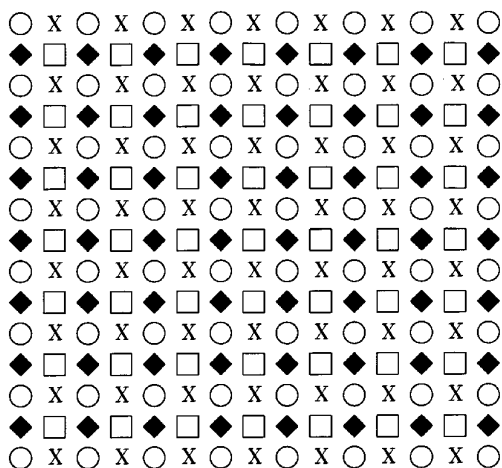


FIG. 8D

13/18

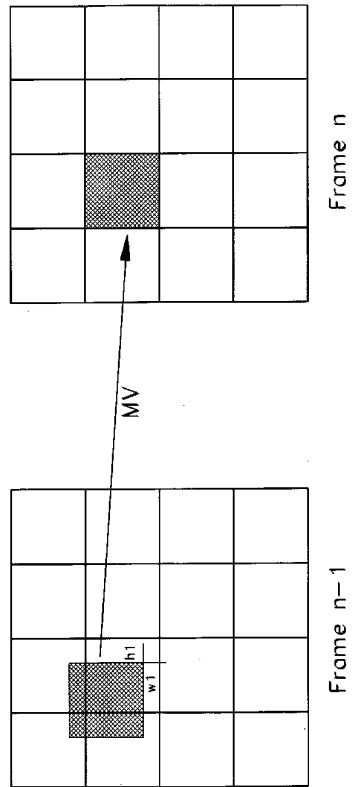


FIG. 8E

14/18

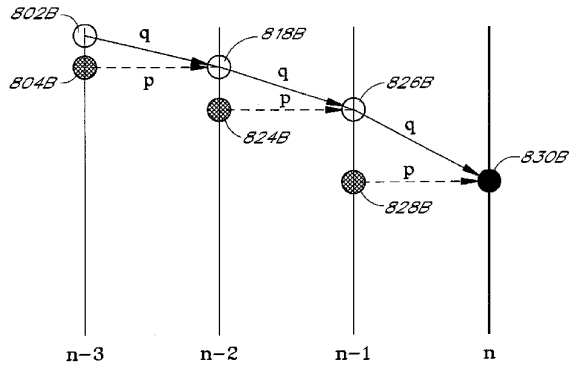


FIG. 8F

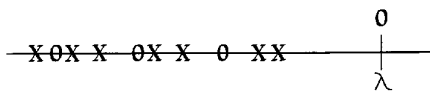
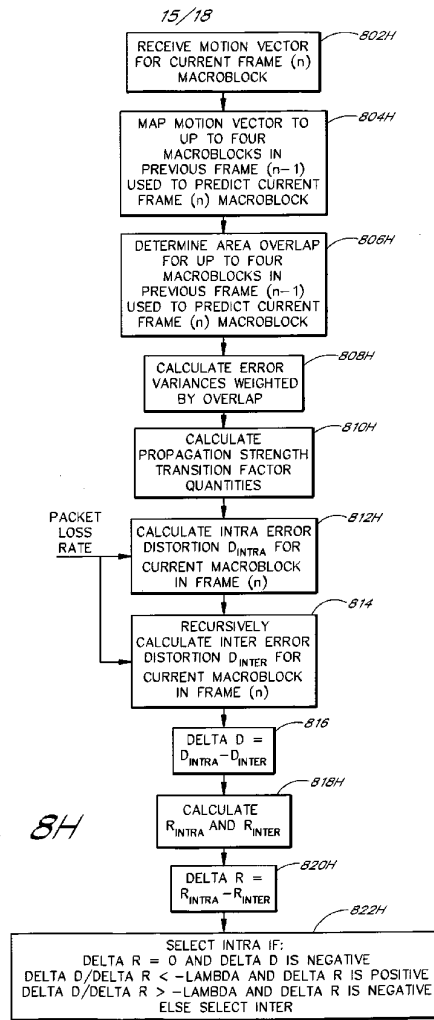


FIG. 8G



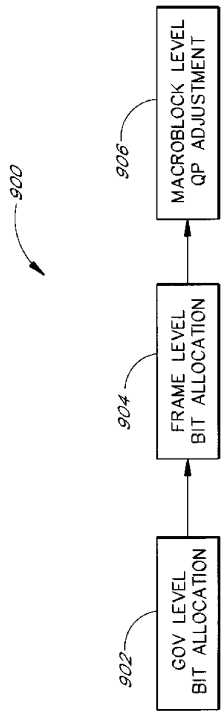


FIG. 9

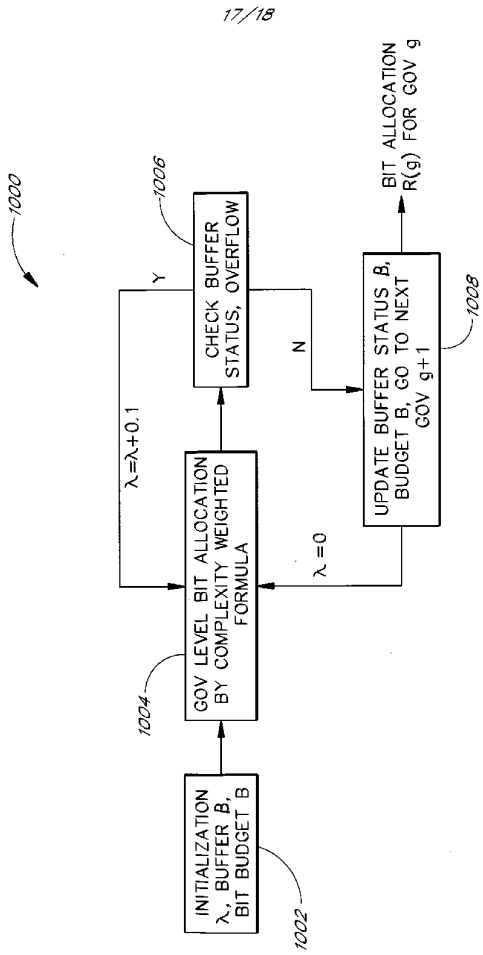


FIG. 10

18/18

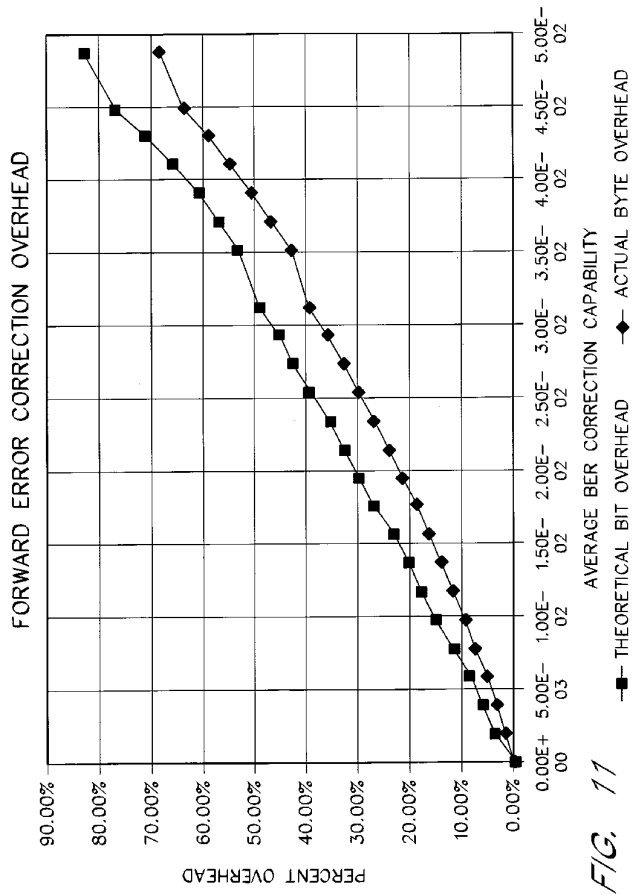


FIG. 11



【国際公開パンフレット(コレクトバージョン)】

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

CORRECTED VERSION

(19) World Intellectual Property Organization  
International Bureau



(43) International Publication Date  
12 September 2002 (12.09.2002)

PCT

(10) International Publication Number  
WO 02/071639 A1

(51) International Patent Classification: H04B 1/66

(21) International Application Number: PCT/US02/06726

(22) International Filing Date: 5 March 2002 (05.03.2002)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:  
60/273,443 5 March 2001 (05.03.2001) US  
60/275,859 14 March 2001 (14.03.2001) US  
60/286,280 25 April 2001 (25.04.2001) US

CIL, CN, CO, CR, CU, CZ (utility model), CZ, DE (utility model), DE, DK (utility model), DK, DM, DZ, EC, EE (utility model), EE, ES, FI (utility model), FI, GB, GD, GL, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PI, PL, PT, RO, RU, SD, SE, SG, SI, SK (utility model), SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZM, ZW.

(84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SI, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NI, SN, TD, TG).

(71) Applicant (for all designated States except US): INTER-VIDEO, INC. [US/US], 47350 Fremont Boulevard, Fremont, CA 94538 (US).

Declaration under Rule 4.17:  
of inventorship (Rule 4.17(vii)) for US only

Published:  
with international search report

(72) Inventors; and  
(75) Inventors/Applicants (for US only): KATSAVOUNIDIS, Ioannis [GR/US]; 222 S. El Molino Avenue, Apartment #7, Pasadena, CA 91101 (US); KUO, Chung-Chieh [US/US]; 5634 Halifax Road, Arcadia, CA 91007 (US).

(48) Date of publication of this corrected version:  
30 October 2003

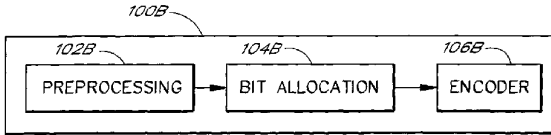
(74) Agent: ALTMAN, Daniel, E.; Knobbe, Martens, Olson and Bear, LLP, 620 Newport Center Drive, 16th Floor, Newport Beach, CA 92660 (US).

(15) Information about Correction:  
see PCT Gazette No. 44/2003 of 30 October 2003, Section II

(81) Designated States (national): AE, AG, AL, AM, AT (utility model), AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA,

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: SYSTEMS AND METHODS FOR ERROR RESILIENT ENCODING



(57) Abstract: The present invention is related to error resilient video encoding. Embodiments of the present invention provide systems (Figure 1B, 100B) and methods for reducing error propagation in a video data stream, for refreshing macroblocks, for reducing frame rates in a video data stream, for generating error correction information for a media stream, for performing bit rate allocation for a video data stream, and for detecting scene changes in a video data stream.

WO 02/071639 A1

【 国際調査報告 】

INTERNATIONAL SEARCH REPORT		International application No. PCT/US02/06726												
<b>A. CLASSIFICATION OF SUBJECT MATTER</b>														
IPC(7) : H04B 1/66 US CL. : 375/240 According to International Patent Classification (IPC) or to both national classification and IPC														
<b>B. FIELDS SEARCHED</b>														
Minimum documentation searched (classification system followed by classification symbols) U.S. : 375/240; 382/296														
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched														
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)														
<b>C. DOCUMENTS CONSIDERED TO BE RELEVANT</b>														
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.												
X	US 6,141,448 A (KHANSARI et al) 31 October 2000, col.6, line 36 to col.7, line 64	1-167												
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input type="checkbox"/> See patent family annex.														
<table border="0"> <tr> <td>* Special categories of cited documents:</td> <td>"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention.</td> </tr> <tr> <td>"A" document defining the general state of the art which is not considered to be of particular relevance</td> <td>"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone</td> </tr> <tr> <td>"E" earlier document published on or after the international filing date</td> <td>"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art</td> </tr> <tr> <td>"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)</td> <td>"g" document member of the same patent family</td> </tr> <tr> <td>"O" document referring to an oral disclosure, use, exhibition or other means</td> <td></td> </tr> <tr> <td>"P" document published prior to the international filing date but later than the priority date claimed</td> <td></td> </tr> </table>			* Special categories of cited documents:	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention.	"A" document defining the general state of the art which is not considered to be of particular relevance	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone	"E" earlier document published on or after the international filing date	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art	"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"g" document member of the same patent family	"O" document referring to an oral disclosure, use, exhibition or other means		"P" document published prior to the international filing date but later than the priority date claimed	
* Special categories of cited documents:	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention.													
"A" document defining the general state of the art which is not considered to be of particular relevance	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone													
"E" earlier document published on or after the international filing date	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art													
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"g" document member of the same patent family													
"O" document referring to an oral disclosure, use, exhibition or other means														
"P" document published prior to the international filing date but later than the priority date claimed														
Date of the actual completion of the international search 22 MAY 2002	Date of mailing of the international search report 05 JUN 2002													
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer CHRIS KELLEY <i>Chris Kelley</i> Telephone No. (703) 305-4700													

## フロントページの続き

(81) 指定国 AP(GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), EA(AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), EP(AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZM, ZW

(特許庁注：以下のものは登録商標)

レーザーディスク

Linux

UNIX

Windows NT

(72) 発明者 カツァヴーニディス イオアニス

アメリカ合衆国 9 4 5 3 8 カリフォルニア州 フレモント フレモント ブルバード 4 7  
3 5 0 インタービデオインコーポレイテッド内

(72) 発明者 クウ チャン - チェー

アメリカ合衆国 9 4 5 3 8 カリフォルニア州 フレモント フレモント ブルバード 4 7  
3 5 0 インタービデオインコーポレイテッド内

Fターム(参考) 5C059 KK50 MA00 MA05 MA14 MA23 MC11 MC32 MC34 MC38 NN15  
NN21 PP05 PP06 PP07 PP16 RB02 RB09 RF02 RF05 SS10  
SS26 TA07 TA60 TA76 TC02 TC14 TC20 TC22 TD02 TD03  
TD05 TD12 UA02 UA31