

(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2011-514076
(P2011-514076A)

(43) 公表日 平成23年4月28日(2011.4.28)

(51) Int.Cl.

HO4N 7/26 (2006.01)

F 1

HO4N 7/13

A

テーマコード(参考)

5C159

審査請求 有 予備審査請求 有 (全 61 頁)

(21) 出願番号	特願2010-548816 (P2010-548816)
(86) (22) 出願日	平成21年2月24日 (2009.2.24)
(85) 翻訳文提出日	平成22年10月26日 (2010.10.26)
(86) 國際出願番号	PCT/US2009/034948
(87) 國際公開番号	W02009/108614
(87) 國際公開日	平成21年9月3日 (2009.9.3)
(31) 優先権主張番号	61/031,438
(32) 優先日	平成20年2月26日 (2008.2.26)
(33) 優先権主張国	米国(US)
(31) 優先権主張番号	12/170,991
(32) 優先日	平成20年7月10日 (2008.7.10)
(33) 優先権主張国	米国(US)

(71) 出願人	595020643 クアアルコム・インコーポレイテッド QUALCOMM INCORPORATED アメリカ合衆国、カリフォルニア州 92121-1714、サン・ディエゴ、モアハウス・ドライブ 5775
(74) 代理人	100108855 弁理士 蔵田 昌俊
(74) 代理人	100091351 弁理士 河野 哲
(74) 代理人	100088683 弁理士 中村 誠
(74) 代理人	100109830 弁理士 福原 淑弘

最終頁に続く

(54) 【発明の名称】ビデオデコーダエラー処理

(57) 【要約】

ビデオデコーダが、ビデオデータユニットの破損データセグメント内のエラーを検出し、隠蔽するために、連続エラー処理プロセスを実行する。デコーダはカレントデータユニットを連続的に復号する。エラーを検出すると、デコーダはエラーフラグを設定し、次のユニットの開始時に復号を再同期させる。エラーフラグが設定されている場合、ビデオデコーダは、その後のユニットの開始に基づいて破損データセグメントの終了を識別する。デコーダは、カレントユニットの開始と破損データセグメントの終了との間のデータを隠蔽する。エラーフラグが設定されていない場合、デコーダは、カレントユニットの残りを復号し、カレントユニットに対してエラー処理及び隠蔽を実行することなしに次の利用可能なユニットを復号することに進む。デコーダはまた、損失したビデオデータユニットによって引き起こされた参照ユニット不整合に対処することができる。

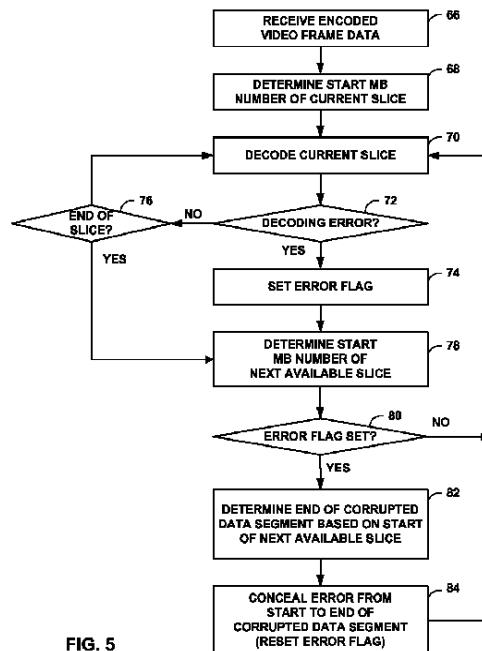


FIG. 5

【特許請求の範囲】**【請求項 1】**

符号化ビデオデータのカレントユニットの開始を決定することと、
前記カレントユニットの少なくとも一部分を復号することと、
前記カレントユニット中の復号エラーを検出することと、
前記符号化ビデオデータの次の利用可能なユニットの開始を決定することと、
前記復号エラーが検出された場合、前記次の利用可能なユニットの前記開始に基づいて
破損データセグメントの終了を決定することと、
前記カレントユニットの前記開始と前記破損データセグメントの前記終了とに基づいて
前記破損データセグメントを隠蔽することと、
を含む、ビデオ復号方法。
10

【請求項 2】

前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータスライス
を含む、請求項 1 に記載の方法。

【請求項 3】

前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの少なくとも
1 つの追加のユニットとを含む、請求項 1 に記載の方法。

【請求項 4】

前記カレントユニットが第 1 のフレームのカレントスライスであり、前記破損データセ
グメントが、前記カレントスライスと第 2 のフレームからの符号化ビデオデータの少なくとも
1 つの追加のスライスとを含む、請求項 1 に記載の方法。
20

【請求項 5】

前記復号エラーが検出された場合、前記復号エラーの前記検出を示すためにエラーフラ
グを設定することと、
前記エラーフラグが設定されているとき、前記次の利用可能なユニットの前記開始に基
づいて前記破損データセグメントの前記終了を決定することと
を更に備える、請求項 1 に記載の方法。

【請求項 6】

前記次の利用可能なユニットが少なくとも 1 つのスライスを備え、前記方法は、前記復
号エラーが検出されたとき、前記スライスに関連するスライスヘッダを探すことと更に備
える、請求項 5 に記載の方法。
30

【請求項 7】

前記復号エラーが検出された場合、前記カレントユニットの復号を止めることと、前記
次の利用可能なユニットを復号することと進むことを更に含む、請求項 5 に記載の方法
。

【請求項 8】

前記復号エラーが検出されなかった場合、第 1 のユニット全体を復号することと更に含
む、請求項 1 に記載の方法。

【請求項 9】

前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエントロピー符
号化工エラーの少なくとも 1 つを含む、請求項 1 に記載の方法。
40

【請求項 10】

前記カレントユニットの開始を決定することが、前記カレントユニットの前記開始に關
連する第 1 のマクロブロック (MB) 番号を決定することを含み、前記破損データセグメ
ントの終了を決定することが、前記第 2 のユニットの開始より前に第 2 の MB 番号を決定
することを含み、隠蔽することが、前記第 1 の MB 番号から前記第 2 の MB 番号までの前
記破損データセグメントを隠蔽することを含む、請求項 1 に記載の方法。

【請求項 11】

前記第 2 のユニットの開始を決定することが、前記第 2 のユニットの前記開始に關連す
る第 3 の MB 番号を決定することを備え、前記第 2 の MB 番号を決定することが、前記第
50

2 の M B 番号を生成するために、前記第 3 の M B 番号から 1 を減算することを備える、請求項 10 に記載の方法。

【請求項 12】

前記第 1 のユニットの開始を決定することが、前記第 1 のユニットの前記開始に関連する第 1 のマクロブロック (M B) 番号と第 1 のフレーム番号とを決定することを備え、前記破損データセグメントの終了を決定することが、前記破損データセグメントの前記終了に関連する第 2 の M B 番号と第 2 のフレーム番号とを決定することを備え、隠蔽することが、前記第 1 の M B 番号及び前記第 2 の M B 番号と前記第 1 のフレーム番号及び前記第 2 のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽することを備え、前記第 1 のフレーム番号と前記第 2 のフレーム番号とが、同じことも、異なることもあることがある、請求項 10 に記載の方法。

10

【請求項 13】

1 つ以上のフレームの損失を検出することと、
別のフレームの参照フレーム不整合を隠蔽するために前記 1 つ以上の損失したフレームを交換することと
を更に含む、請求項 1 に記載の方法。

【請求項 14】

符号化ビデオデータのカレントユニットの開始を決定するための手段と、
前記カレントユニットの少なくとも一部分を復号するための手段と、
前記カレントユニット中の復号エラーを検出するための手段と、
前記符号化ビデオデータの次の利用可能なユニットの開始を決定するための手段と、
前記復号エラーが検出された場合、前記次の利用可能なユニットの前記開始に基づいて破損データセグメントの終了を決定するための手段と、
前記カレントユニットの前記開始と前記破損データセグメントの前記終了とに基づいて前記破損データセグメントを隠蔽するための手段と、
を備える、ビデオ復号装置。

20

【請求項 15】

前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータスライスを含む、請求項 14 に記載の装置。

【請求項 16】

前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの少なくとも 1 つの追加のユニットとを含む、請求項 14 に記載の装置。

30

【請求項 17】

前記カレントユニットが、第 1 のフレームのカレントスライスであり、前記破損データセグメントが、前記カレントスライスと第 2 のフレームからの符号化ビデオデータの少なくとも 1 つの追加のスライスとを含む、請求項 14 に記載の装置。

【請求項 18】

前記復号エラーが検出された場合、前記復号エラーの前記検出を示すためにエラーフラグを設定するための手段と、

前記エラーフラグが設定されているとき、前記次の利用可能なユニットの前記開始に基づいて前記破損データセグメントの前記終了を決定するための手段と
を更に備える、請求項 14 に記載の装置。

40

【請求項 19】

前記次の利用可能なユニットが少なくとも 1 つのスライスを備え、前記装置は、前記復号エラーが検出されたとき、前記スライスに関するスライスヘッダを探すための手段を更に備える、請求項 18 に記載の装置。

【請求項 20】

前記復号エラーが検出された場合、前記カレントユニットの復号を止めるための手段と、前記次の利用可能なユニットを復号することに進むための手段とを更に備える、請求項 18 に記載の装置。

50

【請求項 2 1】

前記復号エラーが検出されなかった場合、第1のユニット全体を復号するための手段を更に備える、請求項14に記載の装置。

【請求項 2 2】

前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエントロピー符号化工エラーの少なくとも1つを備える、請求項14に記載の装置。

【請求項 2 3】

前記カレントユニットの開始を決定するための前記手段が、前記カレントユニットの前記開始に関連する第1のマクロブロック(MB)番号を決定するための手段を備え、前記破損データセグメントの終了を決定するための前記手段が、前記第2のユニットの開始より前に第2のMB番号を決定するための手段を備え、隠蔽するための前記手段が、前記第1のMB番号から前記第2のMB番号までの前記破損データセグメントを隠蔽するための手段を備える、請求項14に記載の装置。10

【請求項 2 4】

前記第2のユニットの開始を決定するための前記手段が、前記第2のユニットの前記開始に関連する第3のMB番号を決定するための手段を備え、前記第2のMB番号を決定するための前記手段が、前記第2のMB番号を生成するために、前記第3のMB番号から1を減算するための手段を備える、請求項23に記載の装置。

【請求項 2 5】

前記第1のユニットの開始を決定するための前記手段が、前記第1のユニットの前記開始に関連する第1のマクロブロック(MB)番号と第1のフレーム番号とを決定するための手段を備え、前記破損データセグメントの終了を決定するための前記手段が、前記破損データセグメントの前記終了に関連する第2のMB番号と第2のフレーム番号とを決定するための手段を備え、隠蔽するための前記手段が、前記第1のMB番号及び前記第2のMB番号と前記第1のフレーム番号及び前記第2のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽するための手段を備え、前記第1のフレーム番号と前記第2のフレーム番号とが、同じことも、異なることもある、請求項23に記載の装置。20

【請求項 2 6】

1つ以上のフレームの損失を検出するための手段と、30

別のフレームの参照フレーム不整合を隠蔽するために前記1つ以上の損失したフレームを交換するための手段と

を更に備える、請求項14に記載の装置。

【請求項 2 7】

符号化ビデオデータのカレントユニットの少なくとも一部分を復号する復号エンジンと40

、カレントユニット中の復号エラーを検出するエラー検出モジュールと、

前記カレントユニットの開始を決定し、前記符号化ビデオデータの次の利用可能なユニットの開始を決定し、前記復号エラーが検出された場合、前記次の利用可能なユニットの前記開始に基づいて破損データセグメントの終了を決定するエラーマッピングモジュールと、

前記カレントユニットの前記開始と前記破損データセグメントの前記終了とに基づいて前記破損データセグメントを隠蔽するエラー隠蔽モジュールと
を備える、ビデオ復号装置。

【請求項 2 8】

前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータスライスを含む、請求項27に記載の装置。

【請求項 2 9】

前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの少なくとも1つの追加のユニットとを含む、請求項27に記載の装置。50

【請求項 3 0】

前記カレントユニットが、第1のフレームのカレントスライスであり、前記破損データセグメントが、前記カレントスライスと第2のフレームからの符号化ビデオデータの少なくとも1つの追加のスライスとを含む、請求項27に記載の装置。

【請求項 3 1】

前記エラー検出モジュールが、前記復号エラーの前記検出を示すためにエラーフラグを設定し、前記エラーマッピングモジュールは、前記エラーフラグが設定されているとき、前記次の利用可能なユニットの前記開始に基づいて前記破損データセグメントの前記終了を決定する、請求項27に記載の装置。

【請求項 3 2】

前記次の利用可能なユニットが少なくとも1つのスライスを備え、前記復号エンジンは、前記復号エラーが検出されたとき、前記スライスに関連するスライスヘッダを探す、請求項31に記載の装置。

10

【請求項 3 3】

前記復号エラーが検出された場合、前記復号エンジンは、前記カレントユニットの復号を止め、前記次の利用可能なユニットを復号することに進む、請求項31に記載の装置。

【請求項 3 4】

前記復号エラーが検出されなかった場合、前記復号エンジンは、第1のユニット全体を復号する、請求項27に記載の装置。

20

【請求項 3 5】

前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエントロピー符号化工エラーの少なくとも1つを備える、請求項27に記載の装置。

【請求項 3 6】

前記エラーマッピングモジュールが、前記カレントユニットの前記開始に関連する第1のマクロブロック(MB)番号を決定し、前記第2のユニットの開始より前に第2のMB番号を決定し、前記エラー隠蔽モジュールが、前記第1のMB番号から前記第2のMB番号までの前記破損データセグメントを隠蔽する、請求項27に記載の装置。

【請求項 3 7】

前記エラーマッピングモジュールが、前記第2のユニットの前記開始に関連する第3のMB番号を決定し、前記第2のMB番号を生成するために、前記第3のMB番号から1を減算することによって前記第2のMB番号を決定する、請求項36に記載の装置。

30

【請求項 3 8】

前記エラーマッピングモジュールが、前記第1のユニットの前記開始に関連する第1のマクロブロック(MB)番号と第1のフレーム番号とを決定し、前記破損データセグメントの前記終了に関連する第2のMB番号と第2のフレーム番号とを決定し、前記エラー隠蔽モジュールが、前記第1のMB番号及び前記第2のMB番号と前記第1のフレーム番号及び前記第2のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽し、前記第1のフレーム番号と前記第2のフレーム番号とが、同じことも、異なることもある、請求項36に記載の装置。

【請求項 3 9】

前記エラー隠蔽モジュールが、1つ以上のフレームの損失を検出し、別のフレームの参照フレーム不整合を隠蔽するために前記1つ以上の損失したフレームを交換する、請求項27に記載の装置。

40

【請求項 4 0】

前記装置がワイヤレス通信装置ハンドセットを備える、請求項27に記載の装置。

【請求項 4 1】

前記装置が集積回路装置を備える、請求項27に記載の装置。

【請求項 4 2】

符号化ビデオデータのカレントユニットの開始を決定することと、

前記カレントユニットの少なくとも一部分を復号することと、

50

前記カレントユニット中の復号エラーを検出することと、
 前記符号化ビデオデータの次の利用可能なユニットの開始を決定することと、
 前記復号エラーが検出された場合、前記次の利用可能なユニットの前記開始に基づいて
 破損データセグメントの終了を決定することと、
 前記カレントユニットの前記開始と前記破損データセグメントの前記終了とに基づいて
 前記破損データセグメントを隠蔽することと、
 を1つ以上のプロセッサに行わせるための命令を含む、コンピュータ可読媒体。

【請求項43】

前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータスライス
 を含む、請求項42に記載のコンピュータ可読媒体。 10

【請求項44】

前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの少なくとも
 1つの追加のユニットとを含む、請求項42に記載のコンピュータ可読媒体。

【請求項45】

前記カレントユニットが、第1のフレームのカレントスライスであり、前記破損データ
 セグメントが、前記カレントスライスと第2のフレームからの符号化ビデオデータの少なくとも
 1つの追加のスライスとを含む、請求項42に記載のコンピュータ可読媒体。

【請求項46】

前記復号エラーが検出された場合、前記復号エラーの前記検出を示すためにエラーフラ
 20 グを設定することと、

前記エラーフラグが設定されているとき、前記次の利用可能なユニットの前記開始に基
 づいて前記破損データセグメントの前記終了を決定することと
 を前記1つ以上のプロセッサに行わせるための命令を更に含む、請求項42に記載のコン
 ピュータ可読媒体。

【請求項47】

前記次の利用可能なユニットが少なくとも1つのスライスを備え、前記命令は、前記復
 号エラーが検出されたとき、前記スライスに関連するスライスヘッダを探すことと前記1
 つ以上のプロセッサに行わせる、請求項46に記載のコンピュータ可読媒体。

【請求項48】

前記復号エラーが検出された場合、前記カレントユニットの復号を止めることと、前記
 次の利用可能なユニットを復号することに進むことを前記1つ以上のプロセッサに行わ
 せるための命令を更に含む、請求項46に記載のコンピュータ可読媒体。 30

【請求項49】

前記命令は、前記復号エラーが検出されなかった場合、第1のユニット全体を復号すること
 を前記1つ以上のプロセッサに行わせる、請求項42に記載のコンピュータ可読媒体
 。

【請求項50】

前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエントロピー符
 号化工エラーの少なくとも1つを含む、請求項42に記載のコンピュータ可読媒体。

【請求項51】

前記命令が、前記カレントユニットの前記開始に関連する第1のマクロブロック(MB)
)番号を決定することと、前記第2のユニットの開始より前に第2のMB番号を決定することと、前記第1のMB番号から前記第2のMB番号までの前記破損データセグメントを
 隠蔽することとを前記1つ以上のプロセッサに行わせる、請求項42に記載のコンピュ
 タ可読媒体。 40

【請求項52】

前記命令が、前記第2のユニットの前記開始に関連する第3のMB番号を決定することと、前記第2のMB番号を生成するために、前記第3のMB番号から1を減算することと
 を前記1つ以上のプロセッサに行わせる、請求項51に記載のコンピュータ可読媒体。

【請求項53】

10

20

30

40

50

前記命令が、前記第1のユニットの前記開始に関連する第1のマクロブロック(MB)番号と第1のフレーム番号とを決定することと、前記破損データセグメントの前記終了に関連する第2のMB番号と第2のフレーム番号とを決定することと、前記第1のMB番号及び前記第2のMB番号と前記第1のフレーム番号及び前記第2のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽することとを前記1つ以上のプロセッサに行わせ、前記第1のフレーム番号と前記第2のフレーム番号とが、同じことも、異なることもある、請求項51に記載のコンピュータ可読媒体。

【請求項54】

前記命令が、

1つ以上のフレームの損失を検出することと、

10

別のフレームの参照フレーム不整合を隠蔽するために前記1つ以上の損失したフレームを交換することと

を前記1つ以上のプロセッサに行わせる、請求項42に記載のコンピュータ可読媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本開示は、デジタルビデオ復号に関し、より詳細には、ビデオ復号エラーを処理するための技法に関する。

【背景技術】

【0002】

デジタルビデオデータを符号化及び復号するために、幾つかのビデオ符号化及び復号技法が開発されてきた。例えば、Moving Picture Experts Group(MPEG)は、MPEG-1、MPEG-2、及びMPEG-4を含む幾つかの技法を開発した。他の例には、International Telecommunication Union(ITU)-T H.263規格、並びにITU-T H.264規格及びその相対物、ISO/IEC MPEG-4、Part 10、即ち、Advanced Video Coding(AVC)がある。これらのビデオ規格は、データ量を低減するために圧縮された形でデータを符号化することによって、ビデオデータの効率的な送信及び記憶をサポートする。

【0003】

情報源は、上記のビデオ符号化技法の1つを採用して、デジタルビデオデータを符号化することができます。情報源は、符号化ビデオデータをアーカイブし、及び/又は符号化ビデオデータを送信チャネルによって宛先装置に送信する。宛先装置は、符号化ビデオデータを受信し、受信したビデオデータを復号して、再生のために元のデジタルビデオデータを復元する。ビデオデータは、エラーを起こしやすいチャネルでの送信中に損失又は破損し、復号エラーを生じることがある。ビデオブロードキャスト及びビデオ電話など、様々な適用例にとってエラーロバストネスは重要である。

【0004】

ビデオデコーダは、安定したビデオ復号をサポートするために、受信したビットストリーム中のエラーを検出しなければならない。更に、ビデオデコーダはまた、品質へのエラーの影響を低減するために、エラー処理を実行しなければならない。復号エラーが検出されたとき、ビデオデコーダは、復号同期が回復できるまで、破損したデータを隠蔽することができる。適切なエラー処理を行わないと、デコーダが正しいデータを破棄するか、又は信頼できない再構成されたデータを表示することがあり、それらの各々は視覚的品質を劣化させる可能性がある。しかしながら、エラー処理は、計算集約的であり、特にハードウェアで実装されたとき、過大な電力を消費する可能性がある。

【発明の概要】

【0005】

本出願は、その内容全体が参照により本明細書に組み込まれる、2008年2月26日に出願された米国仮出願第61/031,438号の利益を主張する。

【0006】

20

30

40

50

本開示は、デジタル符号化ビデオデータを復号するときに発生する復号エラーを効率的に処理するための技法に関する。ビデオデータは、フレーム又はスライスなど、ビデオデータユニットに編成できる。ビデオデータユニットは可変サイズを有し、マクロブロック又はより小さいブロックなど、ビデオブロックを含むことができる。ビデオデコーダは、1つ又は複数のビデオデータユニットの破損データセグメント内のエラーを検出し、隠蔽するために、連続エラー処理プロセスを実行することができる。

【0007】

デコーダは、フレーム又はスライスなど、カレントビデオデータユニットをビデオデータユニットの開始から連続的に復号する。カレントビデオデータユニット中のエラーを検出すると、デコーダは、エラーフラグを設定し、ビットストリーム中の次の利用可能なビデオデータユニットの開始時に復号を再同期させることができる。エラーフラグは、カレントデータユニット中の復号データの少なくとも一部が復号エラーを生じ、破損していることを示す。エラーフラグが設定されている場合、ビデオデコーダは、その後のビデオデータユニットの開始に基づいて破損データセグメントの終了を識別する。

10

【0008】

デコーダは、カレントビデオデータユニットの開始と破損データセグメントの終了との間のデータを隠蔽する。このようにして、デコーダは、正しいデータの損失及び誤ったデータの表示を回避することができる。エラーフラグが設定されておらず、エラーがないことを示す場合、デコーダは、カレントビデオデータユニットの残りを復号し、カレントビデオデータユニットのためのエラー処理及び隠蔽を実行することなしに次の利用可能なビデオデータユニットを復号することに進むことができる。デコーダはまた、損失したビデオフレームなど、損失したビデオデータユニットによって引き起こされた参照ユニット不整合に対処することができる。

20

【0009】

一態様では、本開示は、符号化ビデオデータのカレントユニットの開始を決定することと、カレントユニットの少なくとも一部分を復号することと、カレントユニット中の復号エラーを検出することと、復号エラーが検出された場合、符号化ビデオデータの次の利用可能なユニットの開始を決定することと、次の利用可能なユニットの開始に基づいて破損データセグメントの終了を判断することと、カレントユニットの開始と破損データセグメントの終了とに基づいて破損データセグメントを隠蔽することとを備えるビデオ復号方法を提供する。

30

【0010】

別の態様では、本開示は、符号化ビデオデータのカレントユニットの少なくとも一部分を復号する復号エンジンと、カレントユニット中の復号エラーを検出するエラー検出モジュールと、カレントユニットの開始を決定し、復号エラーが検出された場合、符号化ビデオデータの次の利用可能なユニットの開始を決定し、次の利用可能なユニットの開始に基づいて破損データセグメントの終了を決定するエラーマッピングモジュールと、カレントユニットの開始と破損データセグメントの終了とに基づいて破損データセグメントを隠蔽するエラー隠蔽モジュールとを備えるビデオ復号デバイスを提供する。

40

【0011】

追加の態様では、本開示は、符号化ビデオデータのカレントユニットの開始を決定することと、カレントユニットの少なくとも一部分を復号することと、カレントユニット中の復号エラーを検出することと、復号エラーが検出された場合、符号化ビデオデータの次の利用可能なユニットの開始を決定することと、次の利用可能なユニットの開始に基づいて破損データセグメントの終了を決定することと、カレントユニットの開始と破損データセグメントの終了とに基づいて破損データセグメントを隠蔽することとを、1つ又は複数のプロセッサに行わせる命令を備えるコンピュータ可読媒体を提供する。

【0012】

本開示の1つ又は複数の態様の詳細について添付の図面及び以下の説明において述べる。本開示の他の特徴、目的、及び利点は、説明及び図面、並びに特許請求の範囲から明ら

50

かになろう。

【図面の簡単な説明】

【0013】

【図1】本開示で説明するエラー処理及び隠蔽技法を実行するビデオ符号化及び復号システムを示すブロック図。

【図2】本開示で説明するエラー処理及び隠蔽技法を実行するビデオデコーダの一例を示すブロック図。

【図3】例示的なエラー処理及び隠蔽モジュールを示すブロック図。

【図4】連続エラー処理及び隠蔽技法を実行する際のビデオデコーダの例示的な動作を示すフローチャート。
10

【図5】フレーム内の復号エラーのためのエラー処理及び隠蔽を実行する際のビデオデコーダの例示的な動作を示すフローチャート。

【図6】フレーム内に復号エラーを備える例示的なビットストリームへのエラー処理及び隠蔽技法の適用を示す図。

【図7】2つ以上のフレームにわたる復号エラーのためのエラー処理及び隠蔽を実行する際のビデオデコーダの例示的な動作を示すフローチャート。

【図8】2つ以上のフレームにわたるエラーを含んでいる例示的なビットストリームへのエラー処理及び隠蔽技法の適用を示す図。

【図9A】複数参照フレームを可能にするビットストリームに対するエラー処理及び隠蔽技法の適用を示す図。
20

【図9B】複数参照フレームを可能にするビットストリームに対するエラー処理及び隠蔽技法の適用を示す図。

【図9C】複数参照フレームを可能にするビットストリームに対するエラー処理及び隠蔽技法の適用を示す図。

【図10】複数参照フレームを可能にするビットストリーム中の損失したフレームに対してエラー処理及び隠蔽を実行する際のビデオデコーダの例示的な動作を示すフローチャート。

【図11A】複数参照フレームを可能にする例示的なビットストリームへのエラー処理及び隠蔽技法の適用を示す図。

【図11B】複数参照フレームを可能にする例示的なビットストリームへのエラー処理及び隠蔽技法の適用を示す図。
30

【発明を実施するための形態】

【0014】

本開示は、ビデオデータを復号するときに発生する復号エラーを効率的に処理するための連続エラー処理技法を対象とする。連続エラー処理技法は、エラー隠蔽をサポートするが、隠蔽すべきデータを識別するためのルックアヘッド（先読み）動作を必要としない。本開示の幾つかの態様では、連続エラー処理技法は、特にモバイルデバイスなどの組込みシステム中に実装されたときに、実装の複雑さを低減し、デコーダにおけるバッファサイズ要件を低減し、効率的な電力消費を促進することができる。更に、幾つかの態様では、連続エラー処理技法は、ビデオの視覚的品質を維持するか、又は向上させることに有効である。
40

【0015】

エラー処理では、復号エラーの影響を抑制するために、ビデオ復号エラーが検出されたとき、隠蔽すべきビデオデータを識別する。例えば、ビデオデコーダは、1つ以上のフレーム又はスライスなど、1つ以上のビデオデータユニット内でマクロブロック（MB）などのビデオデータを隠蔽する。例示のために、本開示は、一般にビデオスライスの形態のビデオユニット及びMBの形態のブロックを参照する。しかしながら、本開示は、他のタイプのビデオデータユニット及びブロックに容易に適用可能である。ビデオシーケンスは複数のビデオフレームを含む。各フレームは複数のビデオスライスを含むことができる。各ビデオスライスは、復号プロセスの再同期を可能にする再同期コードワードをもつスラ
50

イスヘッダを含むことができる。

【0016】

エラー隠蔽では、ビデオスライス内のブロックの代わりに他のデータを代入する。様々な隠蔽技法のいずれも使用できる。一般に、ビデオデコーダは、データを別のビデオデータユニットからの同様のデータと交換することによって、破損データを隠蔽することができる。一例として、ビデオデコーダは、スライス中のMBを異なるフレーム中のスライスからの他のMBと交換することによって、そのMBを隠蔽することができる。特に、1つのスライス中のMBは、ビデオシーケンス内で時間的な隣接フレーム中の対応するスライスから空間的に同一位置にあるMBによって交換できる。

【0017】

本開示では、一般に、許容できる視覚的品質を与えながら、単純で電力効率のよい連続エラー処理技法について説明する。エラー処理は、エラー隠蔽は一般に隠蔽すべきMBの番号についての知識を必要とし、エラー処理はそのような情報を与える、という点でエラー隠蔽とは異なる。この情報はエラー範囲と呼ばれる。開示するエラー処理技法とともに使用される特定のエラー隠蔽技法は変更できる。しかしながら、連続エラー処理技法によって生成されるエラー範囲に関して、エラー隠蔽について説明する。

【0018】

動作中、ビデオデコーダは、復号エラーの正確な位置を知ることができず、復号エラーを同時に検出することができない。従って、2つのスライスヘッダ間に存在するMBは、確実に復号されないことがある。視覚的に混乱させる可能性がある、間違って復号されたMBの提示を回避するために、ビデオデコーダは、復号エラーが検出されたときに2つのスライスヘッダ間のすべてのMBを隠蔽するように設計される。このようにして、ビデオデコーダは、場合によっては破損MBをすべて隠蔽することができる。スライスヘッダは、エラーが1つのスライス内にある場合は連続しているか、又は、エラーが複数のスライスにわたる場合は連続していない。

【0019】

どのMBを、またどれだけ多くのMBを隠蔽すべきかを判断するために、デコーダのエラー処理動作は、復号エラーが検出されたカレントスライスの開始と、後のスライスの次の再同期コードワード又はフレームの終了のいずれかとの間にわたるすべてのMBを、ビットストリーム中で次に続くものはどれでも、判断する。言い換えれば、エラー処理動作は、カレントスライス中の最初のMBと、次のスライス又はビデオフレームの終了のいずれかの直前の最後のMBとを識別する。次いで、この情報はエラー隠蔽動作に移動される。

【0020】

フレーム中のMBは番号付けされる。最後のMBは、次のスライスの最初のMBの直前に発生する。スライスなどのビデオユニットのサイズは可変であるので、後のスライス中の最初のMBの番号は容易に知られない。後のスライス中の最初のMBを識別するための1つの手法は、ビットストリーム中でルックアヘッドを行い、カレントスライスの復号を完了する前に、次のスライスヘッダに関連する再同期コードワードを探索することである。次のスライスヘッダをプリデコードし、再同期コードワードを識別することによって、ルックアヘッド動作が次のスライスの最初のMB、従って先行するスライスの最後のMBを決定することができる。このようにして、エラーが検出された場合、最初のMBと最後のMBが利用可能である。最初のMBと最後のMBは、隠蔽のためのデータセグメントを定義する。データセグメントは、1つのスライス又は複数のスライスを含むことができる。

【0021】

フレームが幾つかのスライスに分割されているとき、デコーダは通常、一度に1つのフレームではなく、一度に1つのスライスを復号する。理想的には、デコーダは、カレントスライスに対して復号する必要があるMBの番号を決定するために、次のスライスヘッダを探すためのルックアヘッド動作を適用することができる。復号中にエラーが発見された

10

20

30

40

50

とき、エラーハンドラがエラー隠蔽のためにこれらの番号を与えることができるよう、デコーダは、破損データセグメントの最初のMBと最後のMBを知る必要がある。このようにして、デコーダは、破損データセグメント内のMBのみを隠蔽し、次のスライスヘッダから復号を再開する。スライス構造の助けをかりて、デコーダは、フレーム全体を損失する代わりに、より多くのMBを再構成することができる。

【0022】

ルックアヘッド手法は、幾つかのソフトウェアベースのデコーダ実装形態では効果的であるが、組込みハードウェアデコーダーキテクチャに対して大いに望ましくないことがある。多くの場合、ルックアヘッド動作が実質的にハードウェアの複雑さを増大させ、プロセッサ間でより多くの計算と中断とを必要とし、電力効率を劣化させることがある。更に、復号プロセスが、次のスライスに対するルックアヘッドの後、カレントスライスに戻ることができるよう、ルックアヘッド動作は過大なバッファリングを必要とすることがある。ルックアヘッド動作は、カレントスライスが復号される前に行われ、従って、復号エラーが検出されたか否かにかかわらず、実行される。従って、各スライスをバッファし、次いで復号しなければならない。ビットストリームがエラーを含んでいない場合でも、このルックアヘッド動作はビットストリーム中の全てのスライスに対して適用され、大部分のスライスに対して浪費される複雑な計算が必要になる。

10

【0023】

本開示で説明する連続エラー処理プロセスによれば、ビデオデコーダは、後のスライスの開始を識別するためのルックアヘッドを行う必要がない。代わりに、ビデオデコーダは、後のスライスに進む前に、エラーが検出されるまで、カレントスライスを連続的に復号する。復号の前にビットストリーム中でルックアヘッドを行うのではなく、ビデオデコーダは、エラーが検出されなかった場合、最初に後のスライスの開始を識別することなく、カレントスライス全体を復号することができる。

20

【0024】

このようにして、復号される次のスライスのスライスヘッダが得られるまで、ビデオデコーダはエラー隠蔽動作を延期することができる。デコーダは、カレントスライスを復号するより前に破損データの最後のMB番号を知る必要はないが、そのスライス中で復号エラーが発生したかどうかを示すエラーフラグを維持する。従って、実際には、デコーダは、カレントスライスがエラーを含んでいないと最初に仮定することができる。カレントスライスがエラーを有する場合、デコーダに関連するエラーハンドラは、次のスライスヘッダが復号された後、そのエラーを後で処理することができる。この概念は、あらゆるスライスがエラーを有すると仮定するのであろう典型的なルックアヘッド手法とは逆である。

30

【0025】

本開示による連続エラー処理プロセスでは、カレントスライスの復号中にエラーを検出すると、ビデオデコーダは、エラーフラグを設定し、例えば、後のビデオスライスの開始において再同期コードワードを探し、識別することによって、後のビデオスライスの開始において復号同期を復旧する。後のスライスは、カレントスライスのすぐ後に続くまさにその次のスライスとすることができます。代替的に、復号エラーが、複数のビデオスライス、又は複数のスライスの部分の損失又は破損を生じた場合、後のスライスは、カレントスライスに続くまさにその次のスライスとすることはできない。幾つかの例では、次の利用可能なスライスは、復号されているカレントスライスと同じフレーム中に存在することさえない。

40

【0026】

カレントビデオユニットを復号するか、又は復号エラーを検出すると、ビデオデコーダは、次の利用可能な、即ち、後のスライスに進む。エラーフラグが設定され、エラーが検出されたことがわかった場合、デコーダは、後のスライスの開始に基づいて破損データセグメントの終了を判断する。ただ1つのスライスが破損していた場合でも、破損データセグメントの終了はカレントスライスの終了に対応することがある。代替的に、2つ以上のビデオスライスが破損していた場合、破損データセグメントの終了は後のビデオスライス

50

の終了に対応することがある。

【0027】

カレントビデオスライスの開始と破損データセグメントの終了とを識別するためにエラー処理を実行すると、ビデオデコーダはエラー隠蔽を実行することができる。例えば、ビデオデコーダは、カレントスライスの開始と破損データセグメントの終了との間のデータを隠蔽する。破損データセグメントは、次の利用可能なスライスヘッダの直前に終了する。破損データを隠蔽することによって、ビデオデコーダは、正しいデータの損失及び／又は誤ったデータの表示を回避することができる。エラーフラグが設定されず、カレントスライス中でエラーが検出されなかったことがわかった場合、ビデオデコーダは単に、カレントスライスに対してエラー隠蔽を実行することなく、次の利用可能なビデオスライスを復号することに進むことができる。

10

【0028】

連続エラー処理技法は、本開示の様々な態様によれば、ルックアヘッド動作の必要ななしに、復号及びエラー処理をサポートすることができる。ルックアヘッド動作をなくすことにより連続エラー処理が可能になり、バッファスペース、計算及び電力リソースが少なくて済む。例えば、ビットストリームを、ルックアヘッドのための第1のパスと、後続の復号のための後続の第2のパスとをサポートするためにバッファするのではなく、連続的に復号することができる。

20

【0029】

一般に、ルックアヘッド動作は、復号エラーがビデオシーケンス中に実際に存在するか否かにかかわらず、エラー処理動作の一部として潜在的な復号エラーの範囲を識別しようと試みる。連続エラー処理技法は、ビデオデータユニットを連続的に復号し、復号エラーが検出された場合、エラー処理を起動する。エラーが検出されなかった場合、ビデオ復号は、エラー処理及びエラー隠蔽を行わずに進み、従って、より効率的な復号プロセスをサポートすることができる。

30

【0030】

図1は、本開示で説明するエラー処理及び隠蔽技法を実行するビデオ符号化及び復号システムを示すブロック図である。図1に示すように、システム10は、通信チャネル16を介して符号化ビデオデータを宛先装置14に送信する情報源12を含む。通信チャネル16は、無線周波数(RF)スペクトル又は1つ以上の物理的伝送線路など、ワイヤレス又は有線の任意の通信媒体、又はワイヤレス及び有線の媒体の任意の組合せを備えることができる。通信チャネル16は、ローカルエリアネットワーク、ワイドエリアネットワーク、又はインターネットなどのグローバルネットワークなど、パケットベースのネットワークの一部を形成することができる。通信チャネル16は、一般に符号化ビデオデータを情報源12から宛先装置14に送信するのに好適な任意の通信媒体、又は様々な通信媒体の集合体を表す。

30

【0031】

情報源12は、宛先装置14に送信するためのコード化ビデオデータを生成する。情報源12は、ビデオソース18とビデオエンコーダ20と送信機22とを含むことができる。情報源12のビデオソース18は、ビデオカメラ、予め獲得されたビデオを含んでいるビデオアーカイブ、又はビデオコンテンツプロバイダからのビデオフィードなど、ビデオ獲得装置を含むことができる。更なる代替として、ビデオソース18はソースビデオとしてのコンピュータグラフィックベースのデータ、又はライブビデオ又はアーカイブされたビデオとコンピュータ生成ビデオとの組合せを生成することができる。場合によっては、ビデオ源18がビデオカメラである場合、情報源12は、いわゆるカメラ付き携帯電話又はビデオ電話、又は、携帯電話又は他の装置を含む任意の他のタイプのカメラ付きコンピューティング又は通信装置を形成することができる。他の態様では、ビデオ源18は、情報源12に結合又は統合される。各場合において、獲得されたビデオ、事前獲得されたビデオ、及び／又はコンピュータ生成ビデオは、送信機22と通信チャネル16とを介して情報源12から宛先装置14に送信するために、ビデオエンコーダ20によって符号化さ

40

50

れる。

【0032】

ビデオエンコーダ20はビデオ源18からビデオデータを受信する。ビデオ源18から受信したビデオデータは、ビデオフレームなどの一連のビデオデータユニットを備えるビデオシーケンスで構成される。フレームの一部又は全部は、ビデオスライスなどより小さいビデオデータユニットに分割される。ビデオエンコーダ20は、ビデオデータを符号化するために、個々のビデオフレーム又はスライス内にピクセルのブロック（本明細書ではビデオブロックと呼ぶ）に対して動作することができる。フレーム又はスライスは、複数のビデオブロックを含んでいることができる。ビデオブロックは、サイズを固定することも変更することもでき、指定の符号化規格に応じてサイズが異なることがある。一般にマクロブロック（MB）と呼ばれる 16×16 ピクセルビデオブロックをサブブロックに構成することができる。10

【0033】

一例として、International Telecommunication Union Standardization Sector (ITU-T) H.264/MPEG-4、Part10、Advanced Video Coding (AVC)（以下「H.264 / MPEG-4 AVC」規格）は、ルーマコンポーネント (luma components) では 16×16 、 8×8 又は 4×4 、及びクロマコンポーネントでは 8×8 など、様々なブロックサイズのイントラ予測、並びにルーマコンポーネントでは 16×16 、 16×8 、 8×16 、 8×8 、 8×4 、 4×8 及び 4×4 、並びにクロマコンポーネントでは対応するスケーリングされたサイズなど、様々なブロックサイズのインター予測をサポートする。一般に、MB及び様々なサブブロックをビデオブロックであると考えることができる。従って、MBはビデオブロックであると考えられ、区分又は下位区分された場合、MBは、それ自体がビデオブロックのセットを定義すると考えることができる。幾つかの態様では、エラー処理技法は、スライス又はフレーム内のMB番号に基づいてエラー隠蔽を指示することができる。しかしながら、他の態様では、エラー処理技法は、より大きい又はより小さいビデオブロックサイズを処理することができる。20

【0034】

ビデオフレーム又はビデオスライスなどの様々なビデオデータユニットに対する技法について本開示で説明するが、本技法は、ビデオデータの任意の符号化及び復号に等しく適用可能である。更に、H.264 / MPEG - 4 AVC 規格に従って符号化及び復号されるビデオデータに関する技法について本開示で説明する。ただし、例示のためにこの規格に関する技法について説明する。MPEG - 1、MPEG - 2 及びMPEG - 4におけるMoving Picture Experts Group (MPEG) によって定義された規格、ITU - T H . 263 規格、Society of Motion Picture and Television Engineers (SMPTE) 421Mビデオコーデック規格（一般に「VC - 1」と呼ばれる）、中国のAudio Video Coding Standard Workgroup（一般に「AVC」と呼ばれる）によって定義された規格、及びある規格団体によって定義されるか、又はプロプライエタリ規格としてある組織によって開発された任意の他のビデオ符号化規格など、様々な他のビデオ符号化規格のいずれかにそのような技法を容易に適用することができる。30

【0035】

例示のために、限定なしに、連続エラー処理の適用について、H.264 / MPEG - 4 AVC 符号化に関して説明する。ビデオエンコーダ20は、例えば、H.264 / MPEG - 4 AVC 規格に記載されているように、イントラ符号化及びインター符号化予測方式に従って、各ブロック（例えば、マクロブロック（MB））を符号化することができる。ビデオブロックのイントラ又はインターベースの予測の後に、ビデオエンコーダ20は、上記のH.264 / MPEG - 4 AVC 規格に従って、ビデオブロックに対して幾つかの他の動作を実行することができる。これらの追加の動作は、変換係数を生成するための（H.264 / AVC 又は離散コサイン変換 DCT 中に使用される 4×4 又は 8×8 整数変換などの）変換動作、量子化動作及びエントロピー符号化動作を含むことができる。符号化データは符号化ビデオデータビットストリームの一部を形成することができる40

。符号化フレームは1つ以上のスライスを含んでいることができる。各スライスは、復号プロセスを同期及び/又は再同期させるための1つ以上のコードワードを含むことができる、Network Abstraction Layer (N A L) ユニットヘッダなどのヘッダによって指定できる。各スライスは、M B及びより小さいブロックなど、1つ以上のブロックを含むことができる。

【 0 0 3 6 】

スライスは、多種多様なパターンに従って選択されるブロックを含むことができる。例えば、スライスは、フレームの所与の数の行に連続的に構成されたブロックを含むことができる。スライスは、代替的に、一行に連続的に構成された幾つかのブロックを含むことができ、そこで、スライスは、あるランレンジスの連続的な行順序付きブロックを定義することによって、ビデオデータに適合することができる。別の例として、スライスは、フレームの所与の数の列に構成されたブロック、又はあるランレンジスの連続的に構成された列順序付きブロックを含むことができる。従って、スライスは、固定数又は不变数などの任意の数のブロックを含むことができるが、異なるビデオデータを収容するように適合するために、後続又は前のスライスよりも多い又は少ないブロックを含むことができる。

【 0 0 3 7 】

スライスを符号化した後、ビデオエンコーダ20は、フレームの各スライスを、H . 264 / M P E G - 4 A V C 規格によって定義されたネットワークアブストラクションレイヤ (N A L) ユニットなどのユニットとして指定し、上述のように、各ユニットの開始を示すために、対応するスライス又はH . 264 / M P E G - 4 A V C によって定義されたN A L ユニットヘッダなどの再同期ヘッダを用いて各スライスを符号化することができる。このようにしてビデオデータを符号化することは、頻繁な再同期のための機会を与え、それにより、送信中に損失又は破損したビデオデータの影響を実質的に低減し、損失したビデオデータから生じるエラーの伝搬を低減することによって、復号を改善することができる。

【 0 0 3 8 】

例えば、フレームをスライスに分割することによって、スライス又はその一部分を、チャネル16を介した送信のために单一のパケット内で符号化することができる。そのスライスがチャネル16を介した送信中に損失又は破損した場合、フレーム全体がパケット中で符号化された場合に必要になるような、フレーム全体を交換、再構成又は隠蔽する必要なしに、そのスライスを再構成又は交換することができる。その上、復号が過去のデータ、場合によっては破損したデータにもはや依拠しないように、各再同期ヘッダを使用して復号プロセスを再始動することができるので、スライス又は再同期ヘッダはエラーの伝搬をなくすことができる。従って、情報源12は、送信機22を介して符号化ビデオデータを複数のパケットとして宛先装置14に送信することができ、各パケットは、エラーを復号する効率的な処理をサポートするために、別個のスライスを符号化することができる。

【 0 0 3 9 】

受信機24は、チャネル16を介して情報源12から符号化ビデオデータを受信する。宛先装置14は、受信機24と、ビデオデコーダ26と、ディスプレイデバイス28とを含むことができる。ビデオデコーダ26は、符号化ビデオデータを復号して、表示装置28上で再生するために元のビデオデータを得ることができる。表示装置28は、陰極線管 (C R T) 、液晶表示器 (L C D) 、プラズマ表示器、発光ダイオード (L E D) 表示器、有機L E D 表示器、又は別のタイプの表示器など、様々な表示装置のいずれかを備えることができる。

【 0 0 4 0 】

上述のように、チャネル16を介した符号化ビデオデータの送信中に、符号化ビデオデータの様々なスライス又はM Bは、例えば、干渉、輻輳、又はチャネル16に沿って信号伝搬に影響を及ぼす他の異常により損失又は破損することがある。ワイヤレス適用例の場合、損失又は破損したビデオデータは、エラーを起こしやすいネットワークを介したワイヤレス送信により起こることがある。復号エラーは、ビデオブロードキャスト又はビデオ

電話などの様々な適用例に悪影響を及ぼすことがある。受信機 24 は、損失又は破損したビデオデータを含むことがある符号化ビデオデータを受信する。ビデオデコーダ 26 は、一部のフレーム又はスライス、若しくは複数のフレーム又はスライスにわたるこれらの損失又は破損したデータを復号しようと試みる。これらの損失又は破損したデータは、正しいデータの損失及び／又は誤ったデータの提示を生じることがある復号エラーを生成する。

【 0 0 4 1 】

ビデオデコーダ 26 は、連続エラー処理動作及びエラー隠蔽動作を実行するように構成できる。エラー処理動作では、ビデオデコーダ 26 は、復号エラーを検出し、破損データセグメントと呼ばれる、エラーによって破損されたデータの総範囲を決定する。破損データセグメントは一般に、最初に符号化されたビットストリーム中のデータに対して、幾つかの重大な点で、損失した、破損した、又は誤った受信ビットストリーム中のデータのセグメントを指す。エラーがスライス中に検出された場合、スライス中の全ての MB は好ましくは隠蔽される。しかしながら、復号エラーによる影響を受ける最初の MB と最後の MB を決定することが必要である。カレントスライスの最初の MB は、そのスライスの復号の開始時に容易に決定できる。しかしながら、スライスの可変サイズのために、スライスの最後の MB は容易に知られない。その上、復号エラーが複数のスライスにわたる場合、隠蔽すべき破損データセグメント中の最後の MB は、後のスライス内に存在することがある。言い換れば、破損データセグメントは、1つのスライス又は複数のスライスを含むことがある。

10

20

【 0 0 4 2 】

いずれの場合も、隠蔽すべき最後の MB は、適切に受信される次の利用可能なスライスの最初の MB を検出することによって、例えば、スライスに関連するヘッダ中で関連する次の再同期マーカの検出に基づいて識別される。その場合、潜在的な破損データの最後の MB は、次の利用可能なスライスの最初の MB の直前の MB として識別される。本開示で説明する連続エラー処理技法は、ルックアヘッド動作を必要とすることなく、MB の識別情報を隠蔽することを可能にする。このようにして、連続エラー処理技法は、ビデオデコーダ 26 によって実行されるエラー処理及び隠蔽動作の効率を向上させることができる。

【 0 0 4 3 】

一般に、次のスライスをバッファし、ルックアヘッドする代わりに、ビデオデコーダ 26 は、最初に符号化ビデオデータのカレントスライスの開始を決定し、カレントスライスを復号し始める。復号エラーが検出されたとき、ビデオデコーダ 26 は、受信したビデオデータの次の利用可能なスライスの開始を決定することによって、復号を再同期させることに進むことができる。例えば、復号エラーに応答して、ビデオデコーダ 26 は、カレントスライスを復号することを止め、ビットストリームを解析することによって識別できる次の利用可能なスライスを復号することに進むことができる。次の利用可能なスライスはビットストリーム中の後のスライスである。後のスライスの開始に基づいて、ビデオデコーダ 26 は、破損データセグメントの終了を決定することができ、その終了は、カレントスライスの終了か、又はカレントスライスと次の利用可能なスライスとの間に存在する別のスライスの終了とすることができます。ビデオデコーダ 26 が検出されたエラーを含むスライスを識別したことを考えて、ビデオデコーダ 26 は、カレントスライスの開始と次の利用可能なスライスの開始とに基づいて破損データセグメント中のエラーを隠蔽することができる。

30

40

【 0 0 4 4 】

例示として、ビデオデコーダ 26 は、フレームを受信し、フレームの第 1 のスライスの開始を決定することができる。スライスがフレーム中の第 1 のスライスである場合、第 1 のスライスの開始は、第 1 のスライスの MB 番号、例えば、MB 番号 0 を含むことができる。次に、ビデオデコーダ 26 は、ルックアヘッド手法とは逆に、第 1 のスライスの最後の MB を決定することなく、第 2 のスライス又はスライスヘッダにアクセスすることさえなく、第 1 のスライスを復号することができる。このようにして、ビデオデコーダ 26 は

50

、通常ならば次のスライスをルックアヘッドし、次いで復号のためにカレントスライスに戻るために必要となるのである追加のバッファをなくすことができる。

【0045】

第1のスライス中にエラーを検出すると、ビデオデコーダ26は、発生した復号エラーを示すためにエラーフラグを設定することができる。エラーフラグは、それぞれエラー又はエラーがないことを示すように設定されリセットされる単一のビット、又はエラーインジケータの他のタイプ、メッセージ、コマンド、ビットパターン、コードワードなど、様々な形態のいずれかをとることができ、デコーダに関連するメモリ又はレジスタに記憶される。エラーが検出されると、ビデオデコーダ26は、次の利用可能なスライスの開始を決定することによって、例えば、ビットストリーム中で発見できる次のスライスヘッダを参照することによって、復号プロセスを再同期させることに進むことができる。ここでも、この次の利用可能なスライスは、符号化ビットストリーム中のまさにその次のスライスとすることはできず、代わりに、受信したビットストリーム中の次のスライスである。従って、次の利用可能なスライスは、複数のスライスの損失の結果として、ビットストリーム中の第3、第4又は第5の（又は後の）スライスとなる可能性がある。その上、更に以下で説明するように、次の利用可能なスライスは、第1のスライスと同じフレーム内に含まれることさえない。しかしながら、連続エラー処理技法は、単一のスライス、複数のスライス、更には複数のフレームにわたる復号エラーを効率的に処理するために、これらの例の各々において適用できる。

10

【0046】

復号エラーがどれくらい多くのスライス又はフレームにわたるかにかかわらず、ビデオデコーダ26は、次の利用可能なスライスの開始に基づいて、復号エラーを含む破損データセグメントの終了を決定する。一例として、第1のスライスを復号すると、ビデオデコーダ26は、第1のスライスの開始をMB番号0として決定し、復号エラーを検出し、次いで、MB番号Nにおいて次の利用可能なスライスの開始を探す。これは、破損データセグメントが1つ以上のスライスにわたっているが、単一のフレーム内に存在すると仮定する。MB番号Nに基づいて、ビデオデコーダ26は、復号エラーを含む破損データセグメントの終了がMB番号N-1であると決定することができる。例えば、第2のスライス中の最初のMB番号が55である場合、破損データセグメント中の最後のMBのMB番号は54となる。この決定を用いて、ビデオデコーダ26のエラー処理動作は、MB0からMB54にわたる破損データセグメントを隠蔽しなければならないことを示す。

20

【0047】

ビデオデコーダ26は、多くの隠蔽技法に従ってエラーを隠蔽することができる。ビデオデコーダ26は、例えば、MB番号0~54を、前に復号されたフレームなど、隣接フレームからのMB番号0~54と交換することによって、エラーを隠蔽することができる。代替的に、ビデオデコーダ26は、補間、外挿又は動き補償補間など、より精巧な技法を採用して、復号エラーによって潜在的に影響を受けるMBを交換することができる。スライスを復号してエラー隠蔽を実行するときに、ビデオデコーダ26は、必要に応じて、ユーザに提示するために復号されたビデオデータを表示装置28に転送できる。

30

【0048】

ルックアヘッド動作を回避した結果として、ビデオデコーダ26は、頻繁なルックアヘッド動作のためにビットストリームの部分をバッファすることを必要とすることなく、ビットストリームを連続的に復号することに進むことができる。その上、ビデオデコーダ26は、実際のエラーの隠蔽をサポートが必要なときに、エラーが存在するか否かにかかわらず、ルックアヘッド動作を連続的に実行するのではなく、エラー処理動作を起動することができる。多くの適用例では、エラーは、わずかな割合のビデオスライスのみに発生する。連続的にではなく、オンデマンドでエラー処理を適用し、それにより連続ルックアヘッドのバッファリングを回避することによって、ビデオデコーダ26は、消費する計算リソース及び電力リソースをより小さくし、実装のための計算の複雑さを低減することができる。ビデオデコーダ26が、モバイルワイヤレス通信デバイスなどの制限され

40

50

た電力リソースをもつ組込みシステムの一部を形成するとき、電力節約は特に重要である。

【0049】

システム10の可能な実装形態の更なる詳細について次に説明する。場合によっては、情報源12及び宛先装置14は、ほぼ対称に動作することができる。例えば、情報源12及び宛先装置14はそれぞれ、ビデオ符号化及び復号コンポーネントを含むことができる。従って、システム10は、例えばビデオストリーミング、ビデオブロードキャスト又はビデオ電話のための装置12と装置14との間の一方向又は双方向のビデオ送信をサポートすることができる。本明細書で説明する連続エラー処理技法は、例えば、複合コーデック中の符号化コンポーネントと復号コンポーネントの両方を含む装置に適用可能である。10
更に、幾つかの態様では、エラー処理技法は、復号すべきデータを符号化したエンコーダが存在する同じ装置中に存在するデコーダ内に適用される。この場合、例えば、符号化データは、宛先装置14などの別の装置に送信されるのではなく、ローカルでアーカイブされ、次いでローカル再生のために復号される。エラー処理技法は、この例では、データ記憶媒体に対するビデオデータ書き込み及び読み取り動作中に破損したか、又は他のファクタにより破損した符号化ビデオデータに適用される。

【0050】

前述のように、ビデオエンコーダ20及びビデオデコーダ26は、Moving Picture Experts Group (MPEG) - 2、MPEG - 4、ITU-T H.263又はH.264/MPEG - 4 AVCなどのビデオ圧縮規格に従って動作することができる。図1には示されていないが、幾つかの態様では、ビデオエンコーダ20及びビデオデコーダ26はそれぞれ、オーディオエンコーダ及びデコーダとそれぞれ統合でき、適切なMUX-DEMUXユニット、又は他のハードウェア及びソフトウェアを含み、共通のデータストリーム又は別個のデータストリーム中のオーディオとビデオの両方の符号化を処理することができる。このようにして、情報源12及び宛先装置14は、オーディオデータ及びビデオデータを含むマルチメディアデータに対して動作することができる。適用可能な場合、MUX-DEMUXユニットはITU-H.223マルチプレクサプロトコル、又はユーザデータグラムプロトコル(UDP)など他のプロトコルに準拠することができる。20

【0051】

H.264/MPEG - 4 AVC規格は、合同ビデオチーム(JVT)として知られる共同パートナーシップの成果として、ISO/IEC MPEGとともにITU-T Video Coding Experts Group (VCEG)によって公式化された。幾つかの態様では、本開示で説明する技法は、一般にH.264規格に準拠するデバイスに適用することができる。H.264規格は、ITU-T研究グループによる2005年3月付けのITU-T勧告H.264「Advanced Video Coding for generic audiovisual services」に記載されており、本明細書ではH.264規格又はH.264仕様、又はH.264/AVC規格又は仕様と呼ぶ。30

【0052】

場合によっては、ビデオエンコーダ20及びビデオデコーダ26は、空間、時間及び/又は信号対雑音比(SNR)スケーラビリティのためのスケーラブルビデオ符号化(SVC)をサポートするように構成できる。ビデオエンコーダ20及びビデオデコーダ26は、幾つかの態様では、SVCのための微細精度SNRスケーラビリティ(FGS)符号化(fine granularity SNR scalability (FGS) coding)をサポートするように構成できる。ビデオエンコーダ20及びビデオデコーダ26は、ベースレイヤ及び1つ以上のスケーラブルエンハンスマントレイヤの符号化、送信及び復号をサポートすることによって様々な程度のスケーラビリティをサポートすることができる。スケーラブルビデオ符号化の場合、ベースレイヤはベースラインレベルの品質でビデオデータを搬送する。1つ以上のエンハンスマントレイヤは追加データを搬送して、より高い空間的レベル、時間的レベル及び/又はSNRレベルをサポートする。40

【0053】

10

20

30

40

50

ベースレイヤは、エンハンスマントレイヤの送信よりも信頼できる方法で送信できる。例えば、変調信号の最も信頼できる部分はベースレイヤを送信するために使用でき、変調信号のあまり信頼できない部分はエンハンスマントレイヤを送信するために使用できる。ベースレイヤ及びエンハンスマントレイヤは、物理レイヤ上の階層変調を使用して符号化されるので、同じキャリア又はサブキャリア上で送信できるが、伝送特性が異なることによりパケットエラーレート(P E R)が異なることになる。

【0054】

幾つかの態様では、ビデオブロードキャストに関して、本開示で説明する技法を、Forward Link Only(FLO) Air Interface Specification、即ちTechnical Standard TIA-1099(「FLO Specification」)として2007年7月に発表された「Forward Link Only Air Interface Specification for Terrestrial Mobile Multimedia Multicast」を使用して、terrestrial mobile multimedia multicast(TM3)システムでリアルタイムビデオサービスを配信するための拡張H.264ビデオ符号化に適用することもできる。例えば、チャネル16は、FLO仕様などに従ってワイヤレスビデオ情報をブロードキャストするために使用されるワイヤレス情報チャネルを備えることができる。FLO仕様は、ビットストリームシンタックス及びセマンティックス、並びにFLO Air Interfaceに適した復号プロセスを定義する例を含む。

【0055】

代替的に、DVB-H(digital video broadcast-handheld)、ISDB-T(integrated services digital broadcast-terrestrial)、又はDMB(digital media broadcast)などの他の規格に従ってビデオをブロードキャストすることができる。従って、様々な態様では、情報源12は、モバイルワイヤレス端末、ビデオストリーミングサーバ、又はビデオブロードキャストサーバとすることができます。しかしながら、本開示で説明する技法は、特定のタイプのブロードキャスト、マルチキャスト、又はポイントツーポイントシステムに限定されない。ビデオブロードキャストの場合、情報源12は、ビデオデータの幾つかのチャネルを、その各々が図1の宛先装置14と同様とすることができます複数の宛先装置にブロードキャストすることができます。従って、図1には単一の宛先装置14が示されているが、ビデオブロードキャストの場合、情報源12は、より典型的には、ビデオコンテンツを多くの宛先装置に同時にブロードキャストすることができます。

【0056】

送信機22、通信チャネル16、及び受信機24は、イーサネット(登録商標)と電話(例えば、POTS)とケーブルと電力線と光ファイバシステムとのうちの1つ以上を含む有線又はワイヤレス通信システム、及び/又はワイヤレスシステムによる通信のために構成でき、ワイヤレスシステムは、符号分割多元接続(CDMA又はCDMA2000)通信システム、周波数分割多元接続(FDMA)システム、直交周波数分割多元(OFDM)接続システム、GSM(登録商標)(Global System for Mobile Communication)又はGPRS(General packet Radio Service)若しくはEDGE(拡張データGSM環境)などの時分割多元接続(TDMA)システム、TETRA(Terrestrial Trunked Radio)携帯電話システム、広帯域符号分割多元接続(WCDMA)システム、high data rate 1xEV-DO(First generation Evolution Data Only)又は1xEV-DO Gold Multicastシステム、IEEE802.18システム、MediaFLO(登録商標)システム、DMBシステム、DVB-Hシステム、若しくは2つ以上のデバイス間のデータ通信のための別の方程式のうちの1つ以上を備える。

【0057】

ビデオエンコーダ20及びビデオデコーダ26はそれぞれ、1つ以上のマイクロプロセッサ、デジタル信号プロセッサ(DSP)、特定用途向け集積回路(ASIC)、フィールドプログラマブルゲートアレイ(FPGA)、ディスクリート論理、ソフトウェア、ハードウェア、ファームウェア又はそれらの任意の組合せを用いて実装できる。ビデオエンコーダ20及びビデオデコーダ26の各々を1つ以上のエンコーダ又はデコーダ中に含めることができ、そのいずれかは複合エンコーダ/デコーダ(コードック)の一部としてそ

10

20

30

40

50

れぞれモバイル装置、加入者装置、ブロードキャスト装置、サーバなどに統合できる。更に、情報源 1 2 及び宛先装置 1 4 はそれぞれ、適用可能なとき、符号化ビデオの送信及び受信のために適切な変調、復調、周波数変換、フィルタ処理、及び増幅器コンポーネントを含み、適用可能な場合、ワイヤレス通信をサポートするために十分な無線周波（R F）ワイヤレスコンポーネント及びアンテナを含むことができる。しかしながら、説明を簡単にするために、図 1 では、そのようなコンポーネントは概して、情報源 1 2 の送信機 2 2 及び宛先装置 1 4 の受信機 2 4 であるとして要約されている。

【0058】

図 2 は、本開示で説明するエラー処理及び隠蔽技法を実行するビデオデコーダ 2 6 の一例を示すブロック図である。ビデオデコーダ 2 6 は、図 1 の宛先装置 1 4 のビデオデコーダに対応する。図 2 の例に示すように、ビデオデコーダ 2 6 は、受信バッファ 3 0 と、復号エンジン 3 2 と、エラー処理モジュール 3 4 と、エラー隠蔽モジュール 3 6 と、再構成（RECON）バッファ 3 8 を含む。一般に、受信バッファ 3 0 は、ビデオエンコーダ 2 0 によって符号化されたデータなどの符号化ビデオデータを受信し、記憶する。再構成バッファ 3 8 は、復号エンジン 3 2 によって復号され、再構成された復号ビデオデータを記憶する。再構成バッファ 3 8 中の復号ビデオデータは、受信バッファ 3 0 から得られた予測符号化ビデオデータの復号の際に復号エンジン 3 2 によって使用される参照フレームを与えるために使用できる。更に、再構成バッファ 3 8 から得られた復号ビデオデータは、復号ビデオビットストリーム中のエラーを隠蔽するためにエラー隠蔽モジュール 3 6 によって使用できる。再構成バッファ 3 8 中の再構成されたデータは、様々な後処理動作のうちのいずれかに復号ビデオデータとして与えられる。ビデオデコーダ 2 6 によって生成された復号ビデオデータは、任意の適用可能な後処理を受けて、表示装置を介してユーザに提示するために出力バッファに与えられる。復号エンジン 3 2 、エラー処理モジュール 3 4 及びエラー隠蔽モジュール 3 6 は協働して、復号ビデオデータを生成する。復号エンジン 3 2 、エラー処理モジュール 3 4 及びエラー隠蔽モジュール 3 6 は、本開示で説明する連続エラー処理プロセスを適用するように構成される。

【0059】

図 1 及び図 2 に含まれ、示される様々なコンポーネント、ユニット又はモジュール、並びに本開示全体にわたって説明する他のコンポーネントは、ハードウェア及び／又はソフトウェアの任意の好適な組合せによって実現できる。図 1 及び図 2 、並びに図 3 では、様々なコンポーネントは、別個のコンポーネント、ユニット、又はモジュールとして示される。しかしながら、図 1 ~ 図 3 を参照しながら説明する様々なコンポーネントの全て又は幾つかは、共通ハードウェア及び／又はソフトウェア内の組合せユニット又はモジュールに統合できる。従って、コンポーネント、ユニット、又はモジュールとしての特徴の表現は、説明しやすいように特定の機能的特徴を強調するものであり、別々のハードウェア又はソフトウェアコンポーネントによるそのような特徴の実現を必ずしも必要としない。

【0060】

場合によっては、様々なユニットを、1つ以上のプロセッサによって実行されるプログラマブルプロセスとして実装できる。例えば、本開示における様々な特徴を、離散モジュール、ユニット、エンジン又はコンポーネントとして説明したが、それらの特徴は、集積論理装置中で一緒に実装でき、又は個別であるが、相互運用可能な論理装置として別個に実装できる。場合によっては、様々な特徴は、集積回路チップ又はチップセットなどの集積回路装置として実装できる。ソフトウェアで実装した場合、本技法は、実行されると、1つ以上のプロセッサに本開示で説明する方法のうちの1つ以上を実行させる命令を備えるコンピュータ可読記憶媒体によって少なくとも部分的に実現できる。

【0061】

図 2 を更に参照すると、受信バッファ 3 0 は、符号化ビデオデータを記憶するためのダイナミックランダムアクセスメモリなどのメモリモジュールによって実現できる。符号化ビデオデータは、説明したように、1つ以上のフレーム又はスライスなどのビデオデータユニットを備えることができる。フレーム又はスライスは、MB 及びより小さいブロック

10

20

30

40

50

などのブロックを含むことができる。例示のために、符号化ビデオデータは複数のフレームを備え、各フレームは1つ以上のスライスを備え、各スライスは複数のMBを備えると仮定する。

【0062】

復号エンジン32は、受信バッファ30から得られた符号化ビデオデータビットストリームを解析し、符号化ビデオデータを復号する。復号エンジン32は、本開示で説明するビデオ符号化規格のうちの1つ、又は他のビデオ符号化規格によって定義された復号技法を実装することができる。しかしながら、例示のために、復号エンジン32はH.264/MPEG-4 AVC規格による復号技法を実行すると仮定する。復号エンジン32は、復号すべき符号化情報のフレーム、スライス、ブロック及び他のユニットを識別するために、ビット解析を実行して、着信ビットストリームを解析する。更に、復号エンジン32は、符号化情報のユニット中のビデオデータを復号し、再構成するために、再構成を実行する。

10

【0063】

エラー処理モジュール34は、本開示で説明する連続エラー処理技法に従って復号エラーを処理するように構成される。図2の例では、エラー処理モジュール34は、エラー検出とエラー処理の両方を実行することができる。しかしながら、幾つかの実装形態では、別個のエラー検出及びエラー処理モジュールが設けられる。動作中、エラー処理モジュール34は、復号エンジン32によって生成された復号ビデオデータ中の復号エラーを検出し、正しいデータの損失又は誤ったデータの表示を回避するために隠蔽すべきMBなどのデータの範囲を識別する。

20

【0064】

エラー隠蔽モジュール36は、エラー処理モジュール34から復号エラー範囲（エラー範囲）を受信する。復号エラー範囲は、どのMBをエラー隠蔽モジュール36によって隠蔽すべきかを指定する。例えば、エラー処理モジュール34は、ビットストリーム中の第1のスライスの最初のMB0から、次の利用可能なスライスの開始MBNより前の最後のMBN-1にわたる全てのMBをエラー範囲として指定する。エラー隠蔽モジュール36は、復号エンジン32によって生成された識別されたMBを異なるMBと交換するために、様々な隠蔽技法のいずれかに従ってエラーを隠蔽する。例えば、エラー隠蔽モジュール36は、隠蔽されたビデオデータを復号エンジン32に与える。エラー隠蔽モジュール36は、再構成バッファに与えられた復号ビデオデータ中に、隠蔽されたビデオデータを含むことができる。例示として、エラー処理モジュール34がエラー範囲としてMB0～MB54を指定した場合、エラー隠蔽モジュール36は、MB0～54を隠蔽して、隠蔽されたMB[0,54]を復号エンジン32に与えることになる。

30

【0065】

再構成バッファ38は、受信バッファ30と同様に、復号エンジン32によって生成された復号ビデオデータを記憶するダイナミックランダムアクセスメモリなどのメモリモジュールを表す。更に、復号エンジン32は、受信バッファ26から得られた符号化ビデオデータを復号する際に使用するための再構成バッファから復号ビデオデータを取り出す。特に、復号エンジン32は、他のフレームの復号のための参照フレームとして、再構成バッファ38からの復号ビデオデータを使用する。幾つかの実装形態では、エラー隠蔽モジュール36は、エラーが検出されたとき、復号されたMBを隠蔽するために再構成バッファ38内の指定されたMBを上書きすることができる。代替的に、図2の例の場合のように、復号エンジン32が復号MBを再構成バッファ38に転送する前に、エラー隠蔽モジュール36は、指定されたMBを隠蔽するために、復号エンジン32と連携することができる。例えば、エラー隠蔽モジュール36は、隠蔽されたMBなどの隠蔽されたビデオデータを復号エンジン32に与える。次いで、復号エンジン32は、破損データをエラー隠蔽モジュール36からの隠蔽されたビデオデータと交換し、得られたデータを復号ビデオデータとして再構成バッファ38に書き込む。

40

【0066】

50

エラーが検出されたとき、復号エンジン32は、次の利用可能なスライスヘッダのポイントまでのMBを復号し続けるか、又はMBを復号することを止め、ビットストリーム中の次の利用可能なスライスヘッダを探す。エラーが検出されたときにカレントスライスの復号を終了し、エラー隠蔽をサポートするために復号すべき次のスライスに移動することが、より効率的である。次いで、破損データを隠蔽したときに、復号エンジン32は、カレントスライスとして次のスライスの復号を再開する。従って、エラー隠蔽モジュール36は、破損データセグメントを隠蔽するためにMBを書き込むか又は上書きするように構成できる。いずれの場合も、再構成バッファ38に記憶された復号ビデオデータは、復号エンジン32によって生成された復号MBに基づく復号ビデオデータと、復号エラーを隠蔽するためにエラー隠蔽モジュール36によって生成又は選択されたMBとを含むことができる。

10

【0067】

動作中、ビデオデコーダ26の受信バッファ30は、最初に図1の受信機24などの受信機から符号化ビデオデータを受信し、符号化ビデオデータを記憶する。復号エンジン32は、フレームのスライスなど、受信バッファ30に記憶された符号化ビデオデータのユニットを取り出すために、受信バッファ30にアクセスする。スライスを取り出した後だが、スライスを復号するより前に、復号エンジン32は、例えば、復号すべきカレントスライスの最初のMBに関連するMB番号を送信することによって、最初にエラー処理モジュール34にスライスの開始について通知する。この最初のMBはスライス開始MBと呼ばれる。エラー処理モジュール34は、スライスの開始において最初のMBの番号を記憶する。

20

【0068】

復号エンジン32は、スライス中のMBを復号することに進み、復号スライスデータを再構成バッファ38に記憶する。スライスを復号しながら、復号エンジン32は、復号スライスを再構成バッファ38に記憶することに加えて、復号ビデオデータをエラー処理モジュール34に出力することもできる。様々な実装形態では、エラー処理モジュール34は、再構成バッファ38に記憶された復号ビデオデータ、復号エンジンから直接与えられた復号ビデオデータ、又は異なるバッファに記憶された復号ビデオデータを分析する。各場合において、エラー処理モジュール34は、復号エラーが復号ビデオデータ中に存在するかどうかを決定するために、復号ビデオデータを分析するか、又は検査する。従って、エラー検出機能をエラー処理モジュール34内に設けることができる。代替的に、様々な実装形態では、復号エンジン32、又はビデオデコーダ26内の別のコンポーネントが、復号エラー検出機能を含むことがある。

30

【0069】

エラー処理モジュール34は、様々な復号エラーを検出する多くの方法で、復号ビデオデータを分析することができる。例えば、エラー処理モジュール34は、シンタックスエラー、セマンティックエラー、及び／又は他の符号化工ラーを、受信バッファ30によって受信し復号エンジン32によって復号した符号化スライス内で検出する。一般に、復号エンジン32は、例えば、H.264/MPEG-4 AVCなどの適用可能な符号化技法によって指定されたシンタックスを使用して、ビットストリームを復号する。ビットストリームが、適用可能なシンタックスによって指定された予想されるビットパターンから逸脱した場合、シンタックスエラーを検出する。復号が、範囲外データ、無効な情報、又は適用できないコマンドなど、間違った値又は動作をビデオデコーダ26中で生成したとき、セマンティックエラーを検出する。エントロピー符号化値が、例えば、コンテキスト適応型可変長符号化(CAVLC)テーブルなどの適用可能な符号化テーブル中の値又はパターンに一致しなかった場合、符号化工ラーを検出する。コンテキスト適応型2値算術符号化(CABAC)などの他のタイプのエントロピー符号化を使用することができる。

40

【0070】

従って、シンタックスエラー検査を実行するために、エラー処理モジュール34は、復号ビットストリームのビットパターンを、適用可能なシンタックスによって指定された予

50

想されるビットパターンと比較する。ビットパターンが、予想されるビットパターンと矛盾している場合、エラー処理モジュール34は復号エラーが検出されたことを示す。同様に、セマンティックエラー検査を実行するために、エラー処理モジュール34は、値を1つ以上のシンタックス要素に分析して、これらの値が有効であることを確実にする。例えば、エラー処理モジュール34は、値が他の値と矛盾しないかどうか、現在復号されたスライスと矛盾しないかどうか、及び／又は前に復号されたフレーム、スライス又はブロックと矛盾しないかどうかを決定する。シンタックス要素値が、許容される又は予想される値の範囲内になかった場合、エラー処理モジュール34は復号エラーの検出を示す。符号化エラーを検出するために、エラー処理モジュール34は、復号されたコードワードを、C A V L C 又はC A B A C テーブルなどの適用可能なエンタロピー符号化テーブル中のコードワードのリストと比較する。一致がない場合、エラー処理モジュール34は復号エラーの検出を示す。

10

【0071】

復号エラーを検出すると、エラー処理モジュール34は、エラーフラグをビデオデコーダ26に関連するメモリ中に設定し、復号エンジン32について通知する。復号エンジン32は、検出された復号エラーについて通知されたことに応答して、受信バッファ30に記憶された符号化ビデオデータの復号を再同期させる。具体的には、復号エンジン32は、カレントスライスを復号することを止め、ビットストリームから、例えば次の識別可能なスライス又はN A L ユニットヘッダによって識別される次の利用可能なスライスを取り出すことに進む。次のスライスは、ビットストリームを解析することによって復号エンジン32が見分けることができる次のスライスであるという点で、次の「利用可能な」スライスと呼ばれる。復号エラーは、N A L ヘッダ及び／又はフレームヘッダにわたって伝搬していることがあり、それによって、スライス間の区別、更にはフレーム間の区別を、不可能でないとしても困難にする。言い換えれば、復号エラーにより、単一のスライス中にM B の損失又は破損を生じることがあり、その場合、次の再同期ポイントは、受信したビットストリーム又は複数のスライス中のまさにその次のスライスに関連することがあり、ビットストリーム中の後のスライスに関連することがある。従って、次の利用可能なスライスは、例えば、識別可能なN A L ユニットヘッダによって容易に見分けることができる、ビットストリーム中の次のスライスを指すことがある。N A L ヘッダは、破損しておらず、容易に見分けることができる点で、「識別可能」と呼ばれる。

20

30

【0072】

復号エラーが検出されたとき、復号エンジン32は、上述のように、カレントスライスの残りを復号することなく、ビットストリーム中の次の利用可能な再同期ポイントに進むことがある。代替的に、エラーフラグが設定されていても、復号エンジン32は、ビットストリームを復号し続けることがある。次の利用可能なスライスが得られたとき、次いで、復号エンジン32はエラー処理及びエラー隠蔽を実行する。復号エンジン32は、エラーフラグが設定された後、ビットストリームを復号し続けるように構成できるが、通常、カレントスライスを復号するのを止め、代わりにエラーが検出された前のスライスのエラー処理及びエラー隠蔽をサポートするために次のスライスに進むことがより効率的であろう。破損データセグメントは、復号された場合でも、エラー隠蔽モジュール36によって隠蔽されることになる。しかしながら、エラーが検出されたときに復号動作を中止し、次の利用可能なスライスに進むことは、より効率的である。復号エラーが検出されなかったとき、復号エンジン32はカレントスライスの残りを復号することに進む。いずれの場合も、復号エンジン32は、次の利用可能なスライスの開始を示す次の再同期ポイントに到達したとき、そのスライスの最初のM B を決定する。更に、復号エンジン32は、エラー処理モジュール34が、前のビデオスライス中の復号エラーの検出を示すためにエラーフラグを設定したかどうかを決定する。

40

【0073】

エラーフラグが設定されていない場合、復号エンジン32はスライスの最初のM B を記憶し、そのスライスを復号することに進む。この時点で、連続エラー処理手法の場合、前

50

のスライスは、ルックアヘッド手法の場合のように第2のパス上で復号するためにバッファされるのではなく、すでに復号されている。ただし、エラーフラグが設定されている場合、復号エンジン32はエラー処理及びエラー隠蔽を起動する。特に、次の利用可能なスライスに到着したときに、復号エンジン32では、符号化工エラーの検出の後にそのスライスに進んだのか、又は符号化工エラーがない場合で前のスライスの復号を完了したときなのかを識別しなくてよい。むしろ、次の利用可能なスライスに到着すると、復号エンジン32及びエラー処理モジュール34は、エラーフラグを検査して、通常の過程における連続復号を単に続けるべきか、又はエラー処理及びエラー隠蔽を適用すべきかを決定する。言い換えれば、エラー処理モジュール34は、次の利用可能なスライスに到着したのが、通常のエラーのない復号の結果としてなのか、又はエラー処理を必要とするエラーの検出の結果としてなのかを知らなくてよい。

10

【0074】

再び、エラーが検出されたとき、復号エンジン32はカレントスライスを復号することを止め、次の利用可能なスライスを取り出す。次いで、復号エンジン32は、例えばスライスの最初のMBに対応するMB番号を転送することによって、エラー処理モジュール34にその（次の利用可能な）スライスの開始について通知する。エラー処理モジュール34は、このMB番号を受信すると、エラーフラグが設定されているかどうかを検査することによって、復号エラーが前に発生したかどうかを決定する。フラグが設定されていない場合、エラー処理モジュール34は、例えば復号エラーが新たに取り出された次の利用可能なスライス中に検出された場合に後で使用するために、上述のように、この最初のMB番号を記憶する。

20

【0075】

ただし、エラーフラグが設定されている場合、エラー処理モジュール34はエラーが発生したカレントスライスの終了、例えば最後のMBを決定する。例えば、エラー処理モジュール34は、次の利用可能なスライスの最初のMBに対応するMB番号から1を減算する。復号エラーが複数のスライス及び更に複数のフレームにわたるとき、破損データセグメントは、同様に1つ以上のスライス、及びノ/又は1つ以上のフレームにわたる。連続エラー処理技法は、以下でより詳細に説明するように、1つ以上のスライス、又は1つ以上のフレームにわたる復号エラーを処理することができる。

30

【0076】

エラーが発生した破損データセグメントの開始及び終了を決定した後、エラー処理モジュール34は、復号エラー範囲（エラー範囲）と呼ばれるこの情報をエラー隠蔽モジュール36に転送する。この復号エラー範囲に基づいて、エラー隠蔽モジュール36は、再構成バッファ38に記憶された復号ビデオデータを、再構成バッファ38に記憶されることがある、前に復号されたビデオデータと交換することによって、復号エラー範囲によって示される復号エラーを隠蔽する。代替的に、エラー隠蔽モジュール36は、隠蔽されたビデオデータを復号エンジン32に与え、次いで、復号エンジン32は、復号データと隠蔽されたデータとの再構成バッファ38への転送を処理することがある。幾つかの例では、エラー隠蔽モジュール36は、復号エラーを隠蔽する際に使用するために、前に復号されたビデオデータ、例えば前に復号されたMB、スライス又はフレームをキャッシュに格納するか又は局所的に記憶する。代替として、破損データを隠蔽するために、エラー隠蔽モジュール36は、例えば、図2の例に示すように、再構成バッファ38から予め復号されたビデオデータを得ることがある。

40

【0077】

エラー隠蔽モジュール36は、前に復号されたビデオデータにアクセスする方法にかかわらず、上記に記載された隠蔽技法のうちの1つに従って、復号エラー範囲に対応する復号ビデオデータを前に復号されたビデオデータと交換する。エラー処理モジュール34によって識別される復号エラー範囲がMB0からMB54にわたる場合、例えば、エラー隠蔽モジュール36は、MB0からMB54までの全てのMBを隠蔽する。このようにして、全ての起こりうる破損MBを隠蔽することを確実にするために、スライス全体と同様の

50

ものを隠蔽する。

【0078】

エラー隠蔽モジュール36は、隠蔽されたビデオデータを再構成バッファ38に直接書き込む。代替的に、図2に示すように、エラー隠蔽モジュール36は、隠蔽されたビデオデータを復号エンジン32に与えることがある。次いで、復号エンジン32は、適切に復号されたMBと、エラー隠蔽モジュール36からの隠蔽されたビデオデータによって表される隠蔽されたMBとを再構成バッファ38に書き込む。概して本開示で説明するように、エラー隠蔽モジュール36はピクセル領域中のエラー隠蔽を適用することができる。幾つかの実装形態では、エラー隠蔽モジュール36は、エラー隠蔽のための圧縮領域プロセスを適用することができる。例えば、エラー隠蔽モジュール36は、破損MBを隠蔽するために符号化モード（例えば、SKIPモード）を指定するように構成される。従って、本開示で説明する技法は、ピクセル領域隠蔽プロセス又は圧縮領域隠蔽プロセスを含む多種多様なエラー隠蔽プロセスとともに使用される。10

【0079】

破損MBが隠蔽されると、ビデオデコーダ26は、復号ビデオデータに適用される、例えばシャープ化、スケーリングなどの任意の後処理動作を受けた復号ビデオデータを、上述のように、再構成バッファ38を介して表示又は再生のための表示装置28に転送する。次いで、復号エンジン32は、上述した同じエラー処理動作を適用しながら、次の利用可能なスライスを復号することに進む。従って、このエラー処理プロセスは、復号エンジン32がビットストリーム中の後続のスライス及びフレームを通過する際に連続的に適用される。特に、様々な態様では、エラー処理プロセスは、ルックアヘッド動作の必要なしに、連続復号を可能にする。このようにして、ビデオデコーダ26は、宛先装置14内の計算効率及び電力節約を促進することができる。20

【0080】

図3は、図2の例示的なエラー処理モジュール34及びエラー隠蔽モジュール36をより詳細に示すブロック図である。図3の例は、例示のために、エラー処理モジュール34及びエラー隠蔽モジュール36の様々な動作態様を強調するために与えたものであり、本開示で広く説明する連続エラー処理技法を制限するものと考えるべきではない。図3の例では、エラー処理モジュール34は、エラーチェックモジュール40とエラーマッピングモジュール42とを含むことができる。エラーチェックモジュール40は、シンタックスルール44A、セマンティックルール44B及びエントロピールール44Cなどの適用可能なルールに従って、シンタックスエラー検査、セマンティックエラー検査、及び符号化（例えば、CAVL C又はCABAC）エラー検査などの上述のエラー検査を実行する。エラーチェックモジュール40は、エラーステータスインジケータの一例を表すエラーフラグ46を更に含み、維持する。エラーマッピングモジュール42は、破損データセグメントを定義するためにマッピング48を維持する。エラー隠蔽モジュール42は、上述の隠蔽技法のうちの1つ以上を実行する隠蔽エンジン50と、破損データを隠蔽する際に隠蔽エンジン50が使用するビデオデータを受信するビデオバッファ51とを含むことができる。30

【0081】

一例として、受信バッファ30に記憶された符号化ビデオデータを復号するより前に、復号エンジン32は、復号すべきカレントスライスのために、スライスの開始指示（「スライス開始」）をエラー処理モジュール34に送信する。エラーチェックモジュール40及びエラー処理モジュール42は、スライス開始を使用して、復号動作中に検出されたエラーを処理することができる。スライスの開始は、1つ以上のスライス又は部分スライスを含む破損データのより大きいセグメントの開始とすることができるので、スライス開始は、図3に示すように、より一般的には「セグメント開始」と呼ばれる。40

【0082】

ここでも、モジュール及びユニットは、エラー処理モジュール34の様々な機能的態様を強調するために図3及び他の図に示されている。従って、エラーチェックモジュール40、エラーマッピングモジュール42、又は他のモジュールの間の正確な構造的及び機能的関50

係並びに相互作用は、変更されることがある。エラーマッピングモジュール42は、上述のように、復号すべきカレントスライスの最初のMBを備えるセグメント開始を記憶する。MB番号は、復号ビットストリーム中のシンタックス要素から決定できる。例えば、エラーマッピングモジュール42は、最初のMBを開始MB48Aとして記憶する。カレントスライスがフレーム中の最初のスライスである場合、最初のMBはMB0とすることができます。カレントスライスがフレーム中の後のスライスである場合、最初のMBはより大きい番号を有する。

【0083】

セグメント開始を受信すると、エラー検出モジュール40は、シンタックス要素を有する復号MBを備える、復号エンジン32によって生成された復号ビデオデータを分析する。エラー検出モジュール40は、復号されたシンタックス要素に対して、シンタックスエラー検査、セマンティックエラー検査、及びエントロピー符号化エラー検査を実行する。特に、エラー検出モジュール40は、復号ビデオデータ中に復号エラーがあるかどうかを決定するために、シンタクスルール44A、セマンティックルール44B及びエントロピールール44Cを適用する。これらのエラー検査の結果に基づいて、エラー検出モジュール40は、エラー状態を記憶することが可能なエラーフラグ46又は任意の他のインジケータの値を制御する。例示のために、エラー検出モジュール40が復号エラーを検出したと仮定すると、エラー検出モジュール40は、エラーフラグ46を設定し、エラーの指示(図3の「復号エラー」)をビデオデコーダ26の復号エンジン32に転送する。

【0084】

エラー指示に応答して、復号エンジン32は、例えば、カレントスライスの復号を止め、ビットストリーム中の次の利用可能なスライスを探すことによって、復号を中止し、復号を再同期させる。言い換えれば、復号エラーが検出された場合、復号エンジン32はカレントスライスの復号を止め、次の利用可能なスライスを復号することに進むことができる。ここでも、次の利用可能なスライスは、次のスライスのNALユニットヘッダ中の再同期コードワードによって識別される。代替的に、エラー指示がない場合、復号エンジン32は、スライスの終了に達し、次の利用可能なスライスに遭遇するまで、カレントスライスを復号し続けることがある。いずれの場合も、次の利用可能なスライスに達すると、復号エンジン32は、別のセグメント開始指示をエラー検出モジュール40に転送する。エラー検出モジュール40は、新しいセグメント開始、例えば、スライス開始を受信し、エラーフラグ46の状態に基づいて、復号エラーが前に復号されたスライス中に存在したかどうかを決定する。

【0085】

エラーフラグ46の状態に基づいて、エラー検出モジュール40は、復号ビデオを分析し続けるか、又はエラーマッピングモジュール42を起動することができる。例えば、エラーフラグ46が設定されず、エラーが検出されなかつことがわかった場合、エラー検出モジュール40は、新たに受信したスライスに対してエラー検出を実行することに進む。しかしながら、エラーフラグ46が設定され、エラーが検出されたことがわかった場合、エラー範囲、即ち、隠蔽されなければならない破損データセグメントの範囲を決定するために、エラーマッピングモジュール42がエラー検出モジュール40によって起動される。この場合、エラーマッピングモジュール42は、新たに受信したスライス中の最初のMBの番号を決定する。更に、エラーマッピングモジュール42は、復号エラーが、新たに受信したスライス中の最初のMB番号に基づいて検出された破損データセグメント中の最後のMB又は終了MBを決定する。

【0086】

例えば、終了MB番号を決定するために、エラーマッピングモジュール42は、新しいスライス開始のすぐ前に先行する最後のMBの番号を得るために、新たに受信したスライスの最初のMB番号から1を減算する。図3の例で示すように、終了MBは、終了MB48Bとして記憶される。例えば、新たに受信したスライス中の最初のMBが55である場合、破損データ中の最後のMBは55 - 1 = 54となる。従って、エラーマッピングモジ

10

20

30

40

50

ユール42は、エラー範囲を、開始MB48Aと終了MB48Bとの間にわたり、開始MB48Aと終了MB48Bとを含むものとして、例えば、上記の例ではMB0～MB54として定義する。エラー範囲は、復号ビデオデータに対するエラーの悪影響をなくすか又は低減するために隠蔽すべきMBを示す。

【0087】

以下で説明するように、エラー検出モジュール40はまた、幾つかの態様では、開始MB番号48Aと終了MB番号48Bに対応するMBに関連するシンタックス要素の値から、開始フレーム番号と終了フレーム番号とを決定する。即ち、エラー検出モジュール40は、開始MB番号48Aに対応する、受信した復号ビデオデータ中に表されるMBを検査し、そのMBが関連する開始フレーム番号を決定する。エラー検出モジュール40は、その開始フレーム番号をエラーマッピングモジュール42に転送し、エラーマッピングモジュール42は、開始フレーム番号を開始フレーム48Cとして記憶する。エラー検出モジュール40はまた、終了MB番号48Bに対応する、受信した復号ビデオデータ中に表されるMBを検査し、そのMBが関連する終了フレーム番号を決定する。エラー検出モジュール40は、その終了フレーム番号をエラーマッピングモジュール42に転送し、エラーマッピングモジュール42は、終了フレーム番号を終了フレーム48Dとして記憶する。

10

【0088】

上述のように、エラーマッピングモジュール42は、マッピング48からエラー範囲又は他のエラー記述を生成し、そのエラー範囲をエラー隠蔽モジュール36にパスすることができる。エラー範囲は、幾つかの実装形態では、破損データセグメントの開始MB48A及び終了MB48Bだけでなく、開始フレーム48C及び終了フレーム48Dをも含むことができる。例えば、エラーマッピングモジュール42は、復号エラー（及び得られた破損データセグメント）が、開始フレーム48C及び終了フレーム48Dを比較することによって単一のフレーム内にあるかどうか、又は、復号エラーが2つ以上のフレームにわたるかどうかを決定する。MB番号付けは、各フレームの開始時に通常0にリセットされる。従って、復号エラーが単一のフレームよりも多くのフレームに影響を及ぼすとき、MB番号を提示するのみのエラー範囲を与えることは十分ではないことがある。例えば、開始MBが第1のフレーム中のMB74である場合、終了MBは第2のフレーム中のMB56とすることができます。従って、隠蔽目的のためのエラー範囲を正確に解決するために、フレーム番号並びにMB番号を追跡することが必要である。

20

【0089】

一般に、開始フレーム48Cと終了フレーム48Dとによって表される2つのフレーム番号が等しい場合、エラーマッピングモジュール42は、復号エラーが、開始フレーム48Cと終了フレーム48Dの両方によって識別されるフレーム番号に対応する単一のフレーム内にあると決定する。しかしながら、開始フレーム番号48Cと終了フレーム番号48Dとが等しくない場合、エラーマッピングモジュール42は、復号エラーが2つ以上のフレームにわたるように、エラー範囲を決定する。従って、エラーマッピングモジュール42は、開始MB番号48A及び終了MB番号48B、並びに開始フレーム番号48C及び終了フレーム番号48Dを使用して隠蔽すべきMBの範囲を識別するように構成できる。

30

【0090】

エラー隠蔽モジュール36は、エラー範囲を受信し、隠蔽エンジン50を採用してエラー範囲によって識別されたMBを隠蔽する。エラー隠蔽エンジン50は、エラー範囲によって識別されたそれらのMBを隠蔽する際に使用するために、前に復号されたフレームの前に復号されたMBを取り出し、ビデオバッファ51に記憶する（図3の「バッファされたビデオデータ」）。上述のように、幾つかの態様では、前に復号されたMBは、1つ以上の隣接フレームから得られ、隠蔽すべきMBと空間的にコロケートされる。前に復号されたMBは、交換フレームデータを形成するために、エラー範囲中の潜在的に破損したMBを交換し、それによって隠蔽する。幾つかの実装形態では、エラー隠蔽モジュール36

40

50

は、再構成バッファ 3 8 からの復号ビデオデータとして、前に復号された M B を得る。

【 0 0 9 1 】

隠蔽エンジン 5 0 は、エラー範囲によって識別された復号エラーを隠蔽するために、様々な他の隠蔽技法のいずれかを実装することができる。エラー隠蔽エンジン 5 0 は、隠蔽されたビデオデータを交換データとしてビデオ復号エンジン 3 2 に与える。ビデオ復号エンジン 3 2 は、適切に復号されたビデオデータと隠蔽されたビデオデータとを再構成バッファ 3 8 に与える。代替的に、隠蔽されたフレームデータを使用して、復号エンジン 3 2 によって生成され、再構成バッファ 3 8 に記憶された復号ビデオデータの一部に書き込むか、又はその一部を上書きする。いずれの場合も、再構成バッファ 3 8 は、表示装置を駆動する際に使用するための復号ビデオデータを与える。

10

【 0 0 9 2 】

図 4 は、連続エラー処理及び隠蔽技法を実行する際の、ビデオデコーダ 2 6 などのビデオデコーダの例示的な動作を示すフローチャートである。上述のように、ビデオデコーダ 2 6 は、図 1 の受信機 2 4 などの受信機から符号化ビデオデータを受信する(52)。ビデオデコーダ 2 6 は、符号化ビデオデータを受信バッファ 3 0 中にバッファする。復号エンジン 3 2 は、符号化ビデオデータのカレントスライスの開始を決定し(54)、スライスを復号する(56)。後述するように、エラーが検出された場合、エラー処理モジュール 3 4 がカレントスライスの開始を使用して、破損データセグメントのためのエラー範囲を識別する。エラー処理モジュール 3 4 が復号エラーを検出せず(58)、復号がカレントスライスの終了に達しなかった(59)場合、復号エンジン 3 2 はスライスを復号し続ける(56)。

20

【 0 0 9 3 】

復号エラーの検出なしに、スライスの終了に達した(59)場合、復号エンジン 3 2 は、追加の符号化データのうちの次の利用可能なデータを受信する(52)ことに進み、そのスライスに対して動作(54)、(56)及び(58)を繰り返す。復号エラーが検出されなかった(58)とき、ビデオデコーダ 2 6 は、ビットストリーム中の連続するスライスを復号し続ける。しかしながら、復号エラーが検出された(58)とき、エラー処理モジュール 3 4 は、符号化ビデオデータの次の利用可能なスライスの開始を決定し(60)、次の利用可能なスライスの開始に基づいて、(カレントスライスを含む)破損データセグメントの終了を決定する(62)。

30

【 0 0 9 4 】

ここでも、次の利用可能なスライスは、ビットストリーム中で識別できる次の利用可能なセグメントである。従って、破損データが単一のスライスのみに影響を及ぼす場合、次の利用可能なスライスは、復号エラーが検出されたスライスに続くまさにその次のスライスとすることができます。一方、エラーがデータの 2 つ以上のスライスの損失又は破損を生じた場合、次の利用可能なスライスは、当初符号化されたビットストリーム中でカレントスライスのすぐ後に続いていなかった後のスライスとすることができます。

40

【 0 0 9 5 】

上述のように、次の利用可能なスライスの開始に基づいて(60)、エラー処理モジュール 3 4 は、破損データセグメントに関連する破損データセグメントの終了が次の利用可能なスライス中の最初の M B 番号から 1 を減算することによって得られると決定する。エラー処理モジュール 3 4 は、隠蔽のためのエラー範囲を定義するために破損データセグメントの開始及び終了をエラー隠蔽モジュール 3 6 に与える。データ損失の範囲に応じて、エラー範囲は、元のビットストリームの単一のスライス又は複数のスライスをカバーすることがある。

【 0 0 9 6 】

エラー範囲を用いて、エラー隠蔽モジュール 3 4 は、破損データセグメントの開始から破損データセグメントの終了にわたるビデオデータを隠蔽する(64)。破損データセグメントを隠蔽すると、ビデオデコーダ 2 6 は、次の利用可能なスライスに関連する符号化ビデオデータを受信する(52)ことに進むので、次の利用可能なスライスを、即ち、新

50

しいカレントスライスとして復号する。この場合、ビデオデコーダ 2 6 がビデオビットストリームに沿って進む際に、次の利用可能なスライスは、次に図 4 に示すプロセスのためのカレントスライスになり、図 4 で概説する様々な動作を繰り返し、連続的な復号、エラー処理及び隠蔽をサポートする。

【 0 0 9 7 】

上述のように、エラー処理モジュール 3 4 は、復号エラーが、シンタックスエラー検査に基づいて発生したのか、セマンティックエラー検査に基づいて発生したのか、又はエントロピー符号化工エラー検査に基づいて発生したのかを決定する(58)。エラー処理モジュール 3 4 がエラーを検出した場合、エラー処理モジュール 3 4 は、復号エンジン 3 2 にエラーについて通知する。エラー指示に応答して、ビデオデコーダ 3 2 は、受信バッファ 3 0 に記憶された符号化ビデオデータの次の利用可能なスライスの開始を決定する(60)ために、ビットストリームを解析することによって、符号化ビデオデータの復号を再同期させる。特に、復号エンジン 3 2 は、復号エラーが発生したときには再同期のためのビットストリームを解析することによって、又はエラーが検出されなかったときにはカレントスライスを復号する通常の過程でのいずれかで、次の利用可能なスライスに到着する。エラーが検出された場合、エラー処理モジュール 3 4 はエラー範囲、例えば、第 1 のセグメントの開始及び終了を決定し、エラー範囲をエラー隠蔽モジュール 3 6 に与える。

10

【 0 0 9 8 】

図 5 は、フレーム内の復号エラーのためのエラー処理及び隠蔽を実行する際の、ビデオデコーダ 2 6 などのビデオデコーダの例示的な動作を示すフローチャートである。H.264 / M P E G - 4 A V C 規格に従って編成されたビットストリームに関して本技法について以下で説明するが、符号化ビデオデータは複数のフレームを備え、各フレームは複数のスライスを備え、各スライスは複数の符号化 MB を備える。スライスは N A L ユニットヘッダによって識別できる。H.264 規格に関して説明するが、本技法は、やはりこの例示的な態様に厳密に限定されるべきではないが、ビデオデータの符号化のための他の技法に広く適用可能である。

20

【 0 0 9 9 】

図 5 は、概して図 4 に準拠するが、幾つかの追加の詳細を示す。図 5 の例では、ビデオデコーダ 2 6 は、受信バッファ 3 0 を介して符号化ビデオフレームデータを受信する(66)。ビデオデコーダ 2 6 の復号エンジン 3 2 は、第 1 のフレームの第 1 のスライスに最初にアクセスするときに、第 1 のスライスに関連する開始 MB 番号を決定する(68)。第 1 のスライスは、代替的に、復号エンジン 3 2 によって復号されるカレントスライスと呼ばれる。復号エンジン 3 2 がカレントスライスを復号する(70)一方、エラー処理モジュール 3 4 は、例えば、シンタックスエラー、セマンティックエラー又はエントロピー符号化工エラーなどの復号エラーを検出する(72)。復号エラーを検出した(72)場合、エラー処理モジュール 3 4 はエラーフラグを設定する(74)。

30

【 0 1 0 0 】

復号エラーを検出せず(72)、復号エンジン 3 2 がカレントスライスの終了にまだ達しなかった(76)場合、復号エンジン 3 2 はカレントスライスを復号し続ける(70)。復号エラーが検出された(72)場合、又はカレントスライスの終了に達した(76)場合、エラー処理モジュール 3 4 は、次の利用可能なスライスの開始 MB 番号を決定する(78)。エラーが検出されたとき、復号エンジン 3 2 は、復号し続けるか、又はカレントスライスを復号するのを止め、次の利用可能なスライスの位置を特定するためにビットストリームを解析する。エラーが検出されなかった(72)とき、復号エンジン 3 2 は、カレントスライスの復号を完了した後に次の利用可能なスライスに達し、その場合、スライスの終了に達する(76)。従って、復号エンジン 3 2 は、前のスライスの正常な復号時に、又は前のスライス中の復号エラーの検出に続く再同期の一部としてのいずれかで、次の利用可能なスライスに到着する。

40

【 0 1 0 1 】

エラーフラグが設定されていない(80)場合、復号エンジン 3 2 は、次の利用可能な

50

スライスをカレントスライスとして復号する(70)ことに進む。従って、次の利用可能なスライスは、図5で概説する動作のためのカレントスライスになり、復号プロセスが繰り返される。しかしながら、エラーフラグが設定されている(80)場合、エラー処理モジュール34は、次の利用可能なスライスの開始MB番号に基づいて、カレントスライスを含む破損データセグメントの終了MB番号を決定する(82)。例えば、エラー処理モジュール34は、次の利用可能なスライスの開始MB番号から1を減算して、第1のスライスを含むセグメントの終了MB番号を決定する(82)。

【0102】

1つのスライスが破損していた場合、終了MB番号はカレントスライスの終了MB番号とすることができます。しかしながら、2つ以上のスライスが破損していた場合、終了MB番号は後のスライスの終了MB番号とすることができます。いずれにしても、カレントスライスの開始MB番号と終了MB番号とを使用して、エラー処理モジュール34は、エラー隠蔽モジュール36が使用するエラー範囲を定義する。エラー隠蔽モジュール36は、カレントスライスの開始MB番号から破損データセグメントの終了MB番号までの全てのMBを隠蔽することによって、復号エラーを隠蔽する(84)ことに進む。エラー隠蔽を完了する(84)と、ビデオデコーダ26は、復号すべき次のスライスのためのエラーフラグをリセットする、即ち、復号すべき次のスライスにとって適切ではない、長く記憶された状態をクリアする。更に、ビデオデコーダ26は、カレントスライスとして次の利用可能なスライスを復号する(70)ことに進み、図5で概説する様々なエラー処理動作を繰り返す。

10

20

【0103】

図6は、フレーム内に復号エラーを備える例示的なビットストリーム86へのエラー処理及び隠蔽技法の適用を示す図である。ビットストリーム86は、一般にH.264/MPEG-4 AVC規格に従ってフォーマッティングできる。図6の例では、復号エラーは、ビットストリームの中央に発生し、1つのNALユニットヘッダを含んでいる。エラー隠蔽のためのMB0～54を識別するために、連続エラー処理技法を図6のシナリオに適用することができる。図6は、2つのフレーム88A、88B(「フレーム88」)を示し、フレーム88Aは複数のNALヘッダ90A～90Dを備え、フレーム88Bは複数のNALヘッダ90E～90Hを備える。NALヘッダ90Aと90Eとは、それぞれのフレーム88の開始を示すことを表すために、それぞれ影付きで示される。NALヘッダ90A～90H(「NALヘッダ90」)の各々は、スライスを含んでいるNALユニットの開始を示す。各フレーム88は、1つ以上のスライス、及びより可能性がある複数のスライスを含むことができる。各NALユニットは、NALヘッダ90の後続のヘッダの開始時に終了する。

30

【0104】

例えば、第1のスライスは、NALヘッダ90Aと、MB番号0～20に対応する後続の複数のMBとを備えることができ、MB番号20は後続のNALヘッダ90Bの前の最後のMBである。次のスライスは、NALユニットヘッダ90B及びMB番号21で開始する。フレーム88の各々の上部に例示的なMB番号を示す。そのような番号の間隔は、実際のビットストリームのスケール又は比を必ずしも反映するわけではなく、例示のために与えられる。上述のように「スライス」を含んでいるNALユニットは、独立して復号できる。言い換えれば、スライスは、他のNALユニットからの情報なしに復号できる。このようにして、各NALユニットは復号可能MBの自蔵式シーケンスであるので、NALユニットの損失は、所与のフレームに対する先行する又は後続のNALユニットの復号を必ずしも妨げるわけではない。

40

【0105】

前述のように、一部のビデオデコーダ、特にソフトウェアモジュールとして実装されたビデオデコーダは、ルックアヘッドエラー処理技法を実行することによって復号エラーを処理することができる。カレントスライスを復号するために、ルックアヘッド技法は一般に、第1のスライスの最後のブロック又はMBを決定するために、次の利用可能なスライ

50

スヘッダ、例えばN A Lヘッダを復号するためのビデオデコーダを必要とする。即ち、ビデオデコーダは、次のスライスヘッダに対してルックアヘッドを行い、現在復号されているスライスの範囲を決定することができる。カレントスライスを復号している間に復号エラーが検出された場合、隠蔽すべき一部の符号化ビデオデータを識別するために、ビデオデコーダは、カレントスライスを復号するより前に、カレントスライスの最後のM Bを更に識別することができる。ビデオデコーダは、符号化ビデオデータのスライスごとにこのルックアヘッド動作を実行することができるが、それは、非効率、即ち計算の複雑さ及び過大な電力消費を生じることがある。

【0106】

ルックアヘッドエラー処理技法を採用するビデオデコーダは、非連続復号を実行するものとして特徴づけられる。一例として、従来のビデオデコーダは、最初にカレントスライスの開始M Bを決定し、次いで、次の再同期コードワードを検出するためにビットストリーム中でルックアヘッドを行い、それによって、カレントスライス中の最後のM Bを決定する。特に、再同期コードワードは、次に適切に受信したスライスを識別するために使用される。ルックアヘッドを行った後、次いで、デコーダは、カレントスライスのバッファバージョンを復号することに戻る。次のスライスのための最初のM B番号Nを決定することによって、ビデオデコーダは、前のスライス中の最後のM B番号、即ちM B N - 1を決定することができる。最後のM B番号を識別すると、ビデオデコーダは、カレントビデオスライスの開始M Bに戻り、スライスを復号し始める。

【0107】

従って、ルックアヘッドエラー処理技法の実装は、カレントスライス及び次のスライスをバッファするための2つの異なる復号バッファを必要とする。ビデオデコーダは、元のビデオデータを適切に再構成するために、各バッファ間で交互に切り替える。ビデオデコーダは、1つのスライスの現在の状態情報のすべてを記憶し、次のスライスに移動し、次いで第1のスライスに戻る必要があるので、この技法の非連続的な性質により、実装がより錯綜又は複雑化することがある。特に全体的な復号タスクが2つ以上のプロセッサ間で分割されたとき、この動作は組込みシステムでは望ましくないことがある。

【0108】

図6の例を参照しながら、ルックアヘッド動作に基づくエラー処理プロセスについて最初に説明する。この例では、フレーム88Aは、M B 0からM B 20にわたる第1のスライスと、M B 21からM B 54にわたる第2のスライスと、M B 55からM B 71にわたる第3のスライスと、M B 72からM B 98にわたる第4のスライスとを含む。カレントスライスを復号するために、ビデオデコーダは、最初にルックアヘッドを行い、最初の利用可能な再同期ポイントを発見する。M B 15～M B 45が損失、又は破損していると仮定した場合、ビデオデコーダは、N A Lヘッダ90Cにおいて次の再同期ポイントを発見する。その間に、ビデオデコーダは、復号エラーについての実際の知識を有しないが、復号エラーがあった場合、再同期ポイントより前の最後のM Bを得ることができるように、ルックアヘッド動作を通して次の再同期ポイントを得る必要がある。

【0109】

従って、ルックアヘッド動作に依拠するエラー処理プロセスでは、ビデオデコーダは、復号すべきカレントスライスの最初のM B番号を得て、次のスライスヘッダを復号してそのスライスヘッダより前の最後のM B番号を得るために、ルックアヘッドを行い、復号すべきカレントスライスの開始に戻り、復号及びエラーの検出を再開し、最初のM Bと最後のM Bとをエラー隠蔽動作にパスし、復号すべき次のスライスに対して復号同期及びエラー隠蔽を回復する。特に、ビデオデコーダは、ルックアヘッド動作を実行している間、スライス1及び2(M B 0～54)をバッファし、次いで、復号動作を開始するために、スライス1に戻る必要がある。一般に、ビデオデコーダは、実際の復号エラーが存在するか否かにかかわらず、ビットストリーム中で復号すべき全てのスライスに対してこれらのルックアヘッド動作及びバッファ動作を実行する。スライスの実際の復号は、ルックアヘッド動作の後までも行われない。

10

20

30

40

50

【0110】

チャネル16を介した符号化ビデオデータの送信はしばしば、符号化ビデオデータの顕著な損失又は破損なしに行われる。例えば、例示のために、符号化ビデオデータが複数のパケットとしてチャネル16上で送信されると仮定すると、典型的なワイヤレスチャネル16のパケット損失レートはしばしば、5%よりも小さい。例示のために、各パケットが1つのスライスに対応すると仮定すると、有効な符号化ビデオデータの復号中に復号エラーが発生しないことになるので、ルックアヘッドエラー処理技法のための計算の95%が必要とされることになる。従って、ルックアヘッドエラー処理技法は、多数の浪費される動作を生じることになる。

【0111】

エラー処理をソフトウェアで実装したときは、これらの浪費される動作及び上述の他の問題はあまり重要ではないが、特に全体的な復号タスクを2つ以上のプロセッサに分割したときは、組込みシステムがこれらの問題によって大きな影響を受けることがある。特に、ルックアヘッド技法の組込み実装形態、又はより一般的にルックアヘッド技法のハードウェア実装形態では、浪費される動作は、より多くの有益な動作に使用できる電力を不需要に消費することがある。その上、電力消費は、宛先装置14と同様のモバイル宛先装置に共通のバッテリを消耗させるので、無駄な電力消費により、バッテリが再充電を必要とする前の動作可能な時間が短くなることがある。更に、2つのプロセッサがこれらの電力問題を悪化させることがある。更に、次のスライスをバッファするために使用される追加のバッファは、電力を消耗し、従来のビデオデコーダのコスト及びサイズを更に増大させることがある。

10

20

【0112】

図2のビデオデコーダ26など、ビデオデコーダは、フレーム88を受信し、本開示で説明する技法に従って連続エラー処理動作、即ち、非ルックアヘッドエラー処理動作を実行することができる。連続エラー処理プロセスの場合、各スライスの復号順序は順序正しく維持できる。1つのスライスを事前にバッファし、次いで、次のスライスのNALヘッダにアクセスするためにルックアヘッド動作を実行する必要はない。復号エンジン32は、フレーム88Aの第1のNALユニット又はスライス、例えば、NALヘッダ90A及びMB0~20にアクセスし、最初のMB番号、例えば、MB0を決定することによって、カレントスライスの開始を決定することができる。復号エンジン32は、このMB番号0を、図3のエラー検出モジュール40など、エラー処理モジュール34のエラー検出モジュールに転送することができる。上述のように、エラーフラグ46を事前にリセットした後、エラー検出モジュール40は、MB番号0をエラーマッピングモジュール42に転送し、エラーマッピングモジュール42は、MB番号0をマッピング48の開始MB48Aに記憶する。

30

【0113】

一方、復号エンジン32は、符号化MB0~20を復号し、この復号ビデオデータ又は復号MBを再構成バッファ38に記憶する。復号エンジン32はまた、復号MB0~20をエラー検出モジュール40に転送し、エラー検出モジュール40は、エラー検査の適用によって復号エラーが発生したかどうかを決定することができる。例えば、エラー検出モジュール40は、それぞれシンタックスルール44A及びセマンティックルール44Bに従って、シンタックスエラー検査及びセマンティックエラー検査を実行し、並びにエントロピールール44Cに従ってエントロピー符号化検査を実行することができる。

40

【0114】

図6の例では、MB0~20の少なくともいくつか、NALヘッダ90B、及びMB21~54のいくつかは、損失又は破損している。エラー検出モジュール40は、この復号エラーを検出し、エラーを示すためにエラーフラグ46を設定する。フラグ46を設定することによって、エラー検出モジュール40は、次の識別可能なスライスヘッダ、例えば、NALヘッダ90Cが復号されるまで、エラー隠蔽を遅延させることができる。従って、ビデオデコーダ26は、カレントスライスを復号するより前に最後のMB番号を決定す

50

るためルックアヘッドを行う必要はないが、前のスライスがエラーであることを示すために、フラグ46を維持する。このようにして、ビデオデコーダ26は、データがエラーを含んでいないという仮定に基づいて符号化ビデオデータのカレントスライス又はセグメントを復号する。しかしながら、エラーを決定すると、後で、例えば、次のスライスヘッダが復号された後に、エラー処理モジュール34がエラーを処理することができるよう、エラーフラグ46が設定される。

【0115】

例示のために、エラー検出モジュール40が上述の復号エラーを検出し、エラーフラグ46を設定したと仮定すると、エラー検出モジュール40は復号エンジン32に復号エラーについて通知する。従って、復号エンジン32は、復号を中止し、復号同期を回復し、又は次の識別可能なNALヘッダ、即ち、図6に示す例ではNALヘッダ90Cを探すことによってフレーム88Aの復号を再同期させることができる。復号同期を回復した後、復号エンジン32は、符号化ビデオデータの次の利用可能なスライス、例えば、NALヘッダ90C及びMB番号55～72によって識別されたNALユニット又はスライスにアクセスする。上述のように、復号エンジン32は、セグメントの開始に対応するMB番号、例えば、MB番号55にアクセスすることによって、符号化ビデオデータのこの次の利用可能なスライスの開始を決定することができる。復号エンジン32は、このMB番号をエラー検出モジュール40に転送する。

10

【0116】

第2のセグメントの開始、例えば、MB番号55を受信したことに応答して、エラー検出モジュール40はエラーフラグ46の状態を決定する。エラーフラグ46が設定されていない場合、エラー検出モジュール40はMB番号をエラーマッピングモジュール42にパスし、エラーマッピングモジュール42は、次の利用可能なスライスのカレントスライスとしての復号をサポートするために、MB番号55をマッピング48の開始MB48Aに記憶する。代替的に、エラーフラグ46が設定されている場合、エラー検出モジュール40は、MB番号55-1をエラーマッピングモジュール42にパスし、MB番号54をマッピング48の終了MB48Bとして記憶するようにエラーマッピングモジュール42に命令する。次いで、エラーマッピングモジュール42は、復号エラーの範囲を識別するためにマッピング48をエラー隠蔽モジュール36に転送する。エラー範囲は、スライスヘッダ90A、90Bに関連するスライスを含んでいる破損データセグメントの範囲を規定する。図6の例では、上述のように、エラーマッピングモジュール42は開始MB48Aと終了MB48Bとを転送し、隠蔽エンジン50は、開始MB48Aと終了MB48B(MB0～MB54)とによって識別されたMBを再構成バッファ38中に隠蔽する。

20

30

【0117】

復号エンジン32は、NALヘッダ90Dによって識別された次の利用可能なスライス又はNALユニットにアクセスすることによって、フレーム88Aを復号し続けることができる。フレーム88Aが復号されると、復号エンジン32はフレーム88Bを更に復号することができる。図6の例ではフレーム88B中に復号エラーが存在しないので、エラー処理モジュール34はエラーを検出せず、エラー隠蔽モジュール36はこれ以上エラーを隠蔽しない。このようにして、ビデオデコーダ26は各スライスを順に復号することができ、従って、ルックアヘッド動作をサポートするためにカレントスライスをバッファする必要がないので、ビデオデコーダ26はより小さいバッファ32を含むことができる。更に、ビデオデコーダ26は、通常の復号以外の追加の計算がほとんど必要とされないほど電力を消費しない。言い換えれば、連続的にではなく、復号エラーが検出されたときにエラー処理モジュール34が起動され、それによって計算を節約し、電力消費を低減する。

40

【0118】

様々な態様では、連続エラー処理技法は、特に2つ以上のプロセッサによって実装されたときに、上述の復号効率を維持し、並びにビデオデコーダ26のアーキテクチャを単純化することができる。即ち、例えば、1つのプロセッサが復号エンジン32を実装し、別

50

のプロセッサがエラー処理モジュール34を実装する。2つのプロセッサ間の通信が、アーキテクチャを複雑にすることがある。これらの通信を制限することによって、本技法は実装の複雑さを低減することができる。エラーが起きたときのみ通信が発生するので、本技法は、プロセッサ間の通信の数を制限し、それによってエラー処理アーキテクチャの複雑さを低減することができる。アーキテクチャに関して、スライスごとの順序で復号が実行されるので、ソフトウェア又はハードウェアは、カレントスライス及び次の利用可能なスライスのために2倍のバッファを維持し、そのようなバッファ間で交互に切り替える必要がない。更に、エラー隠蔽が必要とされる場合、前のスライスのための復号ステータス情報を記憶する必要がない。

【0119】

10

図7は、2つ以上のフレームにわたる復号エラーのためのエラー処理及び隠蔽を実行する際の、ビデオデコーダ26などのビデオデコーダの例示的な動作を示すフローチャートである。上述のように、ビデオデコーダ26は、複数の符号化ビデオフレームを備えることができる符号化ビデオフレームデータを受信する(92)。ビデオデコーダ26の復号エンジン32は、フレームのカレントスライスに最初にアクセスするときに、エラー処理モジュール34にカレントスライスの開始MB番号と開始フレーム番号とを決定させる(94)。スライスの開始MB番号は、復号エラーが検出されることがあるセグメントの開始を表す。開始フレーム番号は、カレントスライスが存在するフレームの番号を示す。開始MB番号及び開始フレーム番号は、カレントスライスに関連するNALヘッダ中に含まれ得る1つ以上のシンタックス要素から決定できる。

20

【0120】

開始MB番号と開始フレーム番号とを決定すると、復号エンジン32は、カレントスライスを復号し(96)、カレントスライスからの復号MBを、エラー検出のためにエラー処理モジュール34に出力する。エラー処理モジュール34のエラー検出モジュール40は、例えば、シンタックスルール44A、セマンティックルール44B又はエントロピールール44Cに従ってシンタックスエラー検査、セマンティックエラー検査及び/又はエントロピー符号化エラー検査を実行することによって、復号エラーが発生したかどうかを決定する(98)。復号エラーが検出されず(98)、スライスの終了に達していない(100)場合、復号エンジン32はカレントスライスを復号し続ける(96)。復号エラーが検出されない(98)が、カレントスライスの終了に達した(100)場合、カレントスライスは正常に復号された。この場合、復号エンジン32は、次の利用可能なスライスをカレントスライスとして復号することに進む。カレントスライスがビデオシーケンス中の第1のスライスであった場合、次のスライスは第2のスライスである。代替的に、カレントスライスが第1のスライスでなかった場合、次のスライスはビデオシーケンスにおける後のスライスである。後の、次の利用可能なスライスは、カレントスライスと同じフレーム中に存在することも、異なるフレーム中に存在することもある。いずれの場合も、復号エンジン32が次の利用可能なスライスに進むとき、復号エンジン32は、次の利用可能なスライスの開始MB番号と開始フレーム番号とを決定する(103)。

30

【0121】

復号エラーが検出された(98)場合、エラー検出エンジン40は復号エンジン32に復号エラーについて通知し、エラーフラグ46を設定する(102)。復号エンジン34は、エラーを通知されたことに応答して、上述のように、次の利用可能な、又は識別可能なスライスに進むことによって、復号を再同期させる。この場合、復号エンジン32は、次の利用可能なスライスの開始MB番号と開始フレーム番号とを決定する(103)。復号エンジン96は、後で説明するように、次の利用可能なスライスをカレントスライスとして復号することに進む(96)。復号エンジン32は、前のスライスが正常に復号されたときに、又は前のスライス中の復号エラーの検出に続く再同期の一部としてのいずれかで、次の利用可能なスライスに到達する(103)。各場合において、エラー処理モジュール34は、エラーフラグが設定されているかどうかを決定する(104)。エラーフラグが設定されている(104)場合、復号エラーの検出(98)の結果として、次の利用

40

50

可能なスライスがアクセスされた。設定されていない場合、前のスライスの正常な復号及び前のスライスの終了の検出(100)の結果として、次の利用可能なスライスがアクセスされた。

【0122】

エラーフラグが設定されていない(104)場合、復号エンジン32は、次の利用可能なスライスをカレントスライスとして復号することに進む(96)。言い換えれば、復号エンジン32によって得られた次の利用可能なスライスがカレントスライスになり、プロセスが繰り返される。しかしながら、エラーフラグが設定されている(104)場合、エラー処理モジュール34は、隠蔽される必要がある破損データセグメントの範囲を決定する。例えば、エラー検出モジュール40は、次の利用可能なスライスの開始MB番号に基づいて終了MB番号を決定し(106)、次の利用可能なスライスの開始フレーム番号に基づいて終了フレーム番号を決定する(108)。復号エラーが単一のスライスのみに関与する場合、終了MB番号及び終了フレーム番号はカレントスライスの終了MB番号及び終了フレーム番号とすることができます。しかしながら、復号エラーによって複数のスライスが影響を受けた場合、終了MB番号及び終了フレーム番号は、カレントスライスと次の利用可能なスライスとの間に存在し、マルチスライス破損データセグメントの一部を形成する、別の後のスライスに対応する。いずれの場合も、エラーマッピングモジュール42は、次の利用可能なスライスのそれぞれ開始MB番号及び開始フレーム番号に基づいて終了MB番号及び終了フレーム番号を決定することができます。この場合も、終了MB番号は次のスライス-1の開始MB番号とすることができます。次いで、エラー隠蔽モジュール36は、カレントスライスの開始MBから破損データセグメントの終了MBまでのMBを隠蔽することによってエラーを隠蔽する(110)。隠蔽に続き、復号エンジン32は、終了MB及び終了フレームを生成するために使用された、次の利用可能なスライスを、新しいカレントスライスとして復号することに進む(96)。

10

20

30

【0123】

エラーマッピングモジュール42は、終了MB番号及び終了フレーム番号を、終了MB48B及び終了フレーム48Dとしてマッピング48に記憶する。次いで、エラーマッピングモジュール48は、開始MB48A、終了MB48B、開始フレーム48C、及び終了フレーム48Dに基づいてエラー範囲を決定し、この範囲をエラー隠蔽モジュール36の隠蔽エンジン50に転送する。上述のように、隠蔽エンジン40は、様々な隠蔽技法のいずれかに従って、エラー範囲、例えば、開始MB48A及び終了MB48B並びに開始フレーム48C及び終了フレーム48Dに基づいて検出された復号エラーを隠蔽する(110)。動作中、エラーマッピングモジュール48は、開始MB番号及び終了MB番号だけでなくフレーム番号を考慮に入れることによって、エラー範囲を決定することができます。復号エラーがフレーム境界線を超える場合、終了MB番号は、実際に、隠蔽すべきセグメントの開始MB番号よりも小さくなることがある。フレーム番号を追跡することにより、このタイプの状況においてエラー範囲を適切に解決することが可能になる。

【0124】

図8は、2つ以上のフレームにわたるエラーを含んでいる例示的なビットストリームへのエラー処理及び隠蔽技法の適用を示す図である。図8は、図6と同様であるが、復号エラーが2つの連続するフレームにわたり、それによりエラーを検出した後の再同期のための次の利用可能なスライスが、カレントスライスと異なるフレームに存在するシナリオを示す。図8に示すように、フレーム番号を追跡することによって、2つのフレームにわたるデータが損失している場合に、連続エラー処理技法を適用することができる。従来のシステムでは、エラーが2つのフレームにわたるとき、2番目のフレームが完全に廃棄されることがある。しかしながら、連続エラー処理技法の場合、デコーダは、スライスヘッダのフレーム番号を検査して、カレントフレームを終える前に新しいフレームが開始しているかどうかを決定することができる。開始している場合は、デコーダは、カレントフレームの残りのMBを隠蔽し、新しいフレームを処理する。第2のフレームの最初の幾つかのMBを次のスライスヘッダまで隠蔽することができる。ただし、第2のフレーム中の残り

40

50

のMBは保持できる。

【0125】

エラー隠蔽のための第1のフレームのMB73～98と第2のフレームのMB0～76とを識別するために、連続エラー処理技法を図8のシナリオに適用することができる。図8は、2つのフレーム116A、116B（「フレーム116」）を含むビットストリーム114を示し、フレーム116Aは、それぞれのスライスを指定する複数のNALヘッダ118A～118Dを備え、フレーム116Bは、それぞれのスライスを指定する複数のNALヘッダ118E～118Hを備える。NALヘッダ118Aと118Eとは、それぞれのフレーム116の開始を示すことを表すために、それぞれ影付きで示される。NALヘッダ118A～118H（「NALヘッダ118」）の各々は、スライスを含むことができるNALユニットの開始を示す。10

【0126】

各NALユニットは、NALヘッダ118の後続のヘッダの開始時に終了する。例えば、第1のNALユニットは、NALヘッダ118Aと、MB番号0～20に対応するMBを含んでいるスライスとを備えることができ、MB番号20は後続のNALヘッダ118Bの前の最後のMBである。フレーム116の各々の上部にMB番号を示す。スライスを含むことができるNALユニットは、独立して復号できる、即ち、他のNALユニットからの情報なしに復号できる。このようにして、各NALユニットは復号可能MBの自蔵式シーケンスであるので、NALユニットの損失は、先行する又は後続のNALユニットの復号を妨げない。20

【0127】

図2のビデオデコーダ26など、ビデオデコーダは、これらのフレーム116を受信し、本開示で説明する技法に従って非ルックアヘッド連続エラー処理動作を実行することができる。復号エンジン32は、フレーム116Aの第1のNALユニット又はスライス、即ち、NALヘッダ118A及びMB0～20にアクセスし、最初のMB番号、例えば、MB0、並びにフレーム番号を決定することによって、第1のセグメントの開始を決定することができる。復号エンジン32は、このMB番号0を、図3のエラー処理モジュール40など、エラー処理モジュール34のエラー検出モジュールに転送することができる。エラー検出モジュール40は、MB番号0及びフレーム番号をエラーマッピングモジュール42に転送し、エラーマッピングモジュール42は、MB番号0及びフレーム番号を開始MB番号48A及び開始フレーム番号48Cとしてマッピング48に記憶する。この例では、開始MB番号はMB0であり、開始フレーム番号はフレーム0であるとすることができる。30

【0128】

復号エンジン32は、符号化MB0～20を復号し、この復号されたビデオデータ（例えば、復号MB）を再構成バッファ38中に記憶する。復号エンジン32はまた、復号MB0～20をエラー検出モジュール40に与えることができ、エラー検出モジュール40は、エラー検査の適用によって復号エラーが発生したかどうかを決定することができる。エラーが検出されなかった場合、ビデオデコーダ32は、例えば、NALヘッダ118B及びMB番号21～55によって指示された次のスライスにアクセスし、MB開始番号及びフレーム番号をエラー検出モジュール40にパスし、このスライスを復号する。エラー検出モジュール40はエラーフラグ46からステータスを決定する。この例では、前のスライス（MB0～20）における復号エラーが検出されなかつたので、エラーフラグ46は設定されなかつた。エラーマッピングモジュール42は、この第2のスライス（MB21～54）の開始MB番号及びフレーム番号を受信し、開始MB番号及び開始フレーム番号を開始MB48A及び開始フレーム48Cとしてマッピング48に記憶する。復号エンジン32、エラー検出モジュール40、及びエラーマッピングモジュール42は、このようにして、エラーが検出されるまで、後続のスライスのために動作し続けることができる。40

【0129】

10

20

30

40

50

図8の例では、第1のフレーム116Aの最後のスライス(MB73~98)の一部、フレーム116BのNALヘッダ118E、第2のフレーム116Bの第1のスライス(MB0~45)のすべて、フレーム116BのNALヘッダ118F、及びフレーム116Bの第2のスライス(MB45~77)の一部は、損失又は破損している(「損失」)。このエラーにより前に、エラー検出モジュール42は、開始MB番号73と第1のフレーム116Aに対応するフレーム番号とを受信することができる。上述のように、エラーマッピングモジュール42は、このMB番号73及びフレーム番号を、開始MB48A及び開始フレーム48Cとしてマッピング48に記憶することができる。復号エンジン32によって復号する過程で、エラー検出モジュール40は、この復号エラーを検出し、エラーフラグ46を設定することができる。エラー検出モジュール40はまた、復号エンジン32にエラーについて通知することができ、その結果、復号エンジン32は、上述の方法で復号同期を回復する。特に、復号エンジン32は、次のNALユニットヘッダを探すことによって、次の利用可能なスライスにアクセスすることができる。

10

【0130】

図8の例では、復号エンジン32は、NALヘッダ118G及びMB番号77~88を有する次の利用可能なNALユニットを識別する。MB77~88を備えるNALユニットが、ビットストリームを解析することによってNALヘッダを容易に見分けることができる次のNALユニットであるという点で、NALヘッダ118Gは次の利用可能なNALユニットを示す。上述のように、復号エンジン32は、次の利用可能なスライスの開始MB番号、例えば、NALヘッダ118Gによって示されるスライスのMB番号77と、他の符号化ビデオデータに対するこのセグメントの位置、例えば、フレーム番号(1)とを決定することができる。復号エンジン32は、開始MB番号77と、フレーム116Bに対応するフレーム番号(1)とをエラー検出モジュール40に与えることができる。

20

【0131】

エラー検出モジュール40は、MB番号77と、フレーム116Bに対応するフレーム番号1を受信すると、エラーフラグ46のステータスを決定する。例示のために、エラー検出モジュール40がエラーを以前に検出し、エラーフラグ46が設定されていると仮定すると、エラー検出モジュール40は、MB77~88を備えるスライスの開始MB番号77に基づいて前のセグメントの終了を決定する。この例では、エラー検出モジュール40は、MB番号77から1を減算してMB番号76を得ることによって、破損データセグメントの終了を計算する。エラー検出モジュール40は、終了MB番号76及びフレーム番号1をエラーマッピングモジュール42に転送し、エラーマッピングモジュール42はこれらの番号を、それぞれ、終了MB48B及び終了フレーム48Dとしてマッピング48に記憶する。記憶されると、エラーマッピングモジュール42は、「エラー範囲」と呼ばれるマッピング48をエラー隠蔽モジュール36に転送する。

30

【0132】

隠蔽エンジン50は、開始フレーム48C及び終了フレーム48Dから、復号エラーが2つのフレームにわたり、開始MB48A及び終了MB48Bに基づいて、MBがフレーム116Aの終了及びフレーム116Bの開始から損失していると決定することができる。隠蔽エンジン50はまた、復号エンジン32、エラー検出モジュール40、又はエラーマッピングモジュール42のいずれかから最大フレーム長を受信することができ、この例では98MBとして図8に示す。即ち、最大フレーム長は、フレーム116の1つの最後のMB番号として定義できる。最大フレーム長98MBは、quarter common intermediate format (QCIF) フレームにおけるMBの番号に対応する。隠蔽エンジン50は、フレーム番号の差を計算するために、開始フレーム48Cとして記憶された開始フレーム番号から、終了フレーム48Dとして記憶された終了フレーム番号を減算する。

40

【0133】

計算されたフレーム番号差が0、即ち、開始及び終了が同じフレーム番号内にある場合、隠蔽エンジン50は開始MB番号、例えば、MB番号73から、次の利用可能なスライスの開始MB番号の直前の終了MB番号までMBを交換、又は隠蔽することができる。こ

50

の場合も、隠蔽エンジン 50 は、再構成バッファ 38 中の MB を直接隠蔽するか、又は再構成バッファに与えられた復号ビデオ MB を隠蔽する際に使用するために、隠蔽される MB を復号エンジン 32 に与えることができる。図 8 を参照すると、計算されたフレーム番号差が 1 よりも小さい又は 1 に等しい、即ち、復号エラーがフレーム 0 (116A) 及びフレーム 1 (116B) の部分にわたる場合、隠蔽エンジン 50 は、開始 MB 48A として記憶された開始 MB 番号、例えば、MB 番号 73 に対応する MB を、最大フレーム長、例えば、MB 番号 98 に対応する MB まで、交換又は隠蔽することができる。このようにして、エラーが 2 つのフレームにわたるとき、エラーを含んだ第 1 のスライスの開始からフレームの終わりまで第 1 のフレームの残りが隠蔽される。

【0134】

10

以下で説明するように、フレーム差が 1 よりも大きい場合、隠蔽エンジン 50 は追加のフレームを挿入し、エラーが発生し第 1 のフレームと、同じエラーが発生した最後のフレームとの間の中間フレームの各々のための MB 番号 0 ~ 98 に対応する MB を隠蔽することができる。この場合、開始フレーム番号と終了フレーム番号との間の 2 以上のフレーム差は、少なくとも 1 つのフレーム全体が損失したことを示す。フレーム差 3 は 2 つのフレーム全体が損失したことを示し、フレーム差 4 は 3 つのフレーム全体が損失したことを示し、以下同様に示す。最後のフレームにおいて、隠蔽エンジン 50 は、MB 番号 0 から終了 MB 48B として記憶された MB 番号、例えば 76 までに対応する MB を隠蔽することができる。

【0135】

20

図 8 に示すインスタンスでは、復号エラーが、連続するフレーム 116A 及び 116B にわたるので、隠蔽エンジン 50 は、差が 1 に等しいと決定し、開始 MB 48A として記憶された MB 番号、例えば 73 から最大フレーム長、MB 番号 98 までに対応する MB を隠蔽する。差が 2 以上ではないので、隠蔽エンジン 50 は復号エラーが及ぶ最後のフレームに達し、MB 番号 0 から終了 MB 48A に記憶された MB 番号、例えば 76 までに対応する MB を隠蔽する。要約すると、フレーム 1 の MB 73 ~ 98 及び MB 0 ~ 76 が隠蔽される。フレーム差が 2 よりも大きい又は 2 に等しい場合、単に損失した参照フレームを廃棄する代わりに、隠蔽エンジン 50 は、損失したフレームの数に等しい数のフレームを挿入し、それによって、参照フレーム不整合を回避する。従来のビデオデコーダにおいて一般的であるように、なくなったフレームを廃棄しないことによって、本技法は、図 9A 、図 9B 及び図 9C に関して以下で説明するように、復号パフォーマンスを改善することができる。

30

【0136】

図 9A 、図 9B 及び図 9C は、複数参照フレームを可能にするビットストリームに対するエラー処理及び隠蔽技法の適用を示す図である。図 9A は、「I」、「P₀」、「P₁」、「P₂」、「P₃」及び「P₄」と標示された複数のフレームを含むビットストリーム 120 を示す。I フレームは、H.264 / MPEG 4 AVC イントラフレームとして知られているフレームを指し、P フレームは、H.264 / MPEG 4 AVC インターホードフレームを指す。一般に、各 P フレームが I フレーム中に存在する少なくとも一部のビデオデータを参照するように、P フレームを符号化することができるという点で、I フレームは、1 つ以上の P フレームのための参照を形成することができる。従って、I フレームが前又は後続のフレームに依存しないという点で、I フレームは、インター予測に依拠せず、復号を再同期させるために使用できる。従って、図 9 には示されていないが、ビデオデコーダ 26 が、NAL ヘッダによって識別される次の利用可能なスライスを取り出し、以前に損失又は破損した MB によるエラーを繰り越すことなしに I フレームを復号することができるように、I フレームは NAL ヘッダの後に来ることができる。更に、P フレームは他の P フレームを参照することができる。

40

【0137】

H.264 プロセスでは、複数参照フレームを使用することは、符号化効率を改善するための一般的な特徴である。H.264 プロセスは、前のフレームのみを参照フレームと

50

して使用することを制限せず、動き補償予測が16個の過去のフレームほど離れた参照フレームに基づくことを可能にする。この特徴は、符号化効率を改善するだけでなく、エラー伝搬を軽減するのを助けることができ、通常、エンコーダにおけるエラー防止ツールとして挙げられる。

【0138】

フレームが完全に損失したとき、デコーダは、損失したフレームがあることを知る必要なしに次のフレームを復号し続けることができる。複雑なエラー隠蔽方法を使用して損失したフレーム全体を推定しない限り、デコーダは一般により良い結果を与えることができないので、前のフレームのみを参照フレームとして使用するとき、この手法は効果的である。ただし、(H.264の場合のように)複数参照フレームを使用するとき、フレーム損失により、正しく受信されたフレームが、予測のために間違ったフレームを参照する可能性が高くなる。1つ以上の損失したフレームによって引き起こされる間違った参照を、本開示では参照フレーム不整合と呼ぶ。10

【0139】

各フレームの上部の矢印は、複数のPフレーム、例えば、P₀、P₁、P₂、P₃及びP₄、並びにIフレームの間の参照フレーム依存関係を示す。ビットストリーム中のシンタックス要素は、例えば、フレーム番号又はフレーム番号オフセット値に基づいて、受信したPフレームごとに参照フレームを識別する参照フレームインデックス情報を含むことができる。図8Bに示すように、フレームP₀及びP₁は、インター予測のための参照フレームとしてのIフレームに依存し、フレームP₂及びP₄は、フレームP₁に依存し、フレームP₃は、フレームP₂に依存する。図1のビデオエンコーダ20など、ビデオエンコーダは、上述のように、ビットストリームを圧縮し、送信効率を改善するために、上記の依存関係によって示された方法で、ビットストリーム120を符号化することができる。20

【0140】

また上述したように、ビットストリーム120が、送信チャネル16を介した受信機24への送信中に損失又は破損し、ビデオデコーダ26は、1つ以上のフレームが損失又は破損しているビットストリーム122を受信することになる場合がある。図9Aに示すように、ビットストリーム122は、Iフレーム、並びにフレームP₀、P₁及びP₄をのみ含む。この例では、フレームP₂及びP₃が、送信中に損失又は破損した。従って、連続的に受信したフレームP₁とP₄との間のフレーム番号差は3であり(4-1)、2つのフレーム全体の損失を示す。連続的に受信したフレーム間のフレーム番号差が2の場合、1つのフレーム全体の損失が示される。30

【0141】

送信中に、単一のフレームが損失することがある。代替的に、複数のフレームが損失することがある。各場合において、参照フレーム不整合は、視覚的品質を劣化させ、エラーの伝搬を許す。複数のフレームが同じパケット内に配置される、ブロードキャストなどの適用例において複数のフレームの損失が起こることがある。従って、パケットが損失すると、複数のフレームが損失する。ビットストリーム124Bのフレームの上部の矢印は、エラー隠蔽が実行されない場合に起こりうる考えられる依存関係を示す。特に、矢印は、フレームP₄が、フレームP₁の代わりに今度は不適切にIフレームに依存する、変化した依存を示す。40

【0142】

図9Aの例では、適用可能な参照フレームインデックス情報によって示されるように、フレームP₄が3つ先行するフレームに依存することが意図されると仮定する。しかしながら、損失したフレームがあると、フレームP₄は、フレームP₁ではなく、2つの損失したフレームをカウントせず3つ先行するフレーム(即ち、Iフレーム)に依存する。言い換えれば、P₄は、依然として3フレーム戻ってポイントするが、当初意図された参照フレームP₂及びP₃がビットストリームから損失しているとき、Iフレームにポイントする。この参照フレーム不整合は、主に損失したフレームによるものである。復号プロセ50

スはビットストリームにおいて実際に受信したフレームデータに基づいてビデオフレームを生成するにすぎないので、デコーダはフレーム P₂ 及び P₃ の損失に気づいていないことがある。

【 0 1 4 3 】

フレーム P₁ を復号した後、デコーダは、シーケンスにおける次のフレームとしてフレーム P₄ を受け取り、ビットストリーム中の参照フレームインデックス（3フレーム前）に従って、その参照フレームを、 P₁ フレームではなく I フレームとして識別することができる。従って、フレーム P₂ 及び P₃ の損失は、フレーム P₄ の参照フレーム不整合を引き起こす。この参照フレーム不整合は、ビデオ再生中にフレーム P₄ の視覚的品質を劣化させ、場合によっては生じたエラーを後続のビデオフレームに伝搬することができる。10

【 0 1 4 4 】

本開示で説明する技法によれば、上述のように復号エラーを識別し、損失したフレームを交換してビットストリーム 124 を生成するようにビデオデコーダ 26 を構成することができる。このようにして、1つ以上の介在するフレームが送信中に損失又は破損したとき、ビデオデコーダ 26 は、2つのフレーム間の間隙を識別及び補償し、それによって、参考フレーム不整合を訂正することができる。動作中、図 2 の復号エンジン 32 は、ビットストリーム 122 を受信し、I フレームの各セグメントの開始を決定し、それらのセグメントを復号し、復号ビデオデータをエラー処理モジュール 32 に出力する。図 3 に示すエラー処理モジュール 32 のエラー検出モジュール 40 は、復号エラーが発生したかどうかを決定することに進む。このようにして、エラー検出モジュール 40 が損失したフレーム P₂ 及び P₃ を検出するまで、復号エンジン 32 及びエラー処理モジュール 34 は、フレーム P₀ 及び P₁ を復号し続ける。20

【 0 1 4 5 】

上述のように、これらのエラーを検出するより前に、エラー検出モジュール 40 は、（フレーム P₂ の開始時にエラーが発生したので）セグメントの開始、例えば、MB 番号 0 と、符号化ビデオデータにおけるそのセグメントの相対位置を識別するフレーム番号とを受信することができる。フレーム番号は、例えば、スライスヘッダ中のフレーム番号シンタックス要素にアクセスすることによって、H.264 又は同様の実装形態で得られる。連続するフレーム番号の比較は、介在フレームが損失したかどうかを明らかにすることができます。フレーム番号が使用されている場合、ビデオデコーダは単に連続するフレーム番号を追跡し、フレーム参照差を決定することができる。フレーム番号差が 1 よりも大きい場合、少なくとも 1 つのフレーム全体がなくなっている。なくなっているフレームの数は、フレーム番号差を一般に追跡しなければならない。30

【 0 1 4 6 】

幾つかの例では、フレーム番号が破損された場合、エラー検出モジュール 40 は、開始フレーム 48C 又は終了フレーム 48D など、前に記憶された開始フレーム番号又は終了フレーム番号に 1 を加算することによって開始フレーム番号を決定する。エラーマッピングモジュール 42 は、開始 MB 番号及び開始フレーム番号を、開始 MB 48A 及び開始フレーム 48C としてマッピング 48 に記憶する。エラー検出モジュール 40 は、復号エンジン 32 にエラーについて通知し、エラーフラグ 46 を設定する。40

【 0 1 4 7 】

エラー検出モジュール 40 は、代替又は追加として、リアルタイムプロトコル（RTP）タイミング情報など、エラーを検出する外部機構を採用することができる。RTP タイミング情報の場合、エラー検出モジュール 40 は、連続的に受信したフレーム間の時間差を検査し、一定であると仮定されるフレーム間隔によって時間差を分割することによって、いくつのフレームが損失したか直接決定することができる。一定のフレーム間隔は、ブロードキャスト適用例など、幾つかの適用例で見られる。しかしながら、ビデオ電話など、フレーム間隔が通常一定でない適用例の場合、エラーを決定するフレーム番号方法を採用することができる。

【 0 1 4 8 】

10

20

30

40

50

復号エンジン 3 2 は、次の利用可能な NAL ヘッダにアクセスすることによって、エラーに応答して復号を再同期させることができる。この例では、復号エンジン 3 2 は、フレーム P₄ の第 1 の NAL ヘッダにアクセスし、フレーム P₄ のこのセグメントの開始、例えば開始 MB 番号と、フレーム P₄ に対応するフレーム番号とをエラー検出モジュール 40 にバスする。エラー検出モジュール 40 は、エラーフラグ 46 の状態を決定し、エラーフラグが設定されていることがわかると、受信した MB 番号とフレーム P₄ に対応するフレーム番号とに基づいて前のセグメントの終了を決定する。

【 0 1 4 9 】

例えば、復号エンジン 3 2 が、フレーム P₄ を復号したときに、開始 MB 番号が 0 であり、開始フレーム番号が 5 であると決定したと仮定すると、エラー検出モジュール 40 は、（エラー検出モジュール 40 が最大フレーム長 98 MB について通知されたと仮定して）前のフレームの終了 MB 番号 98、及び終了フレーム番号 4 を計算することができる。エラーマッピングモジュール 42 は、これらの終了 MB 番号及び終了フレーム番号を、終了 MB 48 B 及び終了フレーム 48 D としてマッピング 48 に記憶する。従って、エラー検出モジュール 40 は復号エラーの終了を検出することができず、マッピング 48 が完全に指定されるまで、エラーマッピングモジュール 42 はマッピング 48 を隠蔽エンジン 50 に転送することができないので、フレーム P₄ のための第 1 のスライスヘッダが復号されるまで、エラー隠蔽モジュール 36 はエラーを隠蔽することができない。

【 0 1 5 0 】

次いで、エラーマッピングモジュール 42 は、マッピング 48 をエラー隠蔽モジュール 36 に転送する。エラー隠蔽モジュール 36 は、マッピング 48 を評価し、開始フレーム番号、例えば、2 から終了フレーム番号 4 を減算することに基づいて、2 つのフレームが損失したと決定することができる。更に、開始 MB 番号が 0 であり、終了 MB 番号が 98 であることを考えて、エラー隠蔽モジュール 36 は更に、2 つの完全なフレームが損失又は破損したと決定することができる。従って、図 9B を参照すると、2 つの損失したフレームの外側のスライスにおいて MB 隠蔽を実行することに加えて、隠蔽エンジン 50 は、例えば、直接、又は、復号エンジン 3 2 とともに、2 つの完全なフレームをビットストリーム 122 に挿入して、ビットストリーム 124B を再構成バッファ 38 中に生成することができる。

【 0 1 5 1 】

図 9B に示すように、ビットストリーム 124B は、前に損失したフレームの代わりに 2 つの追加のフレームを含む。フレーム P₄ のビデオデータが受信され、復号された後まで、このフレーム挿入技法は実行されない。これらの追加のフレームは、図 9B のビットストリーム 124B において「P₁」として両方とも示されて、隠蔽エンジン 50 がフレーム P₁ を 2 回繰り返すことによって 2 つの損失したフレームを隠蔽し、フレーム P₄ より前にこれらの 2 つの繰返しフレームを再構成バッファ 38 に挿入することを表す。損失したフレーム P₂ 及び P₃ の代わりに、繰り返された P₁ フレームがある場合、フレーム P₄ は依然として、ビットストリームにおいて 3 フレーム後のフレームを参照するが、この場合、そのフレームは、I フレームではなく、より適切に繰り返されたフレーム P₁ である。

【 0 1 5 2 】

代替として、隠蔽エンジン 50 は、フレームを交換せず、代わりに、フレーム P₄ など、フレームの 1 つの NAL ヘッダ内に符号化された参照フレーム依存関係を修正することができる。例えば、隠蔽エンジン 50 は、マッピング 48 を受信し、フレーム P₄ にアクセスするための 1 つの終了フレーム番号 48D を再構成バッファ 38 に追加する。これに基づいて、フレーム P₄ が正しいフレームを参照するように、隠蔽エンジン 50 はビットストリームの参照インデックスを修正することができる。例えば、フレーム P₄ が、ビットストリーム 122 に示すように I フレームではなくフレーム P₁ に依存するように、隠蔽エンジン 50 は次に、フレーム P₄ の NAL ヘッダにアクセスし、依存シンタックス要素を修正する。新しいフレームを参照フレームバッファに単に挿入することがより容易な

10

20

30

40

50

ことがある。いずれの場合も、ここで、フレーム P₄ は予測符号化のためのその参照としてフレーム P₁ を使用して復号及び再構成でき、それによって、復号エラーがフレーム P₄ 及び他の後続のフレームに伝搬することを防ぎ、フレーム P₄ の正確な再構成を実行する。

【 0 1 5 3 】

ビットストリーム 1 2 4 B は、便利な隠蔽機構として、繰返しフレームを表すが、隠蔽エンジン 5 0 は、損失したフレームを隠蔽又は交換するために、他の隠蔽技法を実行することができる。従って、本開示で説明する効率的なエラー処理技法は、この点について厳密に限定されるべきではない。例えば、フレーム P₁ ~ P₄ を閲覧するときに、より滑らかな動きを生成するために、隠蔽エンジン 5 0 がより複雑な動き補償フレーム補間方式を適用するように、エラー処理技法を実装することができる。従って、フレーム P₁ を単に 2 回繰り返すことによって 2 つの損失したフレーム P₂ 及び P₃ を再現する、又は、代替的に、補間又は他のより複雑な技法を使用して、損失したフレーム P₂ 及び P₃ の近似を再現することができる。

【 0 1 5 4 】

例えば、隠蔽エンジン 5 0 は 2 つのフレームを挿入し、それによって、フレーム P₄ がフレーム P₁ に正しく依存するように依存関係を訂正し、フレーム P₄ を復号する。幾つかの態様では、隠蔽エンジン 5 0 は、再構成バッファ 3 8 中のフレーム P₁ 及び P₄ にアクセスし、上述のように、フレーム補間方式を使用して決定されたフレーム P₁ と P₄ との間の動きに基づいて、繰り返されたフレーム P₁ を推定することができる。このようにして繰り返されたフレーム P₁ を推定することによって、隠蔽エンジン 5 0 は、再生中に閲覧されたときの動きを平滑化する近似フレーム P₂ 及び P₃ を生成することができる。いずれの場合も、参照フレーム不整合を訂正することによって、この隠蔽技法は、復号プロセスを実質的に修正することなしに、フレーム P₄ がフレーム P₁ をその予測参照フレームとして使用することを可能にし、フレーム P₄ が正確に再構成されることを可能にすることができる。

【 0 1 5 5 】

図 9 C は概して、図 9 A 及び図 9 B で概説したシナリオに対応するが、2 つの連続するフレーム P₂ 及び P₃ ではなく、単一のフレーム P₂ の損失を示す。この場合、フレーム不整合は 1 つのみであるが、フレーム P₃ 及び P₄ は依然として間違ったフレームを参照することになる。図 9 C は、元の符号化ビットストリーム 1 2 0 と、フレーム P₂ 全体が損失している受信したビットストリーム 1 2 3 とを示す。この場合、図 9 C のビットストリーム 1 2 5 に示すように、参照フレーム不整合をなくすために、隠蔽エンジン 5 0 は、フレーム P₂ の代わりにフレーム P₁ を繰り返す。従って、フレーム P₄ は正しいフレームを参照することができる。この場合も、R T P 情報を使用して、フレーム番号を比較することによって、又は他の技法によって、参照フレーム不整合を検出することができる。更に、フレーム P₂ を交換することは、フレーム P₁ などフレームを単に繰り返す、又はフレーム P₂ を交換するためにフレームを補間することによって達成できる。

【 0 1 5 6 】

図 1 0 は、複数参照フレームを可能にするビットストリーム中の損失したフレームに対してエラー処理及び隠蔽を実行する際のビデオデコーダの例示的な動作を示すフローチャートである。図 1 0 のフローチャートは、図 7 のフローチャートに実質的に対応する。例えば、図 1 0 は、ビデオデコーダ 2 6 が符号化ビデオフレームデータを受信し(9 2)、復号されているカレントスライスの開始 M B 番号と開始フレーム番号とを決定する(9 4)プロセスを示す。カレントスライスを復号する(9 6)過程で、復号エラーが検出された(9 8)場合、エラー検出モジュール 4 0 はエラーフラグを設定する(1 0 2)。復号エラーがなく(9 8)、スライスの終了にまだ達していない(1 0 0)場合、ビデオデコーダ 2 6 はカレントスライスを復号し続ける。スライスの終了に達した(1 0 0)、又はエラーが検出され(9 8)、エラーフラグが設定された(1 0 2)場合、ビデオデコーダ 2 6 は次の利用可能なスライスに進み、その開始 M B 番号と開始フレーム番号とを決定す

る（103）。

【0157】

エラーフラグが設定されている（104）場合、エラーマッピングモジュール42は、次の利用可能なスライスの開始MB番号に基づいて、損失又は破損したデータセグメントの終了MB番号を決定し（106）、次の利用可能なスライスの開始フレーム番号に基づいて終了フレーム番号を決定する（108）。更に、1つ以上のフレームが損失しているとき、繰り返されたフレームの挿入を可能にするために、開始フレーム番号と終了フレーム番号との間の差が1よりも大きいかどうかを決定する（126）ように、エラーマッピングモジュール42を構成することができる。1よりも大きい場合、ビットストリームから少なくとも1つのフレームがなくなっている。ビットストリームからなくなっているフレームの数は通常、フレーム番号差-1に等しい。従って、フレーム番号差が3である場合、一般に、ビットストリームからなくなった2つのフレームがある。

10

【0158】

フレーム番号差、即ち、開始フレーム番号と終了フレーム番号との間の差が1よりも大きいと決定すると、エラーマッピングモジュール42は、前のフレームを繰り返し、なくなっているフレームの代わりにそれを挿入する（128）ようにエラー隠蔽モジュール36に指示する。上述のように、前のフレームを繰り返す代わりに、動き補償補間など他の技法によって交換フレームを生成するように隠蔽モジュール36を構成することができる。代替として、フレームを交換する代わりに、例えば、その後のフレームが適切な参照フレームを使用するように、参照インデックスに損失したフレームの数（即ち、フレーム番号差-1）を追加することによって、1つ以上の後のフレームのための参照フレームインデックス情報を調整するように隠蔽モジュール36を構成することができる。

20

【0159】

繰り返された又は補間されたフレームと交換される損失したフレームの数は、フレーム差に基づく（128）。フレーム番号差が3である場合、例えば、前のフレームを1回、即ち、次の2つのフレームのために繰り返すことができる。フレーム番号差が2である場合、前のフレームを1回、即ち、次のフレームのために繰り返すことができる。従って、交換されるフレームの数は、フレーム番号差-1に等しいとすることができる。フレーム番号差が1よりも大きくない（126）、又は損失したフレームの十分な数が交換された（128）場合、例えば、図7を参照しながら説明したように、エラー隠蔽モジュール36は、開始MB番号及び終了MB番号並びに開始フレーム番号及び終了フレーム番号に基づいてエラーを隠蔽し（110）、次の利用可能なスライスをカレントスライスとして復号する（96）ことに先だって、エラーフラグをリセットする（110）。

30

【0160】

前のフレームを繰り返すこと、又は代替隠蔽機構を適用することの他に、エラー隠蔽モジュール36は、交換されなかったMBの隠蔽のための技法を適用すること（110）に進み、次いで、次の利用可能なスライスをカレントスライスとして復号すること（96）に進む。カレントスライスとして使用される次の利用可能なスライスは、例えば、開始MB番号及び開始フレーム番号を得る（103）スライスである。損失したフレームを繰り返すことに加えて、エラー隠蔽モジュール36は、カレントスライス及びフレーム中のMBを隠蔽することができる。

40

【0161】

開始フレーム番号及び終了フレーム番号から、エラーが2つ以上のフレームにわたるということがわかったので、エラー隠蔽モジュール36は、カレントフレームの開始MBから（98MBが適用可能な最大数であると仮定して）MB98までのMBを隠蔽することができる。更に、エラー隠蔽モジュール36は、最後の交換されたフレームに続くMBを次の利用可能なスライスのポイントまで、即ち、終了MB番号まで隠蔽することができる。従って、隠蔽は、第1のフレーム中のMBを隠蔽することと、1つ以上の損失したフレームのために第1のフレームを繰り返すことと、損失したフレームに続くフレーム中のMBを隠蔽することとの組合せを含むことができる。

50

【0162】

図11A及び図11Bは、複数参照フレームを可能にする例示的なビットストリームへのエラー処理及び隠蔽技法の適用を示す図である。図11Aの例では、ビットストリームは、最初のIフレームと4つの連続するPフレーム(P_0 、 P_1 、 P_2 、 P_3)とを含む。復号エラーが、フレーム P_0 の(復号エラーが開始するスライスの開始である)MB88で開始し、フレーム P_1 をわたって、フレーム P_2 のMB57まで続くMBの損失を生じる。この場合、開始MB番号及び開始フレーム番号は、それぞれMB88及びフレーム P_0 である。フレーム P_2 中の次の利用可能なスライスがMB75で開始する場合、終了MB番号及び終了フレーム番号は、それぞれMB74及び(フレーム P_2 の)3である。

【0163】

図11Aの例では、フレーム P_2 と P_0 との間のフレーム番号差は $3 - 1 = 2$ であり、1つのフレーム全体(フレーム P_1)の損失を示す。エラー隠蔽モジュール36は、フレーム P_0 のMB88~98を隠蔽し、フレーム P_0 を繰り返すことによってフレーム P_1 を隠蔽し、フレーム P_2 のMB0~74を隠蔽する。例えば、エラーマッピングモジュール34は、復号エラーがフレーム P_0 及び P_1 にわたることを示し、 P_0 を開始MB88からMB98におけるフレームの終了まで隠蔽すべきであることを示す。図10の動作110では、エラーマッピングモジュール42は、2つのエラー範囲、1つは、(フレーム P_0 を繰り返すことによって交換された)交換されたフレーム P_1 より前の P_0 中の損失したMB、もう1つは、交換されたフレーム P_1 の後のフレーム P_2 中の損失したMBをマッピングすることができる。第1のエラー範囲は、開始MBをフレーム P_0 のMB88として、及び終了MBをフレーム P_0 のMB98として指定する。次いで、エラー隠蔽モジュール36は、この第1のエラー範囲中のMBを隠蔽する。

【0164】

第2のエラー範囲は、開始MBを、交換されたフレーム P_1 の後の次のMB、即ち、フレーム P_2 中のMB0として指定し、既存の終了MB、即ち、フレーム P_2 中のMB74を保持することができる。次いで、エラー隠蔽モジュール36は、この第2のエラー範囲中のMBを隠蔽する。従って、 P_1 を隠蔽するために、得られたフレーム P_0 の繰返しがあれば、エラーマッピングモジュール34は、新しい開始MB番号及び開始フレーム番号を、それぞれMB0及びフレーム P_2 として計算し、前に計算された終了MB及び終了フレーム番号を使用することができる。これらの番号を使用して、エラー隠蔽モジュール36は、開始MB0において開始し、次の利用可能なスライスの開始MB-1として計算された、終了MB74まで続くフレーム P_2 中のMBを隠蔽することができる。

【0165】

フレームが繰り返されるとき、MBの隠蔽を処理するための他の技法を使用することができる。エラー処理モジュール34は、様々な方法で、フレーム P_1 中の損失したMBの隠蔽を処理することができる。従って、図11Aを参照しながら説明したプロセスは、例示のために与えられる。

【0166】

図11Bの例では、復号エラーは、フレーム P_0 の一部、フレーム P_1 及び P_2 のすべて、及びフレーム P_3 の一部にわたる。この場合、開始MBはMB88であり、開始フレームはフレーム P_0 である。次の利用可能なスライスがフレーム P_3 のMB53において開始すると仮定すると、終了MBはMB52である。終了フレームはフレーム P_3 である。この場合、終了フレーム P_3 及び開始フレーム P_0 のフレーム番号差は、 $3 - 0 = 3$ である。従って、ビットストリーム中に2つの損失したフレーム(P_1 及び P_2)がある。

【0167】

エラーマッピングモジュール42は、フレーム P_0 中のエラーが検出されたスライスの開始MB88からの第1のエラー範囲と、フレーム P_3 の開始から終了MB52(次の利用可能なスライスの開始MB-1)まで続く第2のエラー範囲とをマッピングすることができる。しかしながら、ルックアヘッド動作は必要ない。代わりに、ビデオデコーダ26は、フレーム P_3 に達したとき、エラーフラグのステータスに帰属して、エラー隠蔽をト

リガすることができる。エラー隠蔽モジュール 36 は、例えば、直接、復号エンジン 32 を介して、フレーム P₀ 中の M B [88 , 98] を隠蔽し、又は、損失したフレーム P₁ 及び P₂ のためにフレーム P₀ を繰り返し、フレーム P₃ 中の M B [0 , 52] を隠蔽する。次いで、ビデオデコーダ 26 は、フレーム P₃ 中の残りのスライスを復号することに進む。

【 0168 】

上述のように、本技法は、実際のエラーが検出された後にエラー処理が行われるので、低い計算オーバーヘッドを提供することができる。従って、本技法は、不必要的計算を回避することができるので、低い電力消費を提供し、デジタル信号プロセッサ (D S P) がパーサを実装し、特定用途向け集積回路 (A S I C) がピクセル再構成を実装するときなど、2つ以上のプロセッサが関与する本技法の実装における通信及び妨害を低減する。また、本開示で説明する技法は非連続的な方法ではなく連続的な方法で動作するので、ルックアヘッド復号技法と比較すると、本技法はより効果的なハードウェアアーキテクチャ及び実装形態を与えることができる。本技法は、隠蔽を必要とするセグメントの正確な識別を通して復号エラーの影響を低減することによって、ビデオ品質を更に改善することができる。

10

【 0169 】

本明細書で説明した技術は、ハードウェア、ソフトウェア、ファームウェア、又はそれらの任意の組合せで実装できる。モジュール、ユニット、又はコンポーネントとして説明する特徴は、集積論理装置に一緒に、又は個別であるが相互運用可能な論理装置として別々に実装できる。場合によっては、様々な特徴は、集積回路チップ又はチップセットなどの集積回路装置として実装できる。ソフトウェアで実装した場合、これらの技法は、実行されると、上記で説明した方法の1つ以上をプロセッサに実行させる命令を備えるコンピュータ可読媒体によって少なくとも部分的に実現できる。

20

【 0170 】

コンピュータ可読媒体は、パッケージング材料を含むことがある、コンピュータプログラム製品の一部をなすことができる。コンピュータ可読媒体は、ランダムアクセスメモリ (R A M)、同期ダイナミックランダムアクセスメモリ (S D R A M)、読み取り専用メモリ (R O M)、不揮発性ランダムアクセスメモリ (N V R A M)、電気消去可能プログラマブル読み取り専用メモリ (E E P R O M)、フラッシュメモリ、磁気又は光学データ記憶媒体などのコンピュータデータ記憶媒体を備えることができる。本技法は、追加又は代替として、命令又はデータ構造の形態でコードを搬送又は伝達し、コンピュータによってアクセス、読み取り、及び / 又は実行できるコンピュータ可読通信媒体によって、少なくとも部分的に実現できる。

30

【 0171 】

コード又は命令は、1つ以上のD S P、汎用マイクロプロセッサ、A S I C、フィールドプログラマブル論理アレイ (F P G A)、又は他の等価な集積又はディスクリート論理回路など、1つ以上のプロセッサによって実行できる。従って、本明細書で使用する「プロセッサ」という用語は、前述の構造、又は本明細書で説明する技法の実装に好適な他の構造のいずれかを指す。更に、幾つかの態様では、本明細書で説明する機能を、専用のソフトウェアモジュール又はハードウェアモジュールの内部に与えることができる。本開示はまた、本開示で説明した技法の1つ以上を実装する回路を含む様々な集積回路装置のいずれかを企図する。そのような回路は、単一の集積回路チップ、又はいわゆるチップセット中の複数の相互運用可能な集積回路チップで提供できる。そのような集積回路装置は様々な適用例において使用でき、適用例のいくつかは携帯電話ハンドセットなどのワイヤレス通信装置での使用を含む。

40

【 0172 】

開示する技法の様々な態様について説明した。これら及び他の態様は以下の特許請求の範囲内に入る。

【図1】

図1

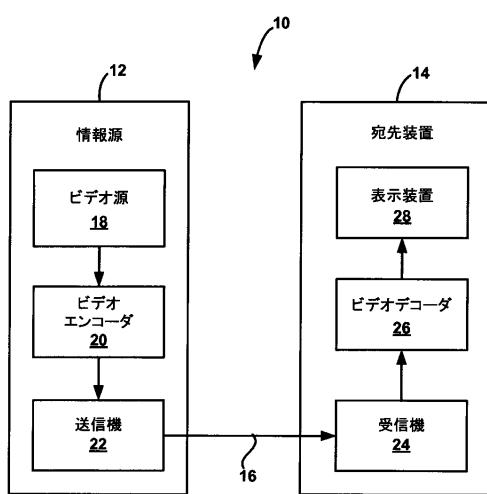


FIG. 1

【図2】

図2

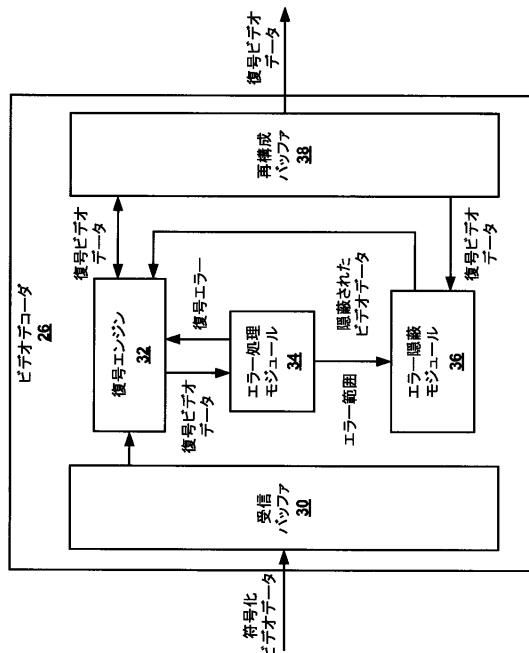


FIG. 2

【図3】

図3

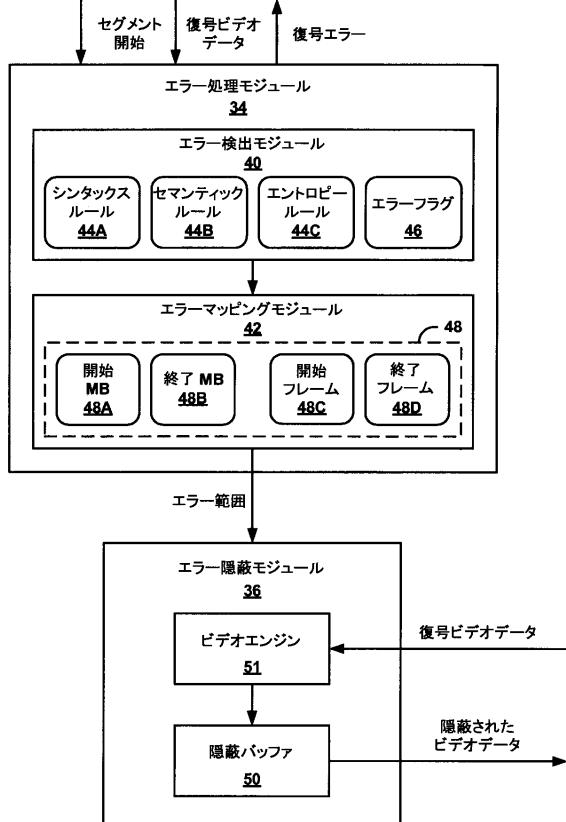


FIG. 3

【図4】

図4

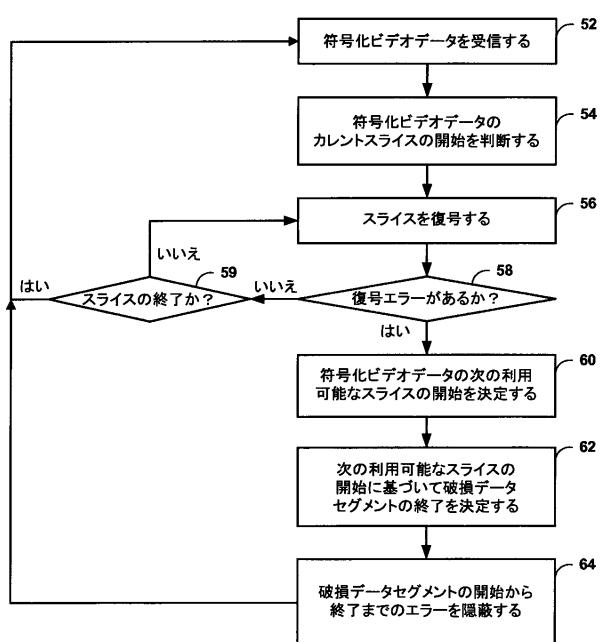


FIG. 4

【 図 5 】

图 5

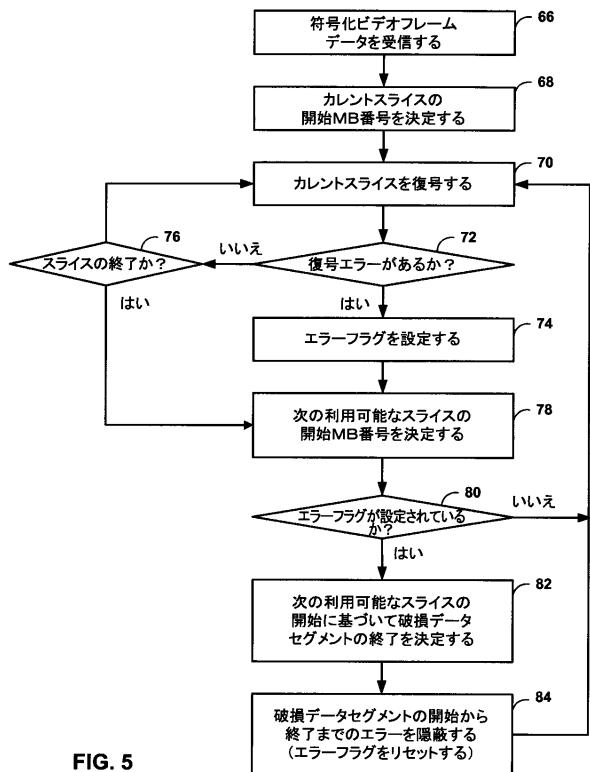


FIG. 5

【 図 6 】

図 6

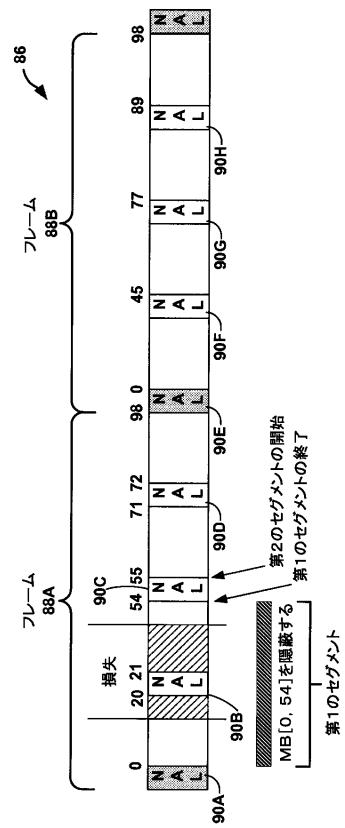


FIG. 6

【 図 7 】

四 7

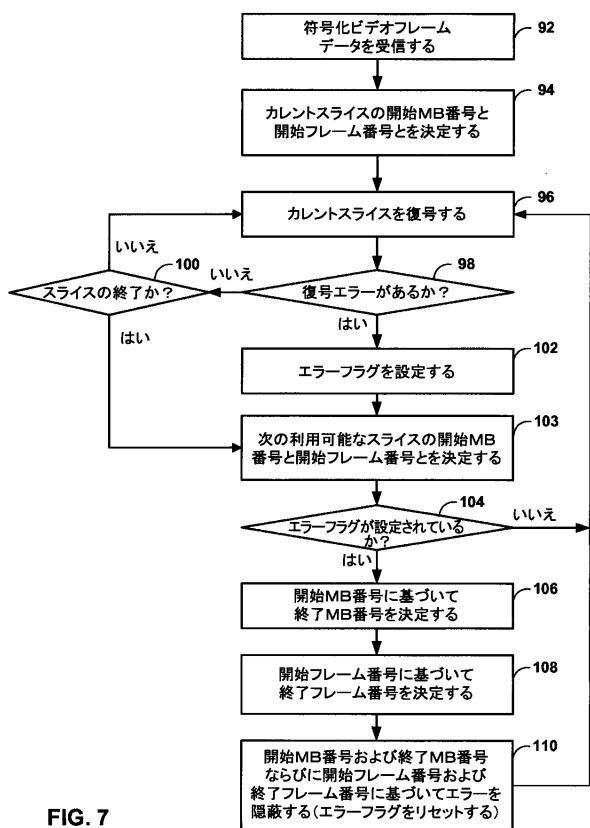


FIG. 7

【 四 8 】

8

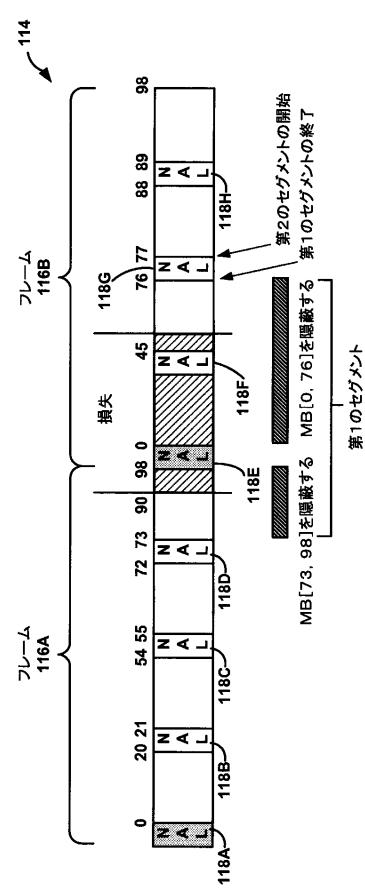


FIG. 8

【 図 9 A 】

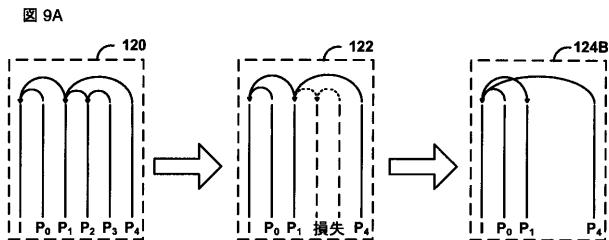


FIG. 9A

【图 9 B】

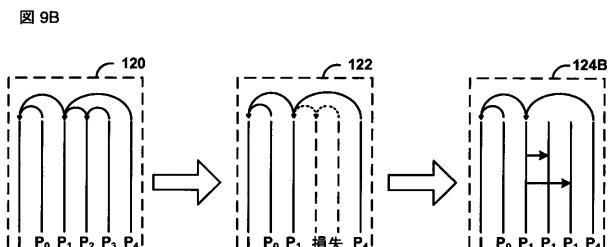


FIG. 9B

【 図 9 C 】

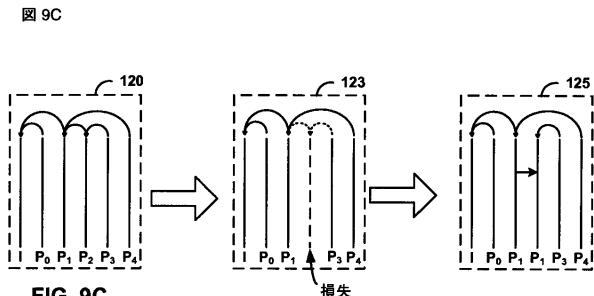


FIG. 9C

【 10 】

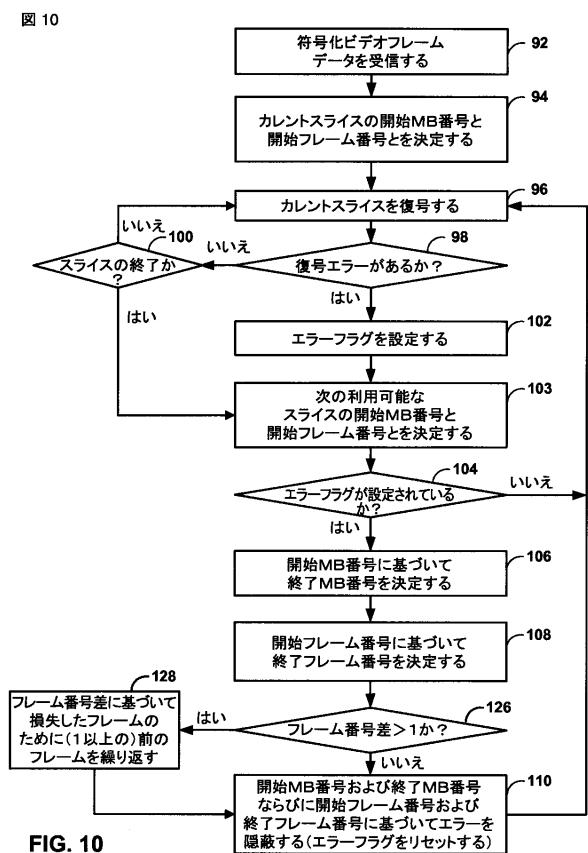


FIG. 10

【 义 1 1 A 】

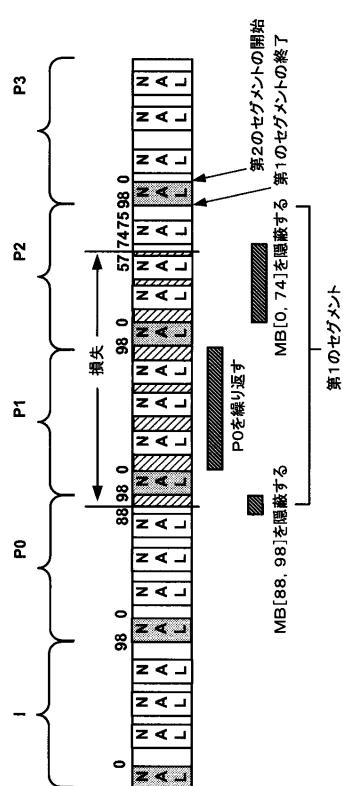


FIG. 11A

【図 11B】

図 11B

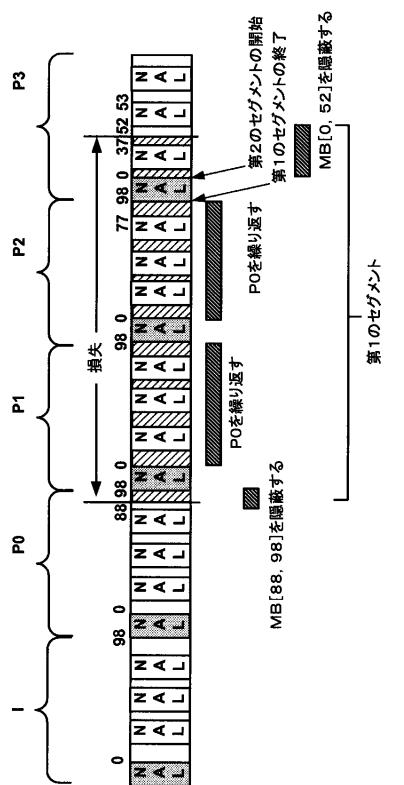


FIG. 11B

【手続補正書】

【提出日】平成22年10月26日(2010.10.26)

【手続補正1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項1】

符号化ビデオデータのカレントユニットの開始を決定することと、
前記符号化ビデオデータの次の利用可能なユニットの開始を決定しないで前記カレント
ユニットの少なくとも一部分を復号することと、

前記カレントユニット中の復号エラーを検出することと、
前記復号エラーが検出されたとき、前記符号化ビデオデータの前記次の利用可能なユニ
ットの前記開始を決定することと、

前記次の利用可能なユニットの前記開始に基づいて破損データセグメントの終了を決定
 することと、

前記カレントユニットの前記開始と前記破損データセグメントの前記終了とに基づいて
 前記破損データセグメントを隠蔽することと、

を含む、ビデオ復号方法。

【請求項2】

前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータストリーム
 を含む、請求項1に記載の方法。

【請求項3】

前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの少なくと

も 1 つの追加のユニットとを含む、請求項 1 に記載の方法。

【請求項 4】

前記カレントユニットが第 1 のフレームのカレントスライスであり、前記破損データセグメントが、前記カレントスライスと第 2 のフレームからの符号化ビデオデータの少なくとも 1 つの追加のスライスとを含む、請求項 1 に記載の方法。

【請求項 5】

前記復号エラーが検出された場合、前記復号エラーの前記検出を示すためにエラーフラグを設定することと、

前記エラーフラグが設定されているとき、前記次の利用可能なユニットの前記開始に基づいて前記破損データセグメントの前記終了を決定することと
を更に備える、請求項 1 に記載の方法。

【請求項 6】

前記次の利用可能なユニットが少なくとも 1 つのスライスを備え、前記方法は、前記復号エラーが検出されたとき、前記スライスに関連するスライスヘッダを探すことと更に備える、請求項 5 に記載の方法。

【請求項 7】

前記復号エラーが検出された場合、前記カレントユニットの復号を止めることと、前記次の利用可能なユニットを復号することに進むこととを更に含む、請求項 5 に記載の方法。

【請求項 8】

前記復号エラーが検出されなかった場合、第 1 のユニット全体を復号することと更に含む、請求項 1 に記載の方法。

【請求項 9】

前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエントロピー符号化工エラーの少なくとも 1 つを含む、請求項 1 に記載の方法。

【請求項 10】

前記カレントユニットの開始を決定することが、前記カレントユニットの前記開始に関連する第 1 のマクロブロック (MB) 番号を決定することを含み、前記破損データセグメントの終了を決定することが、前記第 2 のユニットの開始より前に第 2 の MB 番号を決定することを含み、隠蔽することが、前記第 1 の MB 番号から前記第 2 の MB 番号までの前記破損データセグメントを隠蔽することを含む、請求項 1 に記載の方法。

【請求項 11】

前記第 2 のユニットの開始を決定することが、前記第 2 のユニットの前記開始に関連する第 3 の MB 番号を決定することを備え、前記第 2 の MB 番号を決定することが、前記第 2 の MB 番号を生成するために、前記第 3 の MB 番号から 1 を減算することを備える、請求項 10 に記載の方法。

【請求項 12】

前記第 1 のユニットの開始を決定することが、前記第 1 のユニットの前記開始に関連する第 1 のマクロブロック (MB) 番号と第 1 のフレーム番号とを決定することを備え、前記破損データセグメントの終了を決定することが、前記破損データセグメントの前記終了に関連する第 2 の MB 番号と第 2 のフレーム番号とを決定することを備え、隠蔽することが、前記第 1 の MB 番号及び前記第 2 の MB 番号と前記第 1 のフレーム番号及び前記第 2 のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽することを備え、前記第 1 のフレーム番号と前記第 2 のフレーム番号とが、同じことも、異なることもある、請求項 10 に記載の方法。

【請求項 13】

1 つ以上のフレームの損失を検出することと、

別のフレームの参照フレーム不整合を隠蔽するために前記 1 つ以上の損失したフレームを交換することと

を更に含む、請求項 1 に記載の方法。

【請求項 1 4】

符号化ビデオデータのカレントユニットの開始を決定するための手段と、
前記符号化ビデオデータの次の利用可能なユニットの開始を検出しないで前記カレント
ユニットの少なくとも一部分を復号するための手段と、
前記カレントユニット中の復号エラーを検出するための手段と、
前記復号エラーが検出されたとき、前記符号化ビデオデータの前記次の利用可能なユニ
ットの前記開始を決定するための手段と、
前記次の利用可能なユニットの前記開始に基づいて破損データセグメントの終了を決定
するための手段と、
前記カレントユニットの前記開始と前記破損データセグメントの前記終了とにに基づいて
前記破損データセグメントを隠蔽するための手段と、
を備える、ビデオ復号装置。

【請求項 1 5】

前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータスライス
を含む、請求項 1 4 に記載の装置。

【請求項 1 6】

前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの少なくとも
1 つの追加のユニットとを含む、請求項 1 4 に記載の装置。

【請求項 1 7】

前記カレントユニットが、第 1 のフレームのカレントスライスであり、前記破損データ
セグメントが、前記カレントスライスと第 2 のフレームからの符号化ビデオデータの少
なくとも 1 つの追加のスライスとを含む、請求項 1 4 に記載の装置。

【請求項 1 8】

前記復号エラーが検出された場合、前記復号エラーの前記検出を示すためにエラーフラ
グを設定するための手段と、

前記エラーフラグが設定されているとき、前記次の利用可能なユニットの前記開始に基
づいて前記破損データセグメントの前記終了を決定するための手段と
を更に備える、請求項 1 4 に記載の装置。

【請求項 1 9】

前記次の利用可能なユニットが少なくとも 1 つのスライスを備え、前記装置は、前記復
号エラーが検出されたとき、前記スライスに関連するスライスヘッダを探すための手段を
更に備える、請求項 1 8 に記載の装置。

【請求項 2 0】

前記復号エラーが検出された場合、前記カレントユニットの復号を止めるための手段と
、前記次の利用可能なユニットを復号することに進むための手段とを更に備える、請求項
1 8 に記載の装置。

【請求項 2 1】

前記復号エラーが検出されなかった場合、第 1 のユニット全体を復号するための手段を
更に備える、請求項 1 4 に記載の装置。

【請求項 2 2】

前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエントロピー符
号化工エラーの少なくとも 1 つを備える、請求項 1 4 に記載の装置。

【請求項 2 3】

前記カレントユニットの開始を決定するための前記手段が、前記カレントユニットの前
記開始に関連する第 1 のマクロブロック (MB) 番号を決定するための手段を備え、前記
破損データセグメントの終了を決定するための前記手段が、前記第 2 のユニットの開始よ
り前に第 2 の MB 番号を決定するための手段を備え、隠蔽するための前記手段が、前記第
1 の MB 番号から前記第 2 の MB 番号までの前記破損データセグメントを隠蔽するため
の手段を備える、請求項 1 4 に記載の装置。

【請求項 2 4】

前記第2のユニットの開始を決定するための前記手段が、前記第2のユニットの前記開始に関連する第3のMB番号を決定するための手段を備え、前記第2のMB番号を決定するための前記手段が、前記第2のMB番号を生成するために、前記第3のMB番号から1を減算するための手段を備える、請求項23に記載の装置。

【請求項25】

前記第1のユニットの開始を決定するための前記手段が、前記第1のユニットの前記開始に関連する第1のマクロブロック(MB)番号と第1のフレーム番号とを決定するための手段を備え、前記破損データセグメントの終了を決定するための前記手段が、前記破損データセグメントの前記終了に関連する第2のMB番号と第2のフレーム番号とを決定するための手段を備え、隠蔽するための前記手段が、前記第1のMB番号及び前記第2のMB番号と前記第1のフレーム番号及び前記第2のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽するための手段を備え、前記第1のフレーム番号と前記第2のフレーム番号とが、同じことも、異なることもある、請求項23に記載の装置。

【請求項26】

1つ以上のフレームの損失を検出するための手段と、
別のフレームの参照フレーム不整合を隠蔽するために前記1つ以上の損失したフレームを交換するための手段と
を更に備える、請求項14に記載の装置。

【請求項27】

符号化ビデオデータの次の利用可能なユニットの開始が決定される前に前記符号化ビデオデータのカレントユニットの少なくとも一部分を復号する復号エンジンと、

カレントユニット中の復号エラーを検出するエラー検出モジュールと、

前記カレントユニットの開始を決定し、前記復号エラーが検出されるときに、前記符号化ビデオデータの前記次の利用可能なユニットの前記開始を決定し、前記次の利用可能なユニットの前記開始に基づいて破損データセグメントの終了を決定するエラーマッピングモジュールと、

前記カレントユニットの前記開始と前記破損データセグメントの前記終了とに基づいて前記破損データセグメントを隠蔽するエラー隠蔽モジュールと、
を備える、ビデオ復号装置。

【請求項28】

前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータスライスを含む、請求項27に記載の装置。

【請求項29】

前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの少なくとも1つの追加のユニットとを含む、請求項27に記載の装置。

【請求項30】

前記カレントユニットが、第1のフレームのカレントスライスであり、前記破損データセグメントが、前記カレントスライスと第2のフレームからの符号化ビデオデータの少なくとも1つの追加のスライスとを含む、請求項27に記載の装置。

【請求項31】

前記エラー検出モジュールが、前記復号エラーの前記検出を示すためにエラーフラグを設定し、前記エラーマッピングモジュールは、前記エラーフラグが設定されているとき、前記次の利用可能なユニットの前記開始に基づいて前記破損データセグメントの前記終了を決定する、請求項27に記載の装置。

【請求項32】

前記次の利用可能なユニットが少なくとも1つのスライスを備え、前記復号エンジンは、前記復号エラーが検出されたとき、前記スライスに関連するスライスヘッダを探す、請求項31に記載の装置。

【請求項33】

前記復号エラーが検出された場合、前記復号エンジンは、前記カレントユニットの復号を止め、前記次の利用可能なユニットを復号することに進む、請求項31に記載の装置。

【請求項34】

前記復号エラーが検出されなかった場合、前記復号エンジンは、第1のユニット全体を復号する、請求項27に記載の装置。

【請求項35】

前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエントロピー符号化工ラの少なくとも1つを備える、請求項27に記載の装置。

【請求項36】

前記エラーマッピングモジュールが、前記カレントユニットの前記開始に関連する第1のマクロブロック(MB)番号を決定し、前記第2のユニットの開始より前に第2のMB番号を決定し、前記エラー隠蔽モジュールが、前記第1のMB番号から前記第2のMB番号までの前記破損データセグメントを隠蔽する、請求項27に記載の装置。

【請求項37】

前記エラーマッピングモジュールが、前記第2のユニットの前記開始に関連する第3のMB番号を決定し、前記第2のMB番号を生成するために、前記第3のMB番号から1を減算することによって前記第2のMB番号を決定する、請求項36に記載の装置。

【請求項38】

前記エラーマッピングモジュールが、前記第1のユニットの前記開始に関連する第1のマクロブロック(MB)番号と第1のフレーム番号とを決定し、前記破損データセグメントの前記終了に関連する第2のMB番号と第2のフレーム番号とを決定し、前記エラー隠蔽モジュールが、前記第1のMB番号及び前記第2のMB番号と前記第1のフレーム番号及び前記第2のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽し、前記第1のフレーム番号と前記第2のフレーム番号とが、同じことも、異なることもある、請求項36に記載の装置。

【請求項39】

前記エラー隠蔽モジュールが、1つ以上のフレームの損失を検出し、別のフレームの参照フレーム不整合を隠蔽するために前記1つ以上の損失したフレームを交換する、請求項27に記載の装置。

【請求項40】

前記装置がワイヤレス通信装置ハンドセットを備える、請求項27に記載の装置。

【請求項41】

前記装置が集積回路装置を備える、請求項27に記載の装置。

【請求項42】

符号化ビデオデータのカレントユニットの開始を決定することと、

前記符号化ビデオデータの次の利用可能なユニットの終端を決定しないで前記カレントユニットの少なくとも一部分を復号することと、

前記カレントユニット中の復号エラーを検出することと、

前記復号エラーが検出されたとき、前記符号化ビデオデータの前記次の利用可能なユニットの前記開始を決定することと、

前記次の利用可能なユニットの前記開始に基づいて破損データセグメントの終了を決定することと、

前記カレントユニットの前記開始と前記破損データセグメントの前記終了とに基づいて前記破損データセグメントを隠蔽することと、

を1つ以上のプロセッサに行わせるための命令を含む、コンピュータ可読媒体。

【請求項43】

前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータストリームを含む、請求項42に記載のコンピュータ可読媒体。

【請求項44】

前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの少なくと

も 1 つの追加のユニットとを含む、請求項 4 2 に記載のコンピュータ可読媒体。

【請求項 4 5】

前記カレントユニットが、第 1 のフレームのカレントスライスであり、前記破損データセグメントが、前記カレントスライスと第 2 のフレームからの符号化ビデオデータの少なくとも 1 つの追加のスライスとを含む、請求項 4 2 に記載のコンピュータ可読媒体。

【請求項 4 6】

前記復号エラーが検出された場合、前記復号エラーの前記検出を示すためにエラーフラグを設定することと、

前記エラーフラグが設定されているとき、前記次の利用可能なユニットの前記開始に基づいて前記破損データセグメントの前記終了を決定することと

を前記 1 つ以上のプロセッサに行わせるための命令を更に含む、請求項 4 2 に記載のコンピュータ可読媒体。

【請求項 4 7】

前記次の利用可能なユニットが少なくとも 1 つのスライスを備え、前記命令は、前記復号エラーが検出されたとき、前記スライスに関連するスライスヘッダを探すことを前記 1 つ以上のプロセッサに行わせる、請求項 4 6 に記載のコンピュータ可読媒体。

【請求項 4 8】

前記復号エラーが検出された場合、前記カレントユニットの復号を止めることと、前記次の利用可能なユニットを復号することに進むこととを前記 1 つ以上のプロセッサに行わせるための命令を更に含む、請求項 4 6 に記載のコンピュータ可読媒体。

【請求項 4 9】

前記命令は、前記復号エラーが検出されなかった場合、第 1 のユニット全体を復号することを前記 1 つ以上のプロセッサに行わせる、請求項 4 2 に記載のコンピュータ可読媒体。

【請求項 5 0】

前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエントロピー符号化工エラーの少なくとも 1 つを含む、請求項 4 2 に記載のコンピュータ可読媒体。

【請求項 5 1】

前記命令が、前記カレントユニットの前記開始に関連する第 1 のマクロブロック (MB) 番号を決定することと、前記第 2 のユニットの開始より前に第 2 の MB 番号を決定することと、前記第 1 の MB 番号から前記第 2 の MB 番号までの前記破損データセグメントを隠蔽することとを前記 1 つ以上のプロセッサに行わせる、請求項 4 2 に記載のコンピュータ可読媒体。

【請求項 5 2】

前記命令が、前記第 2 のユニットの前記開始に関連する第 3 の MB 番号を決定することと、前記第 2 の MB 番号を生成するために、前記第 3 の MB 番号から 1 を減算することとを前記 1 つ以上のプロセッサに行わせる、請求項 5 1 に記載のコンピュータ可読媒体。

【請求項 5 3】

前記命令が、前記第 1 のユニットの前記開始に関連する第 1 のマクロブロック (MB) 番号と第 1 のフレーム番号とを決定することと、前記破損データセグメントの前記終了に関連する第 2 の MB 番号と第 2 のフレーム番号とを決定することと、前記第 1 の MB 番号及び前記第 2 の MB 番号と前記第 1 のフレーム番号及び前記第 2 のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽することとを前記 1 つ以上のプロセッサに行わせ、前記第 1 のフレーム番号と前記第 2 のフレーム番号とが、同じことも、異なることもある、請求項 5 1 に記載のコンピュータ可読媒体。

【請求項 5 4】

前記命令が、

1 つ以上のフレームの損失を検出することと、

別のフレームの参照フレーム不整合を隠蔽するために前記 1 つ以上の損失したフレームを交換することと、

を前記 1 つ以上のプロセッサに行わせる、請求項 4 2 に記載のコンピュータ可読媒体。

【手続補正 2】

【補正対象書類名】明細書

【補正対象項目名】0 1 7 2

【補正方法】変更

【補正の内容】

【0 1 7 2】

開示する技法の様々な態様について説明した。これら及び他の態様は以下の特許請求の範囲内に入る。

付記

(1) 符号化ビデオデータのカレントユニットの開始を決定することと、

前記カレントユニットの少なくとも一部分を復号することと、

前記カレントユニット中の復号エラーを検出することと、

前記符号化ビデオデータの次の利用可能なユニットの開始を決定することと、

前記復号エラーが検出された場合、前記次の利用可能なユニットの前記開始に基づいて破損データセグメントの終了を決定することと、

前記カレントユニットの前記開始と前記破損データセグメントの前記終了とに基づいて前記破損データセグメントを隠蔽することと、

を含む、ビデオ復号方法。

(2) 前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータスライスを含む、(1)に記載の方法。

(3) 前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの少なくとも 1 つの追加のユニットとを含む、(1)に記載の方法。

(4) 前記カレントユニットが第 1 のフレームのカレントスライスであり、前記破損データセグメントが、前記カレントスライスと第 2 のフレームからの符号化ビデオデータの少なくとも 1 つの追加のスライスとを含む、(1)に記載の方法。

(5) 前記復号エラーが検出された場合、前記復号エラーの前記検出を示すためにエラーフラグを設定することと、前記エラーフラグが設定されているとき、前記次の利用可能なユニットの前記開始に基づいて前記破損データセグメントの前記終了を決定することと、を更に備える、(5)に記載の方法。

(6) 前記次の利用可能なユニットが少なくとも 1 つのスライスを備え、前記方法は、前記復号エラーが検出されたとき、前記スライスに関連するスライスヘッダを探すことと、を更に備える、(5)に記載の方法。

(7) 前記復号エラーが検出された場合、前記カレントユニットの復号を止めることと、前記次の利用可能なユニットを復号することに進むことを更に含む、(5)に記載の方法。

(8) 前記復号エラーが検出されなかった場合、第 1 のユニット全体を復号することと、を更に含む、(1)に記載の方法。

(9) 前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエントロピー符号化工エラーの少なくとも 1 つを含む、(1)に記載の方法。

(10) 前記カレントユニットの開始を決定することが、前記カレントユニットの前記開始に関連する第 1 のマクロブロック(MB)番号を決定することを含み、前記破損データセグメントの終了を決定することが、前記第 2 のユニットの開始より前に第 2 の MB 番号を決定することを含み、隠蔽することが、前記第 1 の MB 番号から前記第 2 の MB 番号までの前記破損データセグメントを隠蔽することを含む、(1)に記載の方法。

(11) 前記第 2 のユニットの開始を決定することが、前記第 2 のユニットの前記開始に関連する第 3 の MB 番号を決定することを備え、前記第 2 の MB 番号を決定することが、前記第 2 の MB 番号を生成するために、前記第 3 の MB 番号から 1 を減算することを備える、(10)に記載の方法。

(12) 前記第 1 のユニットの開始を決定することが、前記第 1 のユニットの前記開始

に関連する第1のマクロブロック(MB)番号と第1のフレーム番号とを決定することを備え、前記破損データセグメントの終了を決定することが、前記破損データセグメントの前記終了に関連する第2のMB番号と第2のフレーム番号とを決定することを備え、隠蔽することが、前記第1のMB番号及び前記第2のMB番号と前記第1のフレーム番号及び前記第2のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽することを備え、前記第1のフレーム番号と前記第2のフレーム番号とが、同じこととも、異なることもある、(10)に記載の方法。

(13) 1つ以上のフレームの損失を検出することと、別のフレームの参照フレーム不整合を隠蔽するために前記1つ以上の損失したフレームを交換することと、を更に含む、(1)に記載の方法。

(14) 符号化ビデオデータのカレントユニットの開始を決定するための手段と、前記カレントユニットの少なくとも一部分を復号するための手段と、前記カレントユニット中の復号エラーを検出するための手段と、前記符号化ビデオデータの次の利用可能なユニットの開始を決定するための手段と、前記復号エラーが検出された場合、前記次の利用可能なユニットの前記開始に基づいて破損データセグメントの終了を決定するための手段と、

前記カレントユニットの前記開始と前記破損データセグメントの前記終了とに基づいて前記破損データセグメントを隠蔽するための手段と、を備える、ビデオ復号装置。

(15) 前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータスライスを含む、(14)に記載の装置。

(16) 前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの少なくとも1つの追加のユニットとを含む、(14)に記載の装置。

(17) 前記カレントユニットが、第1のフレームのカレントスライスであり、前記破損データセグメントが、前記カレントスライスと第2のフレームからの符号化ビデオデータの少なくとも1つの追加のスライスとを含む、(14)に記載の装置。

(18) 前記復号エラーが検出された場合、前記復号エラーの前記検出を示すためにエラーフラグを設定するための手段と、前記エラーフラグが設定されているとき、前記次の利用可能なユニットの前記開始に基づいて前記破損データセグメントの前記終了を決定するための手段と、を更に備える、(14)に記載の装置。

(19) 前記次の利用可能なユニットが少なくとも1つのスライスを備え、前記装置は、前記復号エラーが検出されたとき、前記スライスに関連するスライスヘッダを探すための手段を更に備える、(18)に記載の装置。

(20) 前記復号エラーが検出された場合、前記カレントユニットの復号を止めるための手段と、前記次の利用可能なユニットを復号することに進むための手段とを更に備える、(18)に記載の装置。

(21) 前記復号エラーが検出されなかった場合、第1のユニット全体を復号するための手段を更に備える、(14)に記載の装置。

(22) 前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエントロピー符号化工エラーの少なくとも1つを備える、(14)に記載の装置。

(23) 前記カレントユニットの開始を決定するための前記手段が、前記カレントユニットの前記開始に関連する第1のマクロブロック(MB)番号を決定するための手段を備え、前記破損データセグメントの終了を決定するための前記手段が、前記第2のユニットの開始より前に第2のMB番号を決定するための手段を備え、隠蔽するための前記手段が、前記第1のMB番号から前記第2のMB番号までの前記破損データセグメントを隠蔽するための手段を備える、(14)に記載の装置。

(24) 前記第2のユニットの開始を決定するための前記手段が、前記第2のユニットの前記開始に関連する第3のMB番号を決定するための手段を備え、前記第2のMB番号を決定するための前記手段が、前記第2のMB番号を生成するために、前記第3のMB番号から1を減算するための手段を備える、(23)に記載の装置。

(25) 前記第1のユニットの開始を決定するための前記手段が、前記第1のユニットの前記開始に関連する第1のマクロブロック(MB)番号と第1のフレーム番号とを決定するための手段を備え、前記破損データセグメントの終了を決定するための前記手段が、前記破損データセグメントの前記終了に関連する第2のMB番号と第2のフレーム番号とを決定するための手段を備え、隠蔽するための前記手段が、前記第1のMB番号及び前記第2のMB番号と前記第1のフレーム番号及び前記第2のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽するための手段を備え、前記第1のフレーム番号と前記第2のフレーム番号とが、同じことも、異なることもある、(23)に記載の装置。

(26) 1つ以上のフレームの損失を検出するための手段と、別のフレームの参照フレーム不整合を隠蔽するために前記1つ以上の損失したフレームを交換するための手段と、を更に備える、(14)に記載の装置。

(27) 符号化ビデオデータのカレントユニットの少なくとも一部分を復号する復号エンジンと、

カレントユニット中の復号エラーを検出するエラー検出モジュールと、

前記カレントユニットの開始を決定し、前記符号化ビデオデータの次の利用可能なユニットの開始を決定し、前記復号エラーが検出された場合、前記次の利用可能なユニットの前記開始に基づいて破損データセグメントの終了を決定するエラーマッピングモジュールと、

前記カレントユニットの前記開始と前記破損データセグメントの前記終了とに基づいて前記破損データセグメントを隠蔽するエラー隠蔽モジュールと、
を備える、ビデオ復号装置。

(28) 前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータスライスを含む、(27)に記載の装置。

(29) 前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの少なくとも1つの追加のユニットとを含む、(27)に記載の装置。

(30) 前記カレントユニットが、第1のフレームのカレントスライスであり、前記破損データセグメントが、前記カレントスライスと第2のフレームからの符号化ビデオデータの少なくとも1つの追加のスライスとを含む、(27)に記載の装置。

(31) 前記エラー検出モジュールが、前記復号エラーの前記検出を示すためにエラーフラグを設定し、前記エラーマッピングモジュールは、前記エラーフラグが設定されているとき、前記次の利用可能なユニットの前記開始に基づいて前記破損データセグメントの前記終了を決定する、(27)に記載の装置。

(32) 前記次の利用可能なユニットが少なくとも1つのスライスを備え、前記復号エンジンは、前記復号エラーが検出されたとき、前記スライスに関連するスライスヘッダを探す、(31)に記載の装置。

(33) 前記復号エラーが検出された場合、前記復号エンジンは、前記カレントユニットの復号を止め、前記次の利用可能なユニットを復号することに進む、(31)に記載の装置。

(34) 前記復号エラーが検出されなかった場合、前記復号エンジンは、第1のユニット全体を復号する、(27)に記載の装置。

(35) 前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエントロピー符号化工エラーの少なくとも1つを備える、(27)に記載の装置。

(36) 前記エラーマッピングモジュールが、前記カレントユニットの前記開始に関連する第1のマクロブロック(MB)番号を決定し、前記第2のユニットの開始より前に第2のMB番号を決定し、前記エラー隠蔽モジュールが、前記第1のMB番号から前記第2のMB番号までの前記破損データセグメントを隠蔽する、(27)に記載の装置。

(37) 前記エラーマッピングモジュールが、前記第2のユニットの前記開始に関連する第3のMB番号を決定し、前記第2のMB番号を生成するために、前記第3のMB番号から1を減算することによって前記第2のMB番号を決定する、(36)に記載の装置。

(38) 前記エラーマッピングモジュールが、前記第1のユニットの前記開始に関連する第1のマクロブロック(MB)番号と第1のフレーム番号とを決定し、前記破損データセグメントの前記終了に関連する第2のMB番号と第2のフレーム番号とを決定し、前記エラー隠蔽モジュールが、前記第1のMB番号及び前記第2のMB番号と前記第1のフレーム番号及び前記第2のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽し、前記第1のフレーム番号と前記第2のフレーム番号とが、同じこととも、異なることもある、(36)に記載の装置。

(39) 前記エラー隠蔽モジュールが、1つ以上のフレームの損失を検出し、別のフレームの参照フレーム不整合を隠蔽するために前記1つ以上の損失したフレームを交換する、(27)に記載の装置。

(40) 前記装置がワイヤレス通信装置ハンドセットを備える、(27)に記載の装置。
。

(41) 前記装置が集積回路装置を備える、(27)に記載の装置。

(42) 符号化ビデオデータのカレントユニットの開始を決定することと、
前記カレントユニットの少なくとも一部分を復号することと、
前記カレントユニット中の復号エラーを検出することと、
前記符号化ビデオデータの次の利用可能なユニットの開始を決定することと、
前記復号エラーが検出された場合、前記次の利用可能なユニットの前記開始に基づいて
破損データセグメントの終了を決定することと、
前記カレントユニットの前記開始と前記破損データセグメントの前記終了とに基づいて
前記破損データセグメントを隠蔽することと、
を1つ以上のプロセッサに行わせるための命令を含む、コンピュータ可読媒体。

(43) 前記カレントユニット及び前記次の利用可能なユニットの各々がビデオデータ
スライスを含む、(42)に記載のコンピュータ可読媒体。

(44) 前記破損データセグメントが、前記カレントユニットと符号化ビデオデータの
少なくとも1つの追加のユニットとを含む、(42)に記載のコンピュータ可読媒体。

(45) 前記カレントユニットが、第1のフレームのカレントスライスであり、前記破
損データセグメントが、前記カレントスライスと第2のフレームからの符号化ビデオデータの
少なくとも1つの追加のスライスとを含む、(42)に記載のコンピュータ可読媒体。
。

(46) 前記復号エラーが検出された場合、前記復号エラーの前記検出を示すために
エラーフラグを設定することと、前記エラーフラグが設定されているとき、前記次の利用可
能なユニットの前記開始に基づいて前記破損データセグメントの前記終了を決定すること
と、を前記1つ以上のプロセッサに行わせるための命令を更に含む、(42)に記載のコン
ピュータ可読媒体。

(47) 前記次の利用可能なユニットが少なくとも1つのスライスを備え、前記命令は
、前記復号エラーが検出されたとき、前記スライスに関連するスライスヘッダを探すこと
を前記1つ以上のプロセッサに行わせる、(46)に記載のコンピュータ可読媒体。

(48) 前記復号エラーが検出された場合、前記カレントユニットの復号を止めること
と、前記次の利用可能なユニットを復号することに進むことを前記1つ以上のプロセッ
サに行わせるための命令を更に含む、(46)に記載のコンピュータ可読媒体。

(49) 前記命令は、前記復号エラーが検出されなかった場合、第1のユニット全体を
復号することを前記1つ以上のプロセッサに行わせる、(42)に記載のコンピュータ可
読媒体。

(50) 前記復号エラーが、シンタックスエラー、セマンティックエラー、又はエント
ロピー符号化工エラーの少なくとも1つを含む、(42)に記載のコンピュータ可読媒体。

(51) 前記命令が、前記カレントユニットの前記開始に関連する第1のマクロブロッ
ク(MB)番号を決定することと、前記第2のユニットの開始より前に第2のMB番号を
決定することと、前記第1のMB番号から前記第2のMB番号までの前記破損データセグ
メントを隠蔽することとを前記1つ以上のプロセッサに行わせる、(42)に記載のコン

ピュータ可読媒体。

(52) 前記命令が、前記第2のユニットの前記開始に関連する第3のMB番号を決定することと、前記第2のMB番号を生成するために、前記第3のMB番号から1を減算することとを前記1つ以上のプロセッサに行わせる、(51)に記載のコンピュータ可読媒体。

(53) 前記命令が、前記第1のユニットの前記開始に関連する第1のマクロブロック(MB)番号と第1のフレーム番号とを決定することと、前記破損データセグメントの前記終了に関連する第2のMB番号と第2のフレーム番号とを決定することと、前記第1のMB番号及び前記第2のMB番号と前記第1のフレーム番号及び前記第2のフレーム番号とに基づいて前記破損データセグメントの少なくとも一部分を隠蔽することとを前記1つ以上のプロセッサに行わせ、前記第1のフレーム番号と前記第2のフレーム番号とが、同じことも、異なることもある、(51)に記載のコンピュータ可読媒体。

(54) 前記命令が、1つ以上のフレームの損失を検出することと、別のフレームの参照フレーム不整合を隠蔽するために前記1つ以上の損失したフレームを交換することと、を前記1つ以上のプロセッサに行わせる、(42)に記載のコンピュータ可読媒体。

【国際調査報告】

INTERNATIONAL SEARCH REPORT		International application No PCT/US2009/034948						
A. CLASSIFICATION OF SUBJECT MATTER INV. H04N7/26 H04N7/36 H04N7/50 H04N7/68								
According to International Patent Classification (IPC) or to both national classification and IPC								
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) H04N								
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 practical, search terms used) EPO-Internal, COMPENDEX, INSPEC								
C. DOCUMENTS CONSIDERED TO BE RELEVANT <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 2px;">Category*</th> <th style="text-align: left; padding: 2px;">Citation of document, with indication, where appropriate, of the relevant passages</th> <th style="text-align: left; padding: 2px;">Relevant to claim No.</th> </tr> </thead> <tbody> <tr> <td style="text-align: center; padding: 2px;">X</td> <td style="padding: 2px;"> XIAOWEI DING ET AL: "A novel bitstream level joint channel error concealment scheme for realtime video over wireless networks" INFOCOM 2004. TWENTY-THIRD ANNUAL JOINT CONFERENCE OF THE IEEE COMPUTER AND COMMUNICATIONS SOCIETIES, IEEE, PISCATAWAY, NJ, USA, vol. 3, 7 March 2004 (2004-03-07), pages 2163-2173, XP010740571 ISBN: 978-0-7803-8355-5 abstract Section II.B, pages 2165-2167 -/-/ </td> <td style="text-align: center; padding: 2px;">1-54</td> </tr> </tbody> </table>			Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.	X	XIAOWEI DING ET AL: "A novel bitstream level joint channel error concealment scheme for realtime video over wireless networks" INFOCOM 2004. TWENTY-THIRD ANNUAL JOINT CONFERENCE OF THE IEEE COMPUTER AND COMMUNICATIONS SOCIETIES, IEEE, PISCATAWAY, NJ, USA, vol. 3, 7 March 2004 (2004-03-07), pages 2163-2173, XP010740571 ISBN: 978-0-7803-8355-5 abstract Section II.B, pages 2165-2167 -/-/	1-54
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.						
X	XIAOWEI DING ET AL: "A novel bitstream level joint channel error concealment scheme for realtime video over wireless networks" INFOCOM 2004. TWENTY-THIRD ANNUAL JOINT CONFERENCE OF THE IEEE COMPUTER AND COMMUNICATIONS SOCIETIES, IEEE, PISCATAWAY, NJ, USA, vol. 3, 7 March 2004 (2004-03-07), pages 2163-2173, XP010740571 ISBN: 978-0-7803-8355-5 abstract Section II.B, pages 2165-2167 -/-/	1-54						
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C. <input type="checkbox"/> See patent family annex.								
* Special categories of cited documents: *A* document defining the general state of the art which is not considered to be of particular relevance *E* earlier document but published on or after the international filing date *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) *O* document referring to an oral disclosure, use, exhibition or other means *P* document published prior to the international filing date but later than the priority date claimed								
T later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the Invention *X* document of particular relevance; the claimed Invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone *Y* document of particular relevance; the claimed Invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art. *&* document member of the same patent family								
Date of the actual completion of the international search 31 July 2009	Date of mailing of the international search report 12/08/2009							
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Authorized officer Colesanti, Carlo							

INTERNATIONAL SEARCH REPORT

International application No PCT/US2009/034948

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Description of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	"Description of core experiments on error resilience in MPEG-4 video" JOINT VIDEO TEAM (JVT) OF ISO/IEC MPEG & ITU-T VCEG(ISO/IEC JTC1/SC29/WG11 AND ITU-T SG16 Q6), XX, XX, no. N1646, 11 April 1997 (1997-04-11), XP030010413 Sections 1.2, 5.3.1, 5.3.2.3 Sections 5.6 E.10 and 5.7 E.11	1-54
X	YAO WANG ET AL: "Error Resilient Video Coding Techniques - Real-time Video Communications over Unreliable Networks" IEEE SIGNAL PROCESSING MAGAZINE, IEEE SERVICE CENTER, PISCATAWAY, NJ, US, vol. 17, no. 4, 1 July 2000 (2000-07-01), pages 61-82, XP011089872 ISSN: 1053-5888 Section "Robust Entropy Coding", page 67 Section "Decoder Error Concealment", page 69	1,14,27, 42
A	HUA YANG ET AL: "Mismatch Impact on Per-Pixel End-to-End Distortion Estimation and Coding Mode Selection" MULTIMEDIA AND EXPO, 2007 IEEE INTERNATIONAL CONFERENCE ON, IEEE, PI, 1 July 2007 (2007-07-01), pages 2178-2181, XP031124091 ISBN: 978-1-4244-1016-3 abstract Section 2.2, page 2179	1-54

フロントページの続き

(81)指定国 AP(BW,GH,GM,KE,LS,MW,MZ,NA,SD,SL,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HR,HU,IE,IS,IT,LT,LU,LV,MC,MK,MT,NL,NO,PL,PT,RO,SE,SI,S,K,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AO,AT,AU,AZ,BA,BB,BG,BH,BR,BW,BY,BZ,CA,CH,CN,CO,CR,CU,CZ,DE,DK,DM,DO,DZ,EC,EE,EG,ES,FI,GB,GD,GE,GH,GM,GT,HN,HR,HU,ID,IL,IN,IS,JP,K,E,KG,KM,KN,KP,KR,KZ,LA,LK,LR,LS,LT,LU,LY,MA,MD,ME,MG,MK,MN,MW,MX,MY,MZ,NA,NG,NI,NO,NZ,OM,PG,PH,PL,PT,RO,RS,RU,SC,SD,SE,SG,SK,SL,SM,ST,SV,SY,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VC,VN,ZA,ZM,ZW

(74)代理人 100075672

弁理士 峰 隆司

(74)代理人 100095441

弁理士 白根 俊郎

(74)代理人 100084618

弁理士 村松 貞男

(74)代理人 100103034

弁理士 野河 信久

(74)代理人 100119976

弁理士 幸長 保次郎

(74)代理人 100153051

弁理士 河野 直樹

(74)代理人 100140176

弁理士 砂川 克

(74)代理人 100101812

弁理士 勝村 紘

(74)代理人 100124394

弁理士 佐藤 立志

(74)代理人 100112807

弁理士 岡田 貴志

(74)代理人 100111073

弁理士 堀内 美保子

(74)代理人 100134290

弁理士 竹内 将訓

(74)代理人 100127144

弁理士 市原 卓三

(74)代理人 100141933

弁理士 山下 元

(72)発明者 リー、イエン - チ

アメリカ合衆国、カリフォルニア州 92121、サン・ディエゴ、モアハウス・ドライブ 57
75

(72)発明者 ダイ、ミン

アメリカ合衆国、カリフォルニア州 92121、サン・ディエゴ、モアハウス・ドライブ 57
75

(72)発明者 テン、チア - ユアン

アメリカ合衆国、カリフォルニア州 92121、サン・ディエゴ、モアハウス・ドライブ 57
75

F ターム(参考) 5C159 LB13 LB16 MA00 MA34 RB09 RE01 RF01 RF09 RF28 TA76
TB06 TC31 UA05