

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第6318158号  
(P6318158)

(45) 発行日 平成30年4月25日(2018.4.25)

(24) 登録日 平成30年4月6日(2018.4.6)

(51) Int.Cl.

F 1

H04N 19/70 (2014.01)

H04N 19/70

H04N 19/105 (2014.01)

H04N 19/105

H04N 19/174 (2014.01)

H04N 19/174

請求項の数 18 (全 38 頁)

(21) 出願番号 特願2015-534584 (P2015-534584)  
 (86) (22) 出願日 平成25年9月23日 (2013.9.23)  
 (65) 公表番号 特表2015-534776 (P2015-534776A)  
 (43) 公表日 平成27年12月3日 (2015.12.3)  
 (86) 國際出願番号 PCT/US2013/061241  
 (87) 國際公開番号 WO2014/052249  
 (87) 國際公開日 平成26年4月3日 (2014.4.3)  
 審査請求日 平成28年8月16日 (2016.8.16)  
 (31) 優先権主張番号 61/708,042  
 (32) 優先日 平成24年9月30日 (2012.9.30)  
 (33) 優先権主張国 米国(US)  
 (31) 優先権主張番号 13/781,710  
 (32) 優先日 平成25年2月28日 (2013.2.28)  
 (33) 優先権主張国 米国(US)

(73) 特許権者 314015767  
 マイクロソフト テクノロジー ライセンシング、エルエルシー  
 アメリカ合衆国 ワシントン州 98052 レッドモンド ワン マイクロソフト ウェイ  
 (74) 代理人 100107766  
 弁理士 伊東 忠重  
 (74) 代理人 100070150  
 弁理士 伊東 忠彦  
 (74) 代理人 100091214  
 弁理士 大貫 進介

最終頁に続く

(54) 【発明の名称】参照ピクチャ・リスト変更情報の条件付きシグナリング

## (57) 【特許請求の範囲】

## 【請求項 1】

ビデオ・デコーダにより実行される方法であって、  
 符号化データを受信するステップと、  
 再構築されたビデオ・フレームを生成するために、前記符号化データを復号化する復号化ステップであって、  
重み付き予測フラグ及びスライス・タイプに基づいて、重み付き予測が、スライスについて有効にされるか又は無効にされるかを判定するステップと、

条件を評価する評価ステップであって、前記条件は、参照ピクチャの総数を示す変数の値が1より大きいかどうかに少なくとも部分的に依存する、評価ステップと、

前記条件が満たされたかどうかに応じて、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従って参照ピクチャ・リスト(RPL)が変更されるかどうかを示すフラグを、前記ビットストリームから条件付きで解析するステップと、

を含む、復号化ステップと、

を含む、

方法。

## 【請求項 2】

前記フラグは、前記ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従って第1のRPLが変更されるかどうかを示す第1のフラグ、又は、前記ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従って第2のRPL

10

20

L が変更されるかどうかを示す第 2 のフラグである、

請求項 1 記載の方法。

**【請求項 3】**

前記フラグは、前記スライスのスライス・ヘッダの RPL 変更構造から、条件付きで解析される、

請求項 1 記載の方法。

**【請求項 4】**

前記条件は、前記フラグを含む RPL 変更構造の処理の一部として評価される、

請求項 1 記載の方法。

**【請求項 5】**

前記条件は、前記スライスのスライス・ヘッダの処理の一部として評価される、

請求項 1 記載の方法。

**【請求項 6】**

前記フラグを含む RPL 変更構造が、前記条件が満たされたかどうかに応じて、条件付きで解析される、

請求項 5 記載の方法。

**【請求項 7】**

前記フラグが 1 に等しい場合、前記 RPL は、リスト・エントリ値のリストとして明示的に指定され、

前記フラグが 0 に等しい場合、前記 RPL は、暗黙的に決定され、

前記フラグが前記ビットストリームに存在しない場合、前記フラグは、0 に等しいと推定される、

請求項 1 記載の方法。

**【請求項 8】**

前記フラグは、1 ビットを用いた符号なし整数として、前記ビットストリームにおいてシグナリングされ、前記フラグの解析プロセスは、前記符号なし整数のバイナリ表現として 1 ビットを読み取る関数の戻り値により指定されることが可能である、

請求項 1 記載の方法。

**【請求項 9】**

前記復号化ステップは、

RPL 変更構造が、前記スライスのスライス・ヘッダ内に存在し得るかどうかを示す別のフラグを、前記ビットストリームから解析する解析ステップであって、前記別のフラグは、ピクチャ・パラメータ・セットの一部としてシグナリングされる、解析ステップをさらに含む、

請求項 1 記載の方法。

**【請求項 10】**

メモリ及び 1 以上の処理装置を備えたコンピューティング・デバイスであって、前記コンピューティング・デバイスは、ビデオ・デコーダ・システムを実装しており、前記ビデオ・デコーダ・システムは、

符号化データを受け入れるよう構成されているバッファと、

前記符号化データの復号化を実行するよう構成されているビデオ・デコーダであって、前記復号化は、

重み付き予測フラグ及びスライス・タイプに基づいて、重み付き予測が、スライスについて有効にされるか又は無効にされるかを判定することと、

条件を評価することであって、前記条件は、参照ピクチャの総数を示す変数の値が 1 より大きいかどうかに少なくとも部分的に依存する、評価することと、

前記条件が満たされたかどうかに応じて、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従って参照ピクチャ・リスト (RPL) が変更されるかどうかを示すフラグを、前記ビットストリームから条件付きで解析することと、

を含む、ビデオ・デコーダと、

10

20

30

40

50

を含む、  
コンピューティング・デバイス。

**【請求項 11】**

前記フラグは、前記ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従って第1のRPLが変更されるかどうかを示す第1のフラグ、又は、前記ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従って第2のRPLが変更されるかどうかを示す第2のフラグである、

請求項10記載のコンピューティング・デバイス。

**【請求項 12】**

前記フラグは、前記スライスのスライス・ヘッダのRPL変更構造から、条件付きで解析される、

請求項10記載のコンピューティング・デバイス。

**【請求項 13】**

前記条件は、前記スライスのスライス・ヘッダの処理の一部として評価される、

請求項10記載のコンピューティング・デバイス。

**【請求項 14】**

前記フラグを含むRPL変更構造が、前記条件が満たされたかどうかに応じて、条件付きで解析される、

請求項13記載のコンピューティング・デバイス。

**【請求項 15】**

前記フラグが1に等しい場合、前記RPLは、リスト・エントリ値のリストとして明示的に指定され、

前記フラグが0に等しい場合、前記RPLは、暗黙的に決定され、

前記フラグが前記ビットストリームに存在しない場合、前記フラグは、0に等しいと推定される、

請求項10記載のコンピューティング・デバイス。

**【請求項 16】**

前記フラグは、1ビットを用いた符号なし整数として、前記ビットストリームにおいてシグナリングされ、前記フラグの解析プロセスは、前記符号なし整数のバイナリ表現として1ビットを読み取る関数の戻り値により指定されることが可能である、

請求項10記載のコンピューティング・デバイス。

**【請求項 17】**

前記復号化は、

RPL変更構造が、前記スライスのスライス・ヘッダ内に存在し得るかどうかを示す別のフラグを、前記ビットストリームから解析することであって、前記別のフラグは、ピクチャ・パラメータ・セットの一部としてシグナリングされる、解析することをさらに含む、

請求項10記載のコンピューティング・デバイス。

**【請求項 18】**

コンピューティング・デバイスに、

符号化データを受信することと、

再構築されたビデオ・フレームを生成するために、前記符号化データを復号化することであって、前記復号化は、

重み付き予測フラグ及びスライス・タイプに基づいて、重み付き予測が、スライスについて有効にされるか又は無効にされるかを判定することと、

条件を評価することであって、前記条件は、参照ピクチャの総数を示す変数の値が1より大きいかどうかに少なくとも部分的に依存する、評価することと、

前記条件が満たされたかどうかに応じて、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従って参照ピクチャ・リスト(RPL)が変更されるかどうかを示すフラグを、前記ビットストリームから条件付きで解析することと、

10

20

30

40

50

を含む、復号化することと、  
を含む動作を実行させる、  
プログラム。

【発明の詳細な説明】

【背景技術】

【0001】

エンジニアは、デジタル・ビデオのビット・レートを低減するために、圧縮（ソース符号化とも呼ばれる）を使用する。圧縮は、ビデオ情報をより低いビット・レート形式に変換することにより、ビデオ情報を記憶して伝送するコストを低減させる。伸張（復号化とも呼ばれる）は、圧縮形式からオリジナル情報のバージョンを再構築する。「コーデック」はエンコーダ／デコーダ・システムである。10

【0002】

ここ20年の間、H.261規格、H.262（MPEG-2又はISO/IEC 13818-2）規格、H.263規格、及びH.264（AVC又はISO/IEC 14496-10）規格、並びに、MPEG-1（ISO/IEC 11172-2）規格、MPEG-4 Visual（ISO/IEC 14496-2）規格、及びSMPTE 421M規格を含む様々なビデオ・コーデック規格が採用されてきた。より最近では、HEVC規格を策定中である。ビデオ・コーデック規格は、通常、特定の機能が符号化及び復号化において使用されるときの、符号化ビデオ・ビットストリームのシンタックスのためのオプション、ビットストリームにおける詳細パラメータを定義している。多くの場合、ビデオ・コーデック規格はまた、デコーダが復号化において正しい結果を達成するために実行すべき復号化オペレーションに関する詳細を提供している。コーデック規格とは別に、様々なプライエタリ・コーデック・フォーマット(proprietary codec format)が、符号化ビデオ・ビットストリームのシンタックスのための他のオプション及び対応する復号化オペレーションを定義している。20

【0003】

ビットストリームにおけるいくつかのタイプのパラメータは、ビデオ符号化及び復号化中に使用される参照ピクチャに関する情報を示す。参照ピクチャは、一般に、他のピクチャの復号化プロセスにおけるピクチャ間予測のために使用され得るサンプルを含むピクチャである。通常、他のピクチャは、復号化の順番において、参照ピクチャに続き、動き補償予測のために参照ピクチャを使用する。いくつかのビデオ・コーデック規格及びフォーマットでは、動き補償予測に使用するために、所与の時間において、複数の参照ピクチャが利用可能である。そのようなビデオ・コーデック規格／フォーマットは、複数の参照ピクチャをどのように管理するかを規定している。30

【0004】

一般に、参照ピクチャ・リスト（「RPL」）は、動き補償予測のために使用される参照ピクチャのリストである。いくつかのビデオ・コーデック規格及びフォーマットでは、参照ピクチャ・セット（「RPS」）は、所与の時間での動き補償予測における使用のために利用可能な参照ピクチャのセットであり、RPLは、RPS内の参照ピクチャの一部である。RPL内の参照ピクチャは、参照インデックスによりアドレス指定される。参照インデックスは、RPL内の参照ピクチャを特定する。符号化及び復号化中、RPSは、新たに復号化されたピクチャと、参照ピクチャとしてもはや使用されないより古いピクチャとを考慮に入れるために、更新され得る。また、より一般に使用される参照ピクチャが、シグナリングするのにより効率的である参照インデックスにより特定されるように、RPL内の参照ピクチャは、並べ替えられ得る。いくつかの最近のコーデック規格では、RPLは、RPSに関する利用可能な情報、ルールに従った変更、及び／又はビットストリームにおいてシグナリングされる変更に基づいて、符号化及び復号化中に構築される。RPLの変更のシグナリングは、大量のビットを消費し得る。40

【発明の概要】

【0005】

要約すると、詳細な説明は、参照ピクチャ・リスト（「RPL」）変更情報のシグナリングにおけるイノベーションを提供する。より一般的に、イノベーションは、RPL変更情報が使用されないときに、又はそのような情報の値を推定することができるときに、RPL変更情報のシグナリングを避けるための様々な方法に関する。

#### 【0006】

本明細書で説明するイノベーションの一態様に従うと、ビデオ・エンコーダは、条件を評価する。評価の結果に応じて、エンコーダは、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従ってRPLが変更されるかどうかを示すフラグを、ビットストリームにおいて条件付きでシグナリングする。対応するビデオ・デコーダは、条件を評価する。評価の結果に応じて、デコーダは、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従ってRPLが変更されるかどうかを示すフラグを、ビットストリームから条件付きで解析する。いくつかの例示的な実施例において、RPLは、予測（「P」）スライス又は双予測（「B」）スライスに関するものであり得る。代替的に、より高いレベルのシンタックス構造が、条件の評価に基づいて、条件付きでシグナリング／解析されてもよい。

10

#### 【0007】

いくつかの例示的な実施例において、RPLが変更されない場合、デフォルトRPLが、RPSからのRPL構築に関するルールに基づいて構築される。RPLが変更される場合、置換RPLが、RPSからの参照ピクチャの選択を示す、シグナリングされたRPL変更情報に基づいて構築される。代替的に、デフォルトRPLを並び替える変更、デフォルトRPLに参照ピクチャを追加する変更、又はデフォルトRPLから参照ピクチャを除去する変更が、デフォルトRPLを調整するために、よりきめの細かい形でシグナリングされてもよい。

20

#### 【0008】

例えば、評価される条件は、参照ピクチャの総数を示す変数に少なくとも部分的に依存する。いくつかの例示的な実施例において、条件は、その変数の値が1より大きいかどうかである。

#### 【0009】

条件は、フラグを含むRPL変更構造の処理の一部として評価され得る。あるいは、条件は、スライス・ヘッダの処理の一部として評価されてもよい。その場合、（フラグを含む）RPL変更構造は、評価の結果に応じて、条件付きでシグナリング又は解析される。

30

#### 【0010】

本明細書で説明する別の態様に従うと、ビデオ・エンコーダは、別の条件を評価する。評価の結果に応じて、エンコーダは、RPLをどのように変更する（例えば、置換する、調整する）かを示すリスト・エントリの1以上のシンタックス要素を、ビットストリームにおいて条件付きでシグナリングする。対応するビデオ・デコーダは、その条件を評価する。評価の結果に応じて、デコーダは、RPLをどのように変更する（例えば、置換する、調整する）かを示すリスト・エントリの1以上のシンタックス要素を、ビットストリームから条件付きで解析する。いくつかの例示的な実施例において、RPLは、Pスライス又はBスライスに関するものであり得る（Bスライスに関して、複数のRPLのそれぞれについて、条件評価と、条件付きシグナリング／解析とが繰り返される）。例えば、他の条件は、参照ピクチャの総数を示す変数、RPLのアクティブ参照ピクチャの数、及び／又は重み付き予測が無効にされるかどうかに少なくとも部分的に依存する。異なるロジックを使用して、現在のスライスがPスライス又はBスライスであるかに応じて、且つ／あるいはどのRPLがシグナリング／解析されているかどうかに応じて、重み付き予測が無効にされるかどうかをチェックすることができる。いくつかの例示的な実施例において、（a）参照ピクチャの総数が2に等しく、（b）RPLのアクティブ参照ピクチャの数が1に等しい場合、リスト・エントリの1以上のシンタックス要素は、ビットストリームには存在せず、リスト・エントリのうちの1つに関する値が推定される。さらに、いくつかの例示的な実施例において、（c）参照ピクチャの総数が2に等しく、（d）RPLのア

40

50

クティブ参照ピクチャの数が 2 に等しく、(e) 重み付き予測が無効にされる場合、リスト・エントリの 1 以上のシンタックス要素は、ビットストリームには存在せず、リスト・エントリのうちの 2 つに関する値が推定される。

#### 【0011】

本明細書で説明するイノベーションの別の態様に従うと、ビデオ・エンコーダは、別の条件を評価する。評価の結果に応じて、エンコーダは、RPL をどのように変更する（例えば、置換する、調整する）かを示すリスト・エントリの 1 以上のシンタックス要素のビットストリームにおけるシグナリングを調整する。詳細には、1 以上のシンタックス要素の少なくとも 1 つの長さ（ビット単位）が調整される。対応するビデオ・デコーダは、その条件を評価する。評価の結果に応じて、デコーダは、RPL をどのように変更するかを示すリスト・エントリの 1 以上のシンタックス要素のビットストリームからの解析を調整する（再度、この場合も、1 以上のシンタックス要素の少なくとも 1 つの長さ（ビット単位）が調整される）。例えば、条件は、重み付き予測が無効にされるかどうかに少なくとも部分的に依存する。異なるロジックを使用して、現在のスライスが P スライス又は B スライスであるかに応じて、且つ / あるいはどの RPL がシグナリング / 解析されているかどうかに応じて、重み付き予測が無効にされるかどうかをチェックすることができる。いくつかの例示的な実施例において、リスト・エントリのインデックス i について、重み付き予測が無効にされる場合、1 以上のシンタックス要素の少なくとも 1 つの長さ（ビット単位）は、i が増えるにつれ、減少する。例えば、いくつかの例示的な実施例において、重み付き予測が無効にされる場合、リスト・エントリ [i] の所与のシンタックス要素の長さは、Ceil(Log2(NumPocTotalCurr-i)) ビットである。一方、重み付き予測が有効にされる場合、リスト・エントリ [i] の所与のシンタックス要素の長さは、Ceil(Log2(NumPocTotalCurr)) ビットである。10

#### 【0012】

符号化又は復号化は、方法の一部として、方法を実行するよう適合されたコンピューティング・デバイスの一部として、又はコンピューティング・デバイスに方法を実行させるコンピュータ実行可能命令を記憶した有体のコンピュータ読み取り可能媒体の一部として実施され得る。20

#### 【0013】

本発明の前述の目的及び他の目的、特徴、並びに利点が、添付の図面を参照しながら進む以下の詳細な説明からより明らかになるであろう。30

#### 【図面の簡単な説明】

#### 【0014】

【図 1】いくつかの説明する実施形態を実装することができる例示的なコンピューティング・システムの図。

【図 2 a】いくつかの説明する実施形態を実装することができる例示的なネットワーク環境の図。

【図 2 b】いくつかの説明する実施形態を実装することができる例示的なネットワーク環境の図。

【図 3】いくつかの説明する実施形態を協働して実装することができる例示的なエンコーダ・システムの図。40

【図 4】いくつかの説明する実施形態を協働して実装することができる例示的なデコーダ・システムの図。

【図 5】いくつかの説明する実施形態を協働して実装することができる例示的なビデオ・エンコーダを示す図。

【図 6】いくつかの説明する実施形態を協働して実装することができる例示的なビデオ・デコーダを示す図。

【図 7 a】いくつかの例示的な実施例に従った、RPL が変更されるかどうかを示すフラグの条件付きシグナリングを示すテーブル。

【図 7 b】他の例示的な実施例に従った、RPL が変更されるかどうかを示す 1 以上のフ50

ラグの条件付きシグナリングを示すテーブル。

【図7c】他の例示的な実施例に従った、RPLが変更されるかどうかを示す1以上のフラグの条件付きシグナリングを示すテーブル。

【図8】いくつかの例示的な実施例に従った、RPLをどのように変更するかを示すリスト・エントリのシンタックス要素の条件付きシグナリングを示すテーブル。

【図9】いくつかの例示的な実施例に従った、RPLをどのように変更するかを示すリスト・エントリのシンタックス要素の条件付きシグナリングを示すテーブル。

【図10】RPLが変更されるかどうかを示すフラグの条件付きシグナリングのため的一般化された技術を示すフローチャート。

【図11】RPLが変更されるかどうかを示すフラグの条件付き解析のため的一般化された技術を示すフローチャート。 10

【図12】RPLをどのように変更するかを示すリスト・エントリのシンタックス要素の条件付きシグナリングのため的一般化された技術を示すフローチャート。

【図13】RPLをどのように変更するかを示すリスト・エントリのシンタックス要素の条件付き解析のため的一般化された技術を示すフローチャート。

【図14】RPLをどのように変更するかを示すリスト・エントリのシンタックス要素のシグナリングを調整するため的一般化された技術を示すフローチャート。

【図15】RPLをどのように変更するかを示すリスト・エントリのシンタックス要素の解析を調整するため的一般化された技術を示すフローチャート。 20

#### 【発明を実施するための形態】

##### 【0015】

詳細な説明は、参照ピクチャ・リスト（「RPL」）変更情報のシグナリングにおけるイノベーションを提供する。そのようなイノベーションは、RPL変更情報が使用されないときに、又はそのような情報の値を推定することができるときに、RPL変更情報のシグナリングを避けるのに役立ち得る。 20

##### 【0016】

いくつかの最近のコーデック規格では、参照ピクチャ・セット（「RPS」）は、動き補償予測における使用のために利用可能な参照ピクチャのセットであり、RPLは、RPSから構築される。予測（「P」）スライスの復号化プロセスでは、RPL\_0と呼ばれる1つのRPLが存在する。双予測（「B」）スライスの復号化プロセスでは、RPL\_0及びRPL\_1と呼ばれる2つのRPLが存在する。Pスライスの復号化プロセスの開始時、RPL\_0は、RPL\_0に関する利用可能な情報（現ピクチャを復号化するデコーダにおいて利用可能な参照ピクチャのセット等）、ルールに従った変更、及び／又はビットストリームにおいてシグナリングされる変更から導出される。同様に、Bスライスの復号化プロセスの開始時、RPL\_0及びRPL\_1は、RPL\_0に関する利用可能な情報とRPL\_1に関する利用可能な情報（現ピクチャを復号化するデコーダにおいて利用可能な参照ピクチャのセット等）、ルールに従った変更、及び／又はビットストリームにおいてシグナリングされる変更から導出される。より一般的に、RPLは、RPLに関する利用可能な情報、ルールに従った変更、及び／又はビットストリームにおいてシグナリングされる変更に基づいて、符号化及び復号化中に構築される。RPLの変更のシグナリングは、大量のビットを消費し得る。いくつかの最近のコーデック規格では、RPL変更情報がどのようにシグナリングされるかにおける非効率が存在する。 40

##### 【0017】

詳細な説明は、RPL変更情報のシグナリングの領域における様々なイノベーションを提供する。いくつかの状況において、これらのイノベーションは、RPL変更情報のシンタックス要素のより効率的なシグナリングをもたらす。例えば、詳細な説明は、RPLをどのように変更するかを示すリスト・エントリのシンタックス要素の条件付きシグナリングを説明する。詳細な説明はまた、そのようなシンタックス要素をシグナリングするためのより少ないビットを使用する方法を説明する。別の例として、詳細な説明は、RPLが変更されるかどうかを示すフラグの条件付きシグナリングを説明する。 50

**【0018】**

いくつかの例示的な実施例において、RPLが変更されない場合、RPSからのRPL構築に関するルールを用いた「暗黙的」アプローチに従って、デフォルトRPLが構築される。RPLが変更される場合、RPSからの参照ピクチャの選択を示すシグナリングされるRPL変更情報を用いた「明示的」シグナリング・アプローチに従って、置換RPLが構築される。代替的に、デフォルトRPL内の参照ピクチャを並べ替える変更、デフォルトRPLに参照ピクチャを追加する変更、又はデフォルトRPLから参照ピクチャを除去する変更が、デフォルトRPLに対する特定の変更として、よりきめの細かい形でシグナリングされ得る。

**【0019】**

10

本明細書で説明するイノベーションのいくつかは、HEVC規格に固有のシンタックス要素及びオペレーションを参照して示される。例えば、HEVC規格のドラフト・バージョンJCTVC-I1003(2012年4月にジュネーブで開催されたビデオ符号化に関する合同組織(「JCT-VC」)の第9回会合の"High efficiency video coding (HEVC) text specification draft 7", JCTVC-I1003\_d5)を参照する。また、2012年10月に上海で開催されたJCT-VCの第11回会合の"High Efficiency Video Coding (HEVC) text specification draft 9", JCTVC-K1003\_d11と題するドラフト・バージョンを参照されたい。

**【0020】**

20

より一般的に、本明細書で説明する例に対する様々な代替例が可能である。例えば、本明細書で説明する方法のいくつかは、説明する方法動作の順番を変えることにより、所定の方法動作を分けることにより、所定の方法動作を繰り返すことにより、又は所定の方法動作を省略することにより、変更することができる。開示する技術の様々な態様が、組み合わせることにより、又は別々に使用され得る。様々な実施形態が、説明するイノベーションのうちの1以上を使用する。本明細書で説明するイノベーションのいくつかは、背景技術において記した問題のうちの1以上に対処する。一般的に、所与の技術/ツールが、そのような問題の全てを解決するわけではない。

**【0021】****I . 例示的なコンピューティング・システム**

図1は、説明するイノベーションのいくつかを実装することができる適切なコンピューティング・システム(100)の一般化された例を示している。コンピューティング・システム(100)は、使用又は機能の範囲に関して限定を示唆するよう意図するものではない。というのは、このイノベーションは、多様な汎用コンピューティング・システム又は専用コンピューティング・システムにおいて実施することができるからである。

30

**【0022】**

図1を参照すると、コンピューティング・システム(100)は、1以上の処理装置(110、115)及びメモリ(120、125)を含む。図1において、この最も基本的な構成(130)は破線内に含まれる。処理装置(110、115)はコンピュータ実行可能命令を実行する。処理装置は、汎用中央処理装置(「CPU」)、特定用途向け集積回路(「ASIC」)におけるプロセッサ、又は任意の他のタイプのプロセッサとすることができます。マルチ処理システムにおいて、複数の処理装置が、処理能力を増大させるために、コンピュータ実行可能命令を実行する。例えば、図1は、中央処理装置(110)に加えて、グラフィックス処理装置又は共処理装置(115)も示している。有体のメモリ(120、125)は、1以上の処理装置によりアクセス可能な、揮発性メモリ(例えば、レジスタ、キャッシュ、RAM)であってもよいし、不揮発性メモリ(例えば、ROM、EEPROM、フラッシュ・メモリ等)であってもよいし、それら2つの何らかの組合せであってもよい。メモリ(120、125)は、1以上の処理装置による実行に適したコンピュータ実行可能命令の形態で、RPL変更情報のシグナリングのための1以上のイノベーションを実装するソフトウェア(180)を記憶する。

40

**【0023】**

50

コンピューティング・システムは追加の特徴を有することができる。例えば、コンピューティング・システム(100)は、ストレージ(140)、1以上の入力デバイス(150)、1以上の出力デバイス(160)、及び1以上の通信コネクション(170)を含む。バス、コントローラ、又はネットワーク等の相互接続機構(図示せず)が、コンピューティング・システム(100)のコンポーネントを相互接続する。通常、オペレーティング・システム・ソフトウェア(図示せず)が、コンピューティング・システム(100)において実行される他のソフトウェアのための動作環境を提供し、コンピューティング・システム(100)のコンポーネントの動作を調整する。

#### 【0024】

有体のストレージ(140)は、取り外し可能であっても取り外し不可能であってもよく、磁気ディスク、磁気テープ若しくは磁気カセット、CD-ROM、DVD、又は、情報を記憶するために使用することができ、コンピューティング・システム(100)内でアクセスされ得る任意の他の媒体を含む。ストレージ(140)は、RPL変更情報のシグナリングのための1以上のイノベーションを実装するソフトウェア(180)の命令を記憶する。

10

#### 【0025】

1以上の入力デバイス(150)は、キーボード、マウス、ペン、又はトラックボール等のタッチ入力デバイス、音声入力デバイス、スキヤニング・デバイス、又はコンピューティング・システム(100)に入力を提供する別のデバイスとすることができます。ビデオ符号化に関して、1以上の入力デバイス(150)は、カメラ、ビデオ・カード、TVチューナ・カード、若しくはアナログ形態あるいはデジタル形態でビデオ入力を受信する同様のデバイス、又はビデオ・サンプルをコンピューティング・システム(100)に読み込むCD-ROMあるいはCD-RWとすることができます。1以上の出力デバイス(160)は、ディスプレイ、プリンタ、スピーカ、CDライタ、又はコンピューティング・システム(100)からの出力を提供する別のデバイスとすることができます。

20

#### 【0026】

1以上の通信コネクション(170)は、通信媒体を介した別のコンピューティング・エンティティへの通信を可能にする。通信媒体は、変調されたデータ信号により、コンピュータ実行可能命令、オーディオ入力、ビデオ入力、オーディオ出力、ビデオ出力、又は他のデータ等の情報を伝達する。変調されたデータ信号とは、信号内の情報を符号化するように設定又は変更された特性の1以上を有する信号である。例えば、通信媒体は、電気信号、光信号、RF、赤外線、又は他のキャリアを使用することができるが、これらに限定されるものではない。

30

#### 【0027】

イノベーションは、コンピュータ読み取り可能媒体の一般的なコンテキストにおいて説明することができる。コンピュータ読み取り可能媒体は、コンピューティング環境内でアクセスされ得る任意の利用可能な有体の媒体である。例えば、コンピューティング・システム(100)において、コンピュータ読み取り可能媒体は、メモリ(120、125)、ストレージ(140)、及びそれらの組合せを含むが、これらに限定されるものではない。

40

#### 【0028】

イノベーションは、コンピュータ実行可能命令の一般的なコンテキストにおいて説明することができる。コンピュータ実行可能命令は、例えば、プログラム・モジュールに含まれ、コンピューティング・システムにおいて、ターゲット実プロセッサ又は仮想プロセッサ上で実行される。一般に、プログラム・モジュールは、特定のタスクを実行するか、又は特定の抽象データ型を実装するルーチン、プログラム、ライブラリ、オブジェクト、クラス、コンポーネント、データ構造等を含む。プログラム・モジュールの機能は、様々な実施形態において、必要に応じて、組み合わされてもよいし、プログラム・モジュール間で分割されてもよい。プログラム・モジュールのコンピュータ実行可能命令は、ローカル・コンピューティング・システム又は分散コンピューティング・システム内で実行され得

50

る。

#### 【0029】

「システム」及び「デバイス」という語は、本明細書において置き換え可能に使用される。文脈が別途明確に示さない限り、これらの語は、コンピューティング・システム又はコンピューティング・デバイスのタイプに関して、いかなる限定も示すものではない。一般に、コンピューティング・システム又はコンピューティング・デバイスは、ローカルであってもよいし、分散されてもよく、専用ハードウェア及び/又は汎用ハードウェアと、本明細書で説明する機能を実装するソフトウェアとの任意の組合せを含み得る。

#### 【0030】

開示する方法は、開示する方法のいずれかを実行するよう構成された専用コンピューティング・ハードウェアを用いて実装することもできる。例えば、開示する方法は、開示する方法のいずれかを実行するよう特別に設計又は構成された集積回路（例えば、特定用途向け集積回路（「ASIC」）（ASICデジタル信号処理装置（「DSP」）、グラフィックス処理装置（「GPU」）、又はフィールド・プログラマブル・ゲート・アレイ（「FPGA」）等のプログラマブル・ロジック・デバイス（「PLD」）等）により実装することができる。

10

#### 【0031】

プレゼンテーションの目的上、詳細な説明では、コンピューティング・システムにおけるコンピュータ・オペレーションを説明するための「決定する」及び「使用する」のような語が使用される。これらの語は、コンピュータにより実行されるオペレーションの高レベル抽象表現であって、人間により実行される動作と混同すべきではない。これらの語に対応する実際のコンピュータ・オペレーションは、実装に応じて変化する。

20

#### 【0032】

##### I I . 例示的なネットワーク環境

図2a及び図2bは、ビデオ・エンコーダ（220）及びビデオ・デコーダ（270）を含む例示的なネットワーク環境（201、202）を示している。エンコーダ（220）及びデコーダ（270）は、適切な通信プロトコルを用いて、ネットワーク（250）を介して接続される。ネットワーク（250）は、インターネット又は別のコンピュータ・ネットワークを含み得る。

#### 【0033】

30

図2aに示されるネットワーク環境（201）において、各リアルタイム通信（「RTT」）ツール（210）は、双方向通信のためのエンコーダ（220）及びデコーダ（270）の両方を含む。所与のエンコーダ（220）は、SMPTE 421M規格、ISO - IEC 14496 - 10規格（H.264又はAVCとしても知られている）、HEVC規格、別の規格、又はプロプライエタリ・フォーマットに準拠する出力を生成することができ、対応するデコーダ（270）は、エンコーダ（220）から符号化データを受信することができる。双方向通信は、ビデオ会議、ビデオ通話、又は他の2パーティの通信シナリオの一部であり得る。図2aのネットワーク環境（201）は、2つのリアルタイム通信ツール（210）を含むが、ネットワーク環境（201）は、マルチパーティ通信に参加する3以上のリアルタイム通信ツール（210）を含んでもよい。

40

#### 【0034】

リアルタイム通信ツール（210）は、エンコーダ（220）による符号化を管理する。図3は、リアルタイム通信ツール（210）に含まれ得る例示的なエンコーダ・システム（300）を示している。代替的に、リアルタイム通信ツール（210）は、別のエンコーダ・システムを使用してもよい。リアルタイム通信ツール（210）はまた、デコーダ（270）による復号化も管理する。図4は、リアルタイム通信ツール（210）に含まれ得る例示的なデコーダ・システム（400）を示している。代替的に、リアルタイム通信ツール（210）は、別のデコーダ・システムを使用してもよい。

#### 【0035】

図2bに示されるネットワーク環境（202）において、符号化ツール（212）は、

50

デコーダ(270)を含む複数の再生ツール(214)に伝送するためのビデオを符号化するエンコーダ(220)を含む。一方向通信は、ビデオが符号化されて1つのロケーションから1以上の他のロケーションに送信される、ビデオ監視システム、ウェブ・カメラ・モニタリング・システム、リモート・デスクトップ会議プレゼンテーション、又は他のシナリオのために提供され得る。図2bのネットワーク環境(202)は、2つの再生ツール(214)を含むが、ネットワーク環境(202)は、それより多い又はそれより少ない再生ツール(214)を含んでもよい。一般に、再生ツール(214)は、再生ツール(214)が受信するビデオのストリームを判定するために、符号化ツール(212)と通信する。再生ツール(214)は、ストリームを受信し、適切な期間の間受信した符号化データをバッファし、復号化及び再生を開始する。

10

### 【0036】

図3は、符号化ツール(212)に含まれ得る例示的なエンコーダ・システム(300)を示している。代替的に、符号化ツール(212)は、別のエンコーダ・システムを使用してもよい。符号化ツール(212)はまた、1以上の再生ツール(214)との接続を管理するためのサーバ・サイド・コントローラ・ロジックも含み得る。図4は、再生ツール(214)に含まれ得る例示的なデコーダ・システム(400)を示している。代替的に、再生ツール(214)は、別のデコーダ・システムを使用してもよい。再生ツール(214)はまた、符号化ツール(212)との接続を管理するためのクライアント・サイド・コントローラ・ロジックも含み得る。

### 【0037】

20

#### I I I . 例示的なエンコーダ・システム

図3は、いくつかの説明する実施形態を協働して実装することができる例示的なエンコーダ・システム(300)のブロック図である。エンコーダ・システム(300)は、リアルタイム通信のための低レイテンシ符号化モード、トランスコーディング・モード、及びファイル又はストリームからのメディア再生のための通常の符号化モード等の複数の符号化モードのうちいずれかで動作することができる汎用符号化ツールであってもよいし、そのような1つの符号化モードのために適合された専用符号化ツールであってもよい。エンコーダ・システム(300)は、オペレーティング・システム・モジュールとして、アプリケーション・ライブラリの一部として、又はスタンドアロン・アプリケーションとして、実装することができる。概して、エンコーダ・システム(300)は、ビデオ・ソース(310)から一連のソース・ビデオ・フレーム(311)を受信し、チャネル(390)への出力として符号化データを生成する。チャネルに出力される符号化データは、RPL変更情報を示すシンタックス要素を含み得る。

30

### 【0038】

ビデオ・ソース(310)は、カメラ、チューナ・カード、記憶媒体、又は他のデジタル・ビデオ・ソースとすることができます。ビデオ・ソース(310)は、例えば、毎秒30フレームといったフレーム・レートで一連のビデオ・フレームを生成する。本明細書で使用されるとき、「フレーム」という語は、一般に、ソースの符号化された又は再構築されたイメージ・データを指す。プログレッシブ・ビデオでは、フレームはプログレッシブ・ビデオ・フレームである。インタレース・ビデオでは、例示的な実施形態において、インタレース・ビデオ・フレームは、符号化の前にインタレース効果が除かれる(de-interlaced)。代替的に、2つの相補的インタレース・ビデオ・フィールドが、インタレース・ビデオ・フレーム又は別々のフィールドとして符号化されてもよい。プログレッシブ・ビデオ・フレームを示すかは別にして、「フレーム」又は「ピクチャ」という語は、単一の対でないビデオ・フィールド、相補的な一对のビデオ・フィールド、所与の時間におけるビデオ・オブジェクトを表すビデオ・オブジェクト・プレーン、又はより大きなイメージにおける関心領域を示し得る。ビデオ・オブジェクト・プレーン又は領域は、シーンの複数のオブジェクト又は領域を含むより大きなイメージの一部であり得る。

40

### 【0039】

到着ソース・フレーム(311)は、複数のフレーム・バッファ記憶領域(321、3

50

22、 . . . 、 32n ) を含むソース・フレーム一時メモリ記憶領域 ( 320 ) に記憶される。フレーム・バッファ ( 321 、 322 等 ) は、ソース・フレーム記憶領域 ( 320 ) 内で 1 つのソース・フレームを保持する。ソース・フレーム ( 311 ) のうちの 1 以上がフレーム・バッファ ( 321 、 322 等 ) に記憶された後、フレーム・セレクタ ( 330 ) が、ソース・フレーム記憶領域 ( 320 ) から個々のソース・フレームを定期的に選択する。エンコーダ ( 340 ) への入力のためにフレーム・セレクタ ( 330 ) によりフレームが選択される順番は、ビデオ・ソース ( 310 ) によりフレームが生成される順番とは異なり得る。例えば、一時的に後方予測を容易にするために、あるフレームが前になる場合がある。エンコーダ ( 340 ) の前に、エンコーダ・システム ( 300 ) は、符号化の前にフレームの前処理 ( 例えば、フィルタリング ) を実行するプリ・プロセッサ ( 図示せず ) を含み得る。前処理は、符号化するためのプライマリ成分及びセカンダリ成分への色空間変換も含み得る。

#### 【 0040 】

エンコーダ ( 340 ) は、符号化フレーム ( 341 ) を生成するために、選択されたフレーム ( 331 ) を符号化するとともに、メモリ管理制御操作 ( 「 M M C O 」 ) 信号 ( 342 ) 又は参照ピクチャ・セット ( 「 R P S 」 ) 情報を生成する。現フレームが、符号化された最初のフレームではない場合、符号化プロセスを実行するとき、エンコーダ ( 340 ) は、復号化フレーム一時メモリ記憶領域 ( 360 ) に記憶された 1 以上の以前に符号化 / 復号化されたフレーム ( 369 ) を使用することができる。そのような記憶された復号化フレーム ( 369 ) は、現ソース・フレーム ( 331 ) のコンテンツのフレーム間予測のための参照ピクチャとして使用される。一般に、エンコーダ ( 340 ) は、動き推定及び動き補償、周波数変換、量子化、並びにエントロピ符号化等の符号化タスクを実行する複数の符号化モジュールを含む。エンコーダ ( 340 ) により実行される正確なオペレーションは、圧縮フォーマットに応じて変わり得る。出力される符号化データのフォーマットは、 Windows ( 登録商標 ) Media Video フォーマット、 VC - 1 フォーマット、 M P E G - x フォーマット ( 例えば、 M P E G - 1 、 M P E G - 2 、又は M P E G - 4 ) 、 H . 2 6 x フォーマット ( 例えば、 H . 2 6 1 、 H . 2 6 2 、 H . 2 6 3 、 H . 2 6 4 ) 、 H E V C フォーマット、又は他のフォーマットであり得る。

#### 【 0041 】

例えば、エンコーダ ( 340 ) において、インタ符号化された予測フレームが、参照ピクチャの例である参照フレームからの予測の観点で表される。動き推定部が、 1 以上の参照フレーム ( 369 ) に関して、ソース・フレーム ( 341 ) のサンプルのブロック又は他のセットの動きを推定する。複数の参照フレームが使用される場合、複数の参照フレームは、異なる時間的方向からのものであってもよいし、同じ時間的方向からのものであってもよい。参照フレーム ( 参照ピクチャ ) は、 1 以上の R P L の一部であり得、参照インデックスが、 1 以上の R P L 内の参照ピクチャをアドレス指定する。適切なときに新たな参照ピクチャが追加されるように、適切なときに動き補償のためにもはや使用されないより古い参照ピクチャが除去されるように、及び適切なときに参照ピクチャが並び替えられるように、 1 以上の R P L が符号化中に構築される。いくつかの実施例において、例えば、現ピクチャを符号化するとき、エンコーダ ( 340 ) は、復号化フレーム記憶領域 ( 360 ) 内の参照ピクチャを含む R P S を決定し、次いで、現ピクチャの所与のスライスを符号化するための 1 以上の R P L を作成する。R P L は、 R P S からの利用可能な参照ピクチャの選択に関するルールを適用することにより作成され得る ( 暗黙的アプローチ ) 。この場合、 R P L 変更情報は、ビットストリームにおいて明示的にシグナリングされない。あるいは、 R P L は、 R P S から利用可能な特定の参照ピクチャを選択することにより作成され得る。この場合、選択される参照ピクチャは、ビットストリームにおいてシグナリングされる R P L 変更情報内に示される。暗黙的アプローチのルールにより構築される R P L と比較すると、 R P L 変更情報は、 R P S 内の参照ピクチャのリストとして、置換 R P L を指定することができる。代替的に、 R P L 変更情報は、よりきめの細かい形で、暗黙的アプローチのルールにより構築された R P L からの 1 以上の参照ピクチャの除去、

10

20

30

40

50

暗黙的アプローチのルールにより構築された RPL への 1 以上の参照ピクチャの追加、及び / 又は暗黙的アプローチのルールにより構築された RPL 内の参照ピクチャの並べ替えを指定してもよい。

#### 【 0042 】

インタ符号化されるフレームを符号化するとき、エンコーダ (340) は、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従って RPL が変更されない動き補償の結果を評価することができ、また、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従って RPL が変更される動き補償の結果（又は、RPL を変更する複数の異なる方法の結果）を評価することができる。エンコーダ (340) は、デフォルト RPL を使用することを決定することができる（ビットストリームにおいて RPL 変更情報がシグナリングされない場合）、あるいは、変更された RPL を使用することを決定することができる（ビットストリームにおいて RPL 変更情報がシグナリングされる場合）。RPL が、デフォルト RPL と比較して変更される（例えば、置換される、調整される）場合、エンコーダ (340) は、次のうちの 1 以上を実行することができる：(a) 参照インデックスによるより効率的なアドレス指定のために参照ピクチャを並べ替える；(b) 符号化中の使用の頻度に少なくとも部分的に基づいて、参照ピクチャを除去する；(c) 符号化中の使用の頻度に少なくとも部分的に基づいて、参照ピクチャを追加する。例えば、エンコーダ (340) は、動き補償のための参照ピクチャの使用が閾量を下回った後、且つ / あるいは他の基準に従って、RPL から所与の参照ピクチャを除去することを決定することができる。別の例として、エンコーダ (340) は、動き補償のための参照ピクチャの使用が閾量を超えた場合、且つ / あるいは他の基準に従って、RPL に所与の参照ピクチャを追加することを決定することができる。別の例として、エンコーダ (340) は、それぞれの参照ピクチャの使用の頻度に基づいて、且つ / あるいは他の基準に従って、RPL 内の参照ピクチャをどのように並べ替えるかを決定することができる。

#### 【 0043 】

動き推定部は、エントロピ符号化される動きベクトル情報等の動き情報を出力する。動き補償部は、動き補償予測値を決定するために、動きベクトルを参照ピクチャに適用する。エンコーダは、ブロックの動き補償予測値と対応するオリジナルの値との間の差（あれば）を決定する。このような予測残差値が、周波数変換、量子化、及びエントロピ符号化を用いて、さらに符号化される。同様に、イントラ予測では、エンコーダ (340) は、ブロックに関するイントラ予測値を決定し、予測残差値を決定し、（周波数変換、量子化、及びエントロピ符号化を用いて）予測残差値を符号化することができる。具体的には、エンコーダ (340) のエントロピ符号化部が、量子化された変換係数値に加えて、所定のサイド情報（例えば、動きベクトル情報、QP 値、モード決定、パラメータ選択、参照インデックス、RPL 変更情報）も圧縮する。一般的なエントロピ符号化技術は、指数ゴロム符号化、算術符号化、差分符号化、ハフマン符号化、ラン・レンゲス符号化、「V2V (variable-length-to-variable-length)」符号化、「V2F (variable-length-to-fixed-length)」符号化、LZ 符号化、辞書符号化、「PIPE (probability interval partitioning entropy coding)」符号化、及びこれらの組合せを含む。エントロピ符号化部は、様々な種類の情報のための様々な符号化技術を使用することができ、特定の符号化技術における複数の符号テーブルの中から選択することができる。

#### 【 0044 】

符号化フレーム (341) 及び MMCO / RPS 情報 (342) が、復号化プロセス・エミュレータ (350) により処理される。復号化プロセス・エミュレータ (350) は、例えば、動き補償においてエンコーダ (340) により使用される参照ピクチャを再構築する復号化タスク等のデコーダの機能の一部を実装している。復号化プロセス・エミュレータ (350) は、所与の符号化フレーム (341) が、符号化される後続フレームのフレーム間予測において参照ピクチャとして使用するために再構築されて記憶される必要があるかどうかを判定するために、MMCO / RPS 情報 (342) を使用する。MMC

10

20

30

40

50

O / R P S 情報 ( 3 4 2 ) により、符号化フレーム ( 3 4 1 ) が記憶される必要があると示される場合、復号化プロセス・エミュレータ ( 3 5 0 ) は、符号化フレーム ( 3 4 1 ) を受信して対応する復号化フレーム ( 3 5 1 ) を生成するデコーダにより行われるであろう復号化プロセスを模擬する。そうする場合、エンコーダ ( 3 4 0 ) が、復号化フレーム記憶領域 ( 3 6 0 ) に記憶された 1 以上の復号化フレーム ( 3 6 9 ) を使用したとき、復号化プロセス・エミュレータ ( 3 5 0 ) は、復号化プロセスの一部として、記憶領域 ( 3 6 0 ) から 1 以上の復号化フレーム ( 3 6 9 ) を使用する。

#### 【 0 0 4 5 】

復号化フレーム一時メモリ記憶領域 ( 3 6 0 ) は、複数のフレーム・バッファ記憶領域 ( 3 6 1 、 3 6 2 、 . . . 、 3 6 n ) を含む。復号化プロセス・エミュレータ ( 3 5 0 ) は、参照ピクチャとして使用するためにエンコーダ ( 3 4 0 ) によりもはや必要とされなくなつたフレームを有する任意のフレーム・バッファ ( 3 6 1 、 3 6 2 等 ) を識別するために、M M C O / R P S 情報 ( 3 4 2 ) を使用して、記憶領域 ( 3 6 0 ) のコンテンツを管理する。復号化プロセスを模擬した後、復号化プロセス・エミュレータ ( 3 5 0 ) は、このように識別されるフレーム・バッファ ( 3 6 1 、 3 6 2 等 ) に、新たに復号化されたフレーム ( 3 5 1 ) を記憶する。10

#### 【 0 0 4 6 】

符号化フレーム ( 3 4 1 ) 及び M M C O / R P S 情報 ( 3 4 2 ) はまた、一時符号化データ領域 ( 3 7 0 ) にバッファされる。符号化データ領域 ( 3 7 0 ) に収集される符号化データは、エレメンタリ符号化ビデオ・ビットストリームのシンタックスの一部として、R P L 変更情報を示すシンタックス要素を含み得る。符号化データ領域 ( 3 7 0 ) に収集される符号化データはまた、( 例えは、1 以上の「 S E I ( supplemental enhancement information ) 」メッセージ又は「 V U I ( video usability information ) 」メッセージにおける 1 以上のパラメータとして、) 符号化ビデオ・データに関連するメディア・メタデータを含み得る。20

#### 【 0 0 4 7 】

一時符号化データ領域 ( 3 7 0 ) からの収集されたデータ ( 3 7 1 ) は、チャネル・エンコーダ ( 3 8 0 ) により処理される。チャネル・エンコーダ ( 3 8 0 ) は、( 例えは、I S O / I E C 1 4 4 9 6 - 1 2 等のメディア・コンテナ・フォーマットに従って ) メディア・ストリームとして伝送するために、収集されたデータをパケット化することができる。そのような場合、チャネル・エンコーダ ( 3 8 0 ) は、メディア伝送ストリームのシンタックスの一部として、シンタックス要素を付加することができる。あるいは、チャネル・エンコーダ ( 3 8 0 ) は、( 例えは、I S O / I E C 1 4 4 9 6 - 1 2 等のメディア・コンテナ・フォーマットに従って ) ファイルとして記憶するために、収集されたデータを編成することができる。そのような場合、チャネル・エンコーダ ( 3 8 0 ) は、メディア記憶ファイルのシンタックスの一部として、シンタックス要素を付加することができる。あるいは、より一般的に、チャネル・エンコーダ ( 3 8 0 ) は、1 以上のメディア・システム多重化プロトコル又は伝送プロトコルを実装することができる。そのような場合、チャネル・エンコーダ ( 3 8 0 ) は、1 以上のプロトコルのシンタックスの一部として、シンタックス要素を付加することができる。チャネル・エンコーダ ( 3 8 0 ) は、チャネル ( 3 9 0 ) への出力を提供する。チャネル ( 3 9 0 ) は、ストレージ、通信コネクション、又は出力のための別のチャネルを表す。3040

#### 【 0 0 4 8 】

##### I V . 例示的なデコーダ・システム

図 4 は、いくつかの説明する実施形態を協働して実装することができる例示的なデコーダ・システム ( 4 0 0 ) のブロック図である。デコーダ・システム ( 4 0 0 ) は、リアルタイム通信のための低レイテンシ復号化モード及びファイル又はストリームからのメディア再生のための通常の復号化モード等の複数の復号化モードのうちいずれかで動作することができる汎用復号化ツールであってもよいし、そのような 1 つの符号化モードのために適合された専用復号化ツールであってもよい。デコーダ・システム ( 4 0 0 ) は、オペレ50

ーティング・システム・モジュールとして、アプリケーション・ライブラリの一部として、又はスタンドアロン・アプリケーションとして、実装することができる。概して、デコーダ・システム(400)は、チャネル(410)から符号化データを受信し、出力先(490)への出力として再構築フレームを生成する。符号化データは、RPL変更情報を示すシンタックス要素を含み得る。

#### 【0049】

デコーダ・システム(400)は、ストレージ、通信コネクション、又は入力としての符号化データのための別のチャネルを表し得るチャネル(410)を含む。チャネル(410)は、チャネル符号化された符号化データを生成する。チャネル・デコーダ(420)は、符号化データを処理することができる。例えば、チャネル・デコーダ(420)は、(例えば、ISO/IEC 14496-12等のメディア・コンテナ・フォーマットに従って)メディア・ストリームとして伝送するために収集されたデータを脱パケット化する。そのような場合、チャネル・デコーダ(420)は、メディア伝送ストリームのシンタックスの一部として付加されたシンタックス要素を解析することができる。あるいは、チャネル・デコーダ(420)は、(例えば、ISO/IEC 14496-12等のメディア・コンテナ・フォーマットに従って)ファイルとして記憶するために収集された符号化ビデオ・データを分離する。そのような場合、チャネル・デコーダ(420)は、メディア記憶ファイルのシンタックスの一部として付加されたシンタックス要素を解析することができる。あるいは、より一般的に、チャネル・デコーダ(420)は、1以上のメディア・システム逆多重化プロトコル又は伝送プロトコルを実装することができる。そのような場合、チャネル・デコーダ(420)は、1以上のプロトコルのシンタックスの一部として付加されたシンタックス要素を解析することができる。10  
20

#### 【0050】

十分な量のデータが受信されるまで、チャネル・デコーダ(420)から出力される符号化データ(421)は、一時符号化データ領域(430)に記憶される。符号化データ(421)は、符号化フレーム(431)及びMMCO/RPS情報(432)を含む。符号化データ領域(430)内の符号化データ(421)は、エレメンタリ符号化ビデオ・ビットストリームのシンタックスの一部として、RPL変更情報を示すシンタックス要素を含み得る。符号化データ領域(430)内の符号化データ(421)はまた、(例えば、1以上のSEIメッセージ又はVUIメッセージにおける1以上のパラメータとして、)符号化ビデオ・データに関連するメディア・メタデータを含み得る。一般に、このような符号化データ(421)がデコーダ(450)により使用されるまで、符号化データ領域(430)は、符号化データ(421)を一時的に記憶する。その時点で、符号化フレーム(431)及びMMCO/RPS情報(432)の符号化データが、符号化データ領域(430)からデコーダ(450)に伝送される。復号化が進むにつれ、新たな符号化データが、符号化データ領域(430)に追加され、符号化データ領域(430)に残っている最も古い符号化データが、デコーダ(450)に伝送される。30

#### 【0051】

デコーダ(450)は、対応する復号化フレーム(451)を生成するために、符号化フレーム(431)を定期的に復号化する。必要に応じて、復号化プロセスを実行するとき、デコーダ(450)は、フレーム間予測のための参照フレーム(参照ピクチャ)として、1以上の以前に復号化されたフレーム(469)を使用することができる。デコーダ(450)は、復号化フレーム一時メモリ記憶領域(460)から、そのような以前に復号化されたフレーム(469)を読み出す。一般に、デコーダ(450)は、エントロピ復号化、逆量子化、逆周波数変換、及び動き補償(これが、RPL変更情報を用いて1以上のRPLを作成し得る)等の復号化タスクを実行する複数の復号化モジュールを含む。デコーダ(450)により実行される正確なオペレーションは、圧縮フォーマットに応じて変わり得る。40

#### 【0052】

例えば、デコーダ(450)は、圧縮フレーム又は一連のフレームの符号化データを受50

信し、復号化フレーム(451)を含む出力を生成する。デコーダ(450)において、バッファは、圧縮フレームの符号化データを受け入れ、受け入れた符号化データをエントロピ復号化部に利用可能にする。エントロピ復号化部は、通常はエンコーダにおいて実行されたエントロピ符号化の逆を適用することにより、エントロピ符号化された量子化データに加えて、エントロピ符号化されたサイド情報(参照インデックス、RPL変更情報等を含む)もエントロピ復号化する。デコーダは、参照ピクチャの1以上のRPLを構築する。ここで、参照インデックスは、1以上のRPL内の参照ピクチャをアドレス指定する。適切なときに新たな参照ピクチャが追加されるように、適切なときに動き補償のためにもはや使用されないより古い参照ピクチャが除去されるように、及び適切なときに参照ピクチャが並び替えられるように、1以上のRPLが構築される。いくつかの実施例において、例えば、現ピクチャを復号化するとき、デコーダ(450)は、復号化フレーム記憶領域(460)内の参照ピクチャを含むRPSを決定し、次いで、現ピクチャの所与のスライスを復号化するための1以上のRPLを作成する。RPLは、RPSからの利用可能な参照ピクチャの選択に関するルールを適用することにより作成され得る。この場合、RPL変更情報は、ビットストリームから解析されない。あるいは、RPLは、RPSから利用可能な特定の参照ピクチャを選択することにより作成され得る。この場合、選択される参照ピクチャは、ビットストリームから解析されるRPL変更情報内に示される。暗黙的アプローチのルールにより構築されるRPLと比較すると、RPL変更情報は、RPS内の参照ピクチャのリストとして、置換RPLを指定することができる。代替的に、RPL変更情報は、よりきめの細かい形で、暗黙的アプローチのルールにより構築されたRPLからの1以上の参照ピクチャの除去、暗黙的アプローチのルールにより構築されたRPLへの1以上の参照ピクチャの追加、及び/又は暗黙的アプローチのルールにより構築されたRPL内の参照ピクチャの並べ替えを指定してもよい。10

#### 【0053】

動き補償部は、再構築されているフレームのサブ・ブロック及び/又はブロック(一般に、ブロック)の動き補償予測を形成するために、動き情報を1以上の参照ピクチャに適用する。イントラ予測モジュールは、隣接する以前に再構築されたサンプル値から、現ブロックのサンプル値を空間的に予測することができる。デコーダ(450)はまた、予測残差を再構築する。逆量子化部は、エントロピ復号化されたデータを逆量子化する。逆周波数変換部は、再構築された周波数領域データを空間領域情報に変換する。予測フレームに関して、デコーダ(450)は、再構築フレームを形成するために、再構築された予測残差を動き補償予測と結合する。デコーダ(450)は、同様に、予測残差を、イントラ予測からの空間予測と結合することができる。ビデオ・デコーダ(450)における動き補償ループは、適応デブロッキング・フィルタを含み、適応デブロッキング・フィルタは、復号化フレーム(451)内のブロック境界ロー及び/又はカラムにわたる不連続さを平滑化する。30

#### 【0054】

復号化フレーム一時メモリ記憶領域(460)は、複数のフレーム・バッファ記憶領域(461、462、...、46n)を含む。復号化フレーム記憶領域(460)はDPBの一例である。デコーダ(450)は、復号化フレーム(451)を記憶することができるフレーム・バッファ(461、462等)を識別するために、MMCO/RPS情報(432)を使用する。デコーダ(450)は、そのフレーム・バッファに復号化フレーム(451)を記憶する。40

#### 【0055】

出力シーケンサ(480)は、出力順で生成される次のフレームが復号化フレーム記憶領域(460)内で利用可能になるときを識別するために、MMCO/RPS情報(432)を使用する。出力順で生成される次のフレーム(481)が、復号化フレーム記憶領域(460)内で利用可能になったとき、そのフレームが、出力シーケンサ(480)により読み出され、出力先(490)(例えば、ディスプレイ)に出力される。一般に、復号化フレーム記憶領域(460)から出力シーケンサ(480)によりフレームが出力さ50

れる順番は、デコーダ(450)によりフレームが復号化される順番とは異なり得る。

**【0056】**

V. 例示的なビデオ・エンコーダ

図5は、いくつかの説明する実施形態を協働して実装することができる一般化されたビデオ・エンコーダ(500)のブロック図である。エンコーダ(500)は、現フレーム(505)を含む一連のビデオ・フレームを受信し、出力として符号化データ(595)を生成する。

**【0057】**

エンコーダ(500)は、ブロックベースであり、実装に依存するブロック・フォーマットを使用する。ブロックは、例えば、周波数変換段階及びエントロピ符号化段階において等の様々な段階において、さらに分割され得る。例えば、フレームは、 $64 \times 64$  ブロック、 $32 \times 32$  ブロック、又は $16 \times 16$  ブロックに分割され得、今度は、それらのブロックが、符号化及び復号化のために、ピクセル値のより小さなブロック及びサブ・ブロックに分割され得る。

10

**【0058】**

エンコーダ・システム(500)は、予測フレーム及びイントラ符号化されたフレームを圧縮する。プレゼンテーションの目的上、図5は、フレーム内符号化のためのエンコーダ(500)を介した「イントラ・パス」と、フレーム間符号化のための「インタ・パス」とを示している。エンコーダ(500)のコンポーネントの多くが、フレーム内符号化及びフレーム間符号化の両方のために使用される。これらのコンポーネントにより実行される正確なオペレーションは、圧縮される情報のタイプに応じて変わり得る。

20

**【0059】**

現フレーム(505)が予測フレームである場合、動き推定部(510)は、1以上の参照フレーム(参照ピクチャ)に関して、現フレーム(505)のピクセル値のブロック、サブ・ブロック、又は他のセットの動きを推定する。フレーム・ストア(520)は、参照フレーム(参照ピクチャ)として使用するために、1以上の再構築された以前のフレーム(525)をバッファする。複数の参照ピクチャが使用される場合、複数の参照ピクチャは、異なる時間的方向からのものであってもよいし、同じ時間的方向からのものであってもよい。複数の参照ピクチャは、1以上のRPL内に表され得、参照インデックスによりアドレス指定される。動き推定部(510)は、サイド情報として、差分動きベクトル情報、参照インデックス、及びRPL変更情報等の動き情報(515)を出力する。符号化中、エンコーダ(500)は、適切なときに新たな参照ピクチャが追加されるよう、適切なときに動き補償のためにはや使用されないより古い参照ピクチャが除去されるように、及び適切なときに参照ピクチャが1以上のRPL内で並び替えられるように、1以上のRPLを構築する。

30

**【0060】**

いくつかの実施例において、現フレームを符号化するとき、エンコーダ(500)は、フレーム・ストア(520)内の参照フレームを含むRPSを決定する。エンコーダ(500)は、通常、フレームの最初のスライスのRPSを決定する。スライスごとに、エンコーダ(500)は、現フレームの所与のスライスを符号化するための1以上のRPLを作成する。RPLを作成するために、エンコーダ(500)は、RPSからの利用可能な参照フレームの選択に関するルールを適用することができる。この場合、RPL変更情報は、符号化データ(595)において明示的にシグナリングされない。あるいは、RPLを作成するために、エンコーダ(500)は、RPSから利用可能な特定の参照フレームを選択することができる。この場合、選択される参照フレームは、符号化データ(595)においてシグナリングされるRPL変更情報内に示される。暗黙的アプローチのルールにより構築されるRPLと比較すると、RPL変更情報は、RPS内の参照ピクチャのリストとして、置換RPLを指定することができる。代替的に、RPL変更情報は、よりきめの細かい形で、ルールにより暗黙的に構築されたRPLからの1以上の参照フレームの除去、ルールにより暗黙的に構築されたRPLへの1以上の参照フレームの追加、及び/

40

50

又はルールにより暗黙的に構築された R P L 内の参照フレームの並べ替えを指定してもよい。

#### 【 0 0 6 1 】

動き補償部 ( 5 3 0 ) は、動き補償された現フレーム ( 5 3 5 ) を形成するとき、再構築された動きベクトルを、1 以上の再構築された参照フレーム ( 5 2 5 ) に適用する。動き補償された現フレーム ( 5 3 5 ) のサブ・ブロック、ブロック等とオリジナルの現フレーム ( 5 0 5 ) の対応する部分との間の差 ( あれば ) が、サブ・ブロック、ブロック等の予測残差 ( 5 4 5 ) である。オリジナルの現フレーム ( 5 0 5 ) により近い再構築フレームを得るために、現フレームの後の再構築の間に、再構築された予測残差が、動き補償された現フレーム ( 5 3 5 ) に付加される。しかしながら、不可逆圧縮 ( lossy compression ) では、それでも、いくつかの情報が、オリジナルの現フレーム ( 5 0 5 ) から失われる。イントラ・パスは、イントラ予測モジュール ( 図示せず ) を含み得る。イントラ予測モジュールは、隣接する以前に再構築されたピクセル値から、現ブロック又はサブ・ブロックのピクセル値を空間的に予測する。  
10

#### 【 0 0 6 2 】

周波数変換部 ( 5 6 0 ) は、空間領域ビデオ情報を周波数領域 ( すなわち、スペクトル変換 ) データに変換する。ブロックベースのビデオ・フレームに関して、周波数変換部 ( 5 6 0 ) は、離散コサイン変換とその整数近似、又は別のタイプの前方ブロック変換をピクセル値データ又は予測残差データのブロック又はサブ・ブロックに適用して、周波数変換係数のブロック / サブ・ブロックを生成する。次いで、量子化部 ( 5 7 0 ) は、変換係数を量子化する。例えば、量子化部 ( 5 7 0 ) は、フレームごとに、スライスごとに、ブロックごとに、又は他の単位で変わるステップ・サイズで、一様でないスカラ量子化を周波数領域データに適用する。  
20

#### 【 0 0 6 3 】

後続の動き推定 / 動き補償のために現フレームの再構築されたバージョンが必要とされる場合、逆量子化部 ( 5 7 6 ) が、量子化された周波数係数データに対して逆量子化を実行する。逆周波数変換部 ( 5 6 6 ) は、逆周波数変換を実行し、再構築された予測残差又はピクセル値のブロック / サブ・ブロックを生成する。予測フレームに関して、エンコーダ ( 5 0 0 ) は、参照ピクチャとして使用され得る再構築フレーム ( 5 0 5 ) を形成するために、再構築された予測残差 ( 5 4 5 ) を動き補償予測 ( 5 3 5 ) と結合する。( 図 5 には示されていないが、イントラ・パスにおいては、エンコーダ ( 5 0 0 ) は、参照ピクチャとして使用されるフレームを再構築するために、予測残差を、イントラ予測からの空間予測と結合することができる。) フレーム・ストア ( 5 2 0 ) は、後続の動き補償予測において参照ピクチャとして使用するために、再構築された現フレームをバッファする。  
30

#### 【 0 0 6 4 】

エンコーダ ( 5 0 0 ) における動き補償ループは、フレーム・ストア ( 5 2 0 ) の前又は後に、適応ループ内デブロック・フィルタ ( 5 1 0 ) を含む。エンコーダ ( 5 0 0 ) は、フレーム内の境界にわたる不連続さを適応的に平滑化するために、ループ内フィルタリングを再構築フレームに適用する。  
40

#### 【 0 0 6 5 】

エントロピ符号化部 ( 5 8 0 ) は、量子化部 ( 5 7 0 ) の出力に加えて、動き情報 ( 5 1 5 ) 及び所定のサイド情報 ( 例えば、Q P 値、参照インデックス、R P L 変更情報 ) も圧縮する。エントロピ符号化部 ( 5 8 0 ) は、符号化データ ( 5 9 5 ) をバッファ ( 5 9 0 ) に供給し、バッファ ( 5 9 0 ) は、符号化データを出力ビットストリームに多重化する。符号化データ ( 5 9 5 ) は、R P L 変更情報を示すシンタックス要素を含み得る。セクション V I I において、そのようなシンタックス要素の例を説明する。

#### 【 0 0 6 6 】

コントローラ ( 図示せず ) は、エンコーダの様々なモジュールから入力を受信する。コントローラは、符号化中、中間結果を評価し、例えば、Q P 値を設定して、レート - 歪み解析を実行する。コントローラは、他のモジュールと協働して、符号化中、符号化パラメ  
50

ータを設定し変更する。具体的には、1以上のRPLを変更する（例えば、置換する、調整する）かどうか及び1以上のRPLをどのように変更する（例えば、置換する、調整する）かを決定するとき、コントローラは、どの参照ピクチャが1以上のRPLに追加されるかを制御し、どのピクチャが1以上のRPLから除去されるかを制御し、参照インデックスによるより効率的なアドレス指定のために、1以上のRPL内の参照ピクチャを並び替えることができる。コントローラは、例えば、シーン・チェンジの後全ての参照ピクチャを除去することにより、IDRピクチャ等の特別な種類のピクチャの符号化の後全ての参照ピクチャを除去することにより、動き補償のための参照ピクチャの使用が閾量を下回った後所与の参照ピクチャを除去することにより、且つ／あるいは他の基準に従って参照ピクチャを除去することにより、RPSから（したがって、RPLから）参照ピクチャを除去することを決定することができる。コントローラは、例えば、ピクチャ・タイプ／ピクチャにおけるスライス・タイプに従って、ピクチャのための一時的なレイヤに従って、且つ／あるいは他の基準に従ってピクチャを追加することにより、RPSに参照ピクチャを追加することを決定することができる。RPLに関して、コントローラは、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従ってRPLが変更されない動き補償の結果を評価することができ、また、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従ってRPLが変更される動き補償の結果（又は、RPLを変更する複数の異なる方法の結果）を評価することができる。コントローラは、ビット・レート及び／又は品質の観点で結果を評価することができる。コントローラは、ルールにより暗黙的に構築されたRPLを選択してもよいし（RPL変更情報がない場合）、あるいは、変更されたRPLを選択してもよい（RPL変更情報により指定される場合）。RPLを変更する（例えば、置換する、調整する）ために、暗黙的に構築されたRPLと比較すると、コントローラは、（a）参照インデックスによるより効率的なアドレス指定のために、参照ピクチャを並び替えることができる、（b）符号化中の使用の頻度に少なくとも部分的に基づいて、参照ピクチャを除去することができる、且つ／あるいは（c）符号化中の使用の頻度に少なくとも部分的に基づいて、参照ピクチャを追加することができる。例えば、コントローラは、動き補償のための参照ピクチャの使用が閾量を下回った後、且つ／あるいは他の基準に従って、RPLから所与の参照ピクチャを除去することを決定することができる。あるいは、コントローラは、動き補償のための参照ピクチャの使用が閾量を超えた場合、且つ／あるいは他の基準に従って、RPLに所与の参照ピクチャを追加することを決定することができる。あるいは、コントローラは、それぞれの参照ピクチャの使用の頻度に基づいて、且つ／あるいは他の基準に従って、RPL内の参照ピクチャをどのように並べ替えるかを決定することができる。コントローラは、ピクチャごとに、スライスごとに、又は何らかの他の単位で、1以上のRPLを構築することができる。

#### 【0067】

実装及び所望の圧縮のタイプに応じて、エンコーダのモジュールを追加してもよいし、省略してもよいし、複数のモジュールに分割してもよいし、他のモジュールと結合してもよいし、且つ／又は同様のモジュールと置換してもよい。代替実施形態において、異なるモジュール及び／又はモジュールの他の構成を有するエンコーダは、説明した技術のうちの1以上を実行する。エンコーダの特定の実施形態は、通常、エンコーダ（500）の変形又は補完バージョンを使用する。エンコーダ（500）内のモジュール間の示された関係は、エンコーダ内の情報の一般的な流れを示すものである。他の関係は、簡潔さのため示されていない。

#### 【0068】

##### V I . 例示的なビデオ・デコーダ

図6は、いくつかの説明する実施形態を協働して実装することができる一般化されたデコーダ（600）のブロック図である。デコーダ（600）は、圧縮フレーム又は一連のフレームの符号化データ（695）を受信し、参照ピクチャとして使用され得る再構築フレーム（605）を含む出力を生成する。プレゼンテーションの目的上、図6は、フレー

10

20

30

40

50

ム内復号化のためのデコーダ(600)を介した「イントラ・パス」と、フレーム間復号化のための「インタ・パス」とを示している。デコーダ(600)のコンポーネントの多くが、フレーム内復号化及びフレーム間復号化の両方のために使用される。これらのコンポーネントにより実行される正確なオペレーションは、伸張される情報のタイプに応じて変わり得る。

#### 【0069】

バッファ(690)は、圧縮フレームの符号化データ(695)を受け入れ、受け入れた符号化データを解析部／エントロピ復号化部(680)に利用可能にする。符号化データ(695)は、RPL変更情報を示すシンタックス要素を含み得る。セクションVIIにおいて、そのようなシンタックス要素の例を説明する。解析部／エントロピ復号化部(680)は、通常はエンコーダにおいて実行されたエントロピ符号化の逆を適用することにより、エントロピ符号化された量子化データに加えて、エントロピ符号化されたサイド情報(参照インデックス、RPL変更情報等を含む)もエントロピ復号化する。

10

#### 【0070】

復号化中、適切なときに新たな参照ピクチャが追加されるように、適切なときに動き補償のためにもはや使用されないより古い参照ピクチャが除去されるように、及び適切なときに参照ピクチャが並び替えられるように、デコーダ(600)は、1以上のRPLを構築する。デコーダ(600)は、1以上のRPLに関する利用可能な情報(例えば、RPS内の利用可能な参照ピクチャ)に基づいて、ルールに従った変更に基づいて、且つ/あるいは符号化データ(695)の一部としてシグナリングされる変更に従って、1以上のRPLを構築することができる。いくつかの実施例において、例えば、現フレームを復号化するとき、デコーダ(600)は、フレーム・ストア(620)内の参照フレームを含むRPSを決定する。デコーダ(600)は、通常、フレームの最初のスライスのRPSを決定する。スライスごとに、デコーダ(600)は、現フレームの所与のスライスを復号化するための1以上のRPLを作成する。RPLを作成するために、場合によっては(符号化データ(695)において示されるように)、デコーダ(600)は、RPSからの利用可能な参照フレームの選択に関するルールを適用する。この場合、RPL変更情報は、符号化データ(695)から解析されない。他の場合には、RPLを作成するために、デコーダ(600)は、RPSから利用可能な特定の参照フレームを選択する。この場合、選択される参照フレームは、符号化データ(695)から解析されるRPL変更情報内に示される。RPL変更情報は、RPS内の参照ピクチャのリストとして、置換RPLを指定することができる。代替的に、RPL変更情報は、よりきめの細かい形で、ルールにより暗黙的に構築されたRPLからの1以上の参照フレームの除去、ルールにより暗黙的に構築されたRPLへの1以上の参照フレームの追加、及び/又はルールにより暗黙的に構築されたRPL内の参照フレームの並べ替えを指定してもよい。

20

#### 【0071】

動き補償部(630)は、再構築されているフレーム(605)のサブ・ブロック及び/又はブロックの動き補償予測(635)を形成するために、動き情報(615)を1以上の参照ピクチャ(625)に適用する。フレーム・ストア(620)は、参照ピクチャとして使用するために、1以上の以前に再構築されたフレームを記憶する。

30

#### 【0072】

イントラ・パスは、イントラ予測モジュール(図示せず)を含み得る。イントラ予測モジュールは、隣接する以前に再構築されたピクセル値から、現ブロック又はサブ・ブロックのピクセル値を空間的に予測する。インタ・パスにおいて、デコーダ(600)は、予測残差を再構築する。逆量子化部(670)は、エントロピ復号化されたデータを逆量子化する。逆周波数変換部(660)は、再構築された周波数領域データを空間領域情報に変換する。例えば、逆周波数変換部(660)は、逆ブロック変換を周波数変換係数に適用して、ピクセル値データ又は予測残差データを生成する。逆周波数変換は、逆離散コサイン変換とその整数近似、又は別のタイプの逆周波数変換であり得る。

40

#### 【0073】

50

予測フレームに関して、デコーダ(600)は、参照ピクチャとして使用され得る再構築フレーム(605)を形成するために、再構築された予測残差(645)を動き補償予測(635)と結合する。(図6には示されていないが、イントラ・パスにおいては、デコーダ(600)は、参照ピクチャとして使用され得るフレームを再構築するために、予測残差を、イントラ予測からの空間予測と結合することができる。)デコーダ(600)における動き補償ループは、フレーム・ストア(620)の前又は後に、適応ループ内デブロック・フィルタ(610)を含む。デコーダ(600)は、フレーム内の境界にわたる不連続さを適応的に平滑化するために、ループ内フィルタリングを再構築フレームに適用する。

## 【0074】

10

図6において、デコーダ(600)はまた、後処理デブロック・フィルタ(608)を含む。後処理デブロック・フィルタ(608)は、任意的に、再構築フレーム内の不連続さを平滑化する。(デリング・フィルタリング等の)他のフィルタリングも、後処理フィルタリングの一部として適用することができる。

## 【0075】

実装及び所望の伸張のタイプに応じて、デコーダのモジュールを追加してもよいし、省略してもよいし、複数のモジュールに分割してもよいし、他のモジュールと結合してもよいし、且つ／又は同様のモジュールと置換してもよい。代替実施形態において、異なるモジュール及び／又はモジュールの他の構成を有するデコーダは、説明した技術のうちの1以上を実行する。デコーダの特定の実施形態は、通常、デコーダ(600)の変形又は補完バージョンを使用する。デコーダ(600)内のモジュール間の示された関係は、デコーダ内の情報の一般的な流れを示すものである。他の関係は、簡潔さのため示されていない。

20

## 【0076】

## VII. 参照ピクチャ・リスト変更情報のシグナリング

このセクションは、RPL変更情報のシグナリングの領域における様々なイノベーションを提供する。いくつかの状況において、これらのイノベーションは、RPL変更情報のシンタックス要素のより効率的なシグナリングをもたらす。

## 【0077】

## A. 参照ピクチャ及びRPL

30

参照ピクチャは、一般に、他のピクチャの復号化プロセスにおけるピクチャ間予測のために使用され得るサンプルを含むピクチャである。通常、他のピクチャは、復号化の順番において、参照ピクチャに続く。動き補償予測に使用するために、所与の時間において、複数の参照ピクチャが利用可能であり得る。

## 【0078】

一般に、参照ピクチャ・リスト(「RPL」)は、動き補償予測のために使用される参照ピクチャのリストである。RPL内の参照ピクチャは、参照インデックスによりアドレス指定される。参照インデックスは、RPL内の参照ピクチャを特定する。符号化及び復号化中、RPSが構築されるとき、新たに復号化されたピクチャを追加するために、参照ピクチャとしてもはや使用されないより古いピクチャを除去するために、且つ／あるいはより一般に使用される参照インデックスのシグナリングをより効率的にするようRPL内の参照ピクチャを並び替えるために、RPL内の参照ピクチャは、しばしば変わり得る。エンコーダ及びデコーダは、1以上のRPLを構築する、変更する等の同一のルールに従い得る。そのようなルールに加えて(あるいは、そのようなルールに代えて)、エンコーダは、エンコーダにより使用される1以上のRPLを合致させるために、デコーダがどのように1以上のRPLを構築すべきか、変更すべきか等を示す情報をデコーダにシグナリングすることができる。通常、RPLは、RPLに関する利用可能な情報(例えば、RPS内の利用可能なピクチャ)、ルールに従った変更、及び／又はビットストリームにおいてシグナリングされる変更に基づいて、符号化及び復号化中に構築される。

40

## 【0079】

50

いくつかの実施例において、現ピクチャに関して、エンコーダ又はデコーダは、復号化ピクチャ・バッファ（「D P B」）等の復号化フレーム記憶領域内の参照ピクチャを含む参照ピクチャ・セット（「R P S」）を決定する。R P Sは、現ピクチャ及び今後符号化されるピクチャの復号化プロセスにおいて使用される参照ピクチャを記載したものである。R P Sに含まれる参照ピクチャは、ビットストリームにおいて明示的にリストされる。

#### 【0080】

エンコーダ又はデコーダは、1ピクチャにつき1回R P Sを決定する。例えば、デコーダは、スライス・ヘッダにおいてシグナリングされるシンタックス要素を用いて、ピクチャのスライスのスライス・ヘッダを復号化した後、R P Sを決定する。参照ピクチャは、ピクチャ・オーダ・カウント（「P O C」）値、その一部、及び／又はビットストリームにおいてシグナリングされる他の情報により特定される。エンコーダ又はデコーダは、現ピクチャのピクチャ間予測において使用され得る、（且つ復号化の順番において現ピクチャに続くピクチャのうち1以上のピクチャ間予測において使用され得る、）ショート・ターム参照ピクチャ及びロング・ターム参照ピクチャのグループを決定する。（エンコーダ又はデコーダはまた、復号化の順番において現ピクチャに続くピクチャのうち1以上のピクチャ間予測において使用され得るが、現ピクチャのためには使用されない参照ピクチャのグループを決定する。）集合的に、参照ピクチャのそのグループが、現ピクチャのR P Sである。

#### 【0081】

現ピクチャの所与のスライスについて、エンコーダ又はデコーダは、1以上のR P Lを作成する。エンコーダ又はデコーダは、現ピクチャのピクチャ間予測において使用され得る、ショート・ターム参照ピクチャ及びロング・ターム参照ピクチャのグループを結合することにより、一時的なバージョンのR P L（例えば、R P L 0又はR P L 1）を作成する。「暗黙的」アプローチのルールに従ってR P Lを構築するために、エンコーダ又はデコーダは、一時的なバージョンのR P L内の参照ピクチャを使用してもよいし、一時的なバージョンのR P L内の参照ピクチャの一部（例えば、一時的なバージョンのR P L内の最初の×個のピクチャ）のみを使用してもよい。「暗黙的」アプローチでは、R P L変更情報は、ビットストリームにおいてシグナリングされず、ビットストリームから解析されない。「明示的」アプローチでは、R P Lを構築するために、エンコーダ又はデコーダは、ビットストリームにおいてシグナリングされたR P L変更情報／ビットストリームから解析されたR P L変更情報を使用して、一時的なバージョンのR P Lから特定の参照ピクチャを選択する。「暗黙的」アプローチのルールにより構築されるR P Lと比較すると、R P L変更情報は、R P Lからの1以上の参照ピクチャの除去、R P Lへの1以上の参照ピクチャの追加、及び／又はR P L内の参照ピクチャの並べ替えを指定することができる。

#### 【0082】

代替的に、エンコーダ又はデコーダは、参照ピクチャからR P Lを作成するために、別のアプローチを使用してもよい。

#### 【0083】

##### B . R P L変更フラグの条件付きシグナリング

本明細書で説明するイノベーションの一態様に従うと、エンコーダは、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従ってR P Lが変更されるかどうかを示すフラグを、条件付きでシグナリングする。対応するデコーダは、そのようなフラグを、条件付きで解析する。

#### 【0084】

いくつかの例示的な実施例において、そのフラグは、ref\_pic\_list\_modification\_flag\_l0又はref\_pic\_list\_modification\_flag\_l1である（一般に、フラグは、ref\_pic\_list\_modification\_flag\_lXであり、Xは0又は1であり得る）。フラグref\_pic\_list\_modification\_flag\_lXの値が1に等しい場合、R P L\_Xは、list\_entry\_lX[i]値のリストとして、明示的に指定される（再度、Xは0又は1である）。フラグref\_pic\_list\_modificatio

10

20

30

40

50

`n_flag_IX`の値が0に等しい場合、`RPL_X`は、暗黙的に決定される。`ref_pic_list_modification_flag_IX`が存在しない場合、0に等しいと推定される。

#### 【0085】

図7aは、例示的な実施例における`ref_pic_lists_modification()`シンタックス構造の例示的なシンタックス(700)を示している。この構造は、スライス・ヘッダの一部としてシグナリングされ得る。例示的なシンタックス(700)において、`NumPocTotalCurr`が1より大きいときのみ、`ref_pic_list_modification_flag_IX`が送信される。`NumPocTotalCurr`は、現在の符号化又は復号化のために適用可能な参照ピクチャの総数を示す変数である。符号化又は復号化の例示的な実施例において、変数`NumPocTotalCurr`が、現ピクチャのスライスに関して導出されるとき、その変数は、現ピクチャの符号化又は復号化のための参照ピクチャとして使用される、ショート・ターム参照ピクチャ及びロング・ターム参照ピクチャのカウントを示す。  
10

#### 【0086】

図7aに示されるように、`ref_pic_list_modification_flag_IX`の条件付きシグナリングは、変数`NumPocTotalCurr`の値に依存する。`NumPocTotalCurr`が1以下の場合、`RPL`の変更の可能性はなく、したがって、フラグ`ref_pic_list_modification_flag_IX`を送信する必要はない。条件が満たされた場合、この条件付きシグナリングは、全てのスライスにつき1つ又は2つのフラグを節約することができる。図7aにおける変更は、フラグ`ref_pic_list_modification_flag_IX`がシグナリングされるかどうかに関する条件「if(`NumPocTotalCurr>1`)」を含む。この条件は、(Pスライス又はBスライスの)list 0及び  
20 /又は(Bスライスの)list 1についてチェックされ得る。

#### 【0087】

代替的に、1以上の`RPL`変更フラグを含む`RPL`変更構造(例えば、`ref_pic_lists_modification()`構造)のシグナリング及び解析は、スライス・ヘッダ処理の一部として、又は他の方法で条件を評価することにより制御されてもよい。図7bは、この条件に基づくフラグ`ref_pic_list_modification_flag_I0`及びフラグ`ref_pic_list_modification_flag_I1`の条件付きシグナリング及び解析のアプローチを示している。具体的には、図7bは、`ref_pic_lists_modification()`シンタックス構造を含み得るスライス・シンタックス構造の例示的なシンタックス(750)を示している。`ref_pic_lists_modification()`シンタックス構造は、図7cのシンタックス(760)内に示されている。スライス・ヘッダの例示的なシンタックス(750)では、フラグ`lists_modification_present_flag`が、スライスに対して適用されるピクチャ・パラメータ・セットにおいてシグナリングされる。`lists_modification_present_flag`が0に等しい場合、構造`ref_pic_lists_modification()`は、スライス・ヘッダ内に存在しない。`lists_modification_present_flag`が1に等しい場合、構造`ref_pic_lists_modification()`は、変数`NumPocTotalCurr`の値に応じて、スライス・ヘッダ内に存在し得る。変数`NumPocTotalCurr`が1より大きい場合、図7cのシンタックス(760)内に示される`ref_pic_lists_modification()`構造がシグナリングされる。そうでなければ(変数`NumPocTotalCurr`が1以下の場合)、`ref_pic_lists_modification()`構造はシグナリングされず、リスト・エントリの値が推定される。  
30

#### 【0088】

図7a～図7c、図8、及び図9において、「`u(n)`」という語は、nビットを用いる符号なし整数を表している。(「`u(v)`」のように) nが「v」である場合、ビットの数は、他のシンタックス要素の値に応じて変わる。`u(n)`の解析プロセスは、最上位ビットが最初に書き込まれる符号なし整数のバイナリ表現としてnビットを読み取る関数の戻り値により指定され得る。

#### 【0089】

##### C. リスト・エントリのシンタックス要素のシグナリング

本明細書で説明するイノベーションの別の態様に従うと、エンコーダは、`RPL`をどのように変更するかを示すリスト・エントリのシンタックス要素を条件付きでシグナリングする。対応するデコーダは、そのようなシンタックス要素を条件付きで解析する。  
50

**【 0 0 9 0 】**

いくつかの例示的な実施例において、そのようなシンタックス要素は、R P L 0 の list\_entry\_I0[i] シンタックス要素又は R P L 1 の list\_entry\_I1[i] シンタックス要素に関するものである（一般に、シンタックス要素は、list\_entry\_IX であり、X は 0 又は 1 であり得る）。図 8 は、ref\_pic\_lists\_modification() シンタックス構造の例示的なシンタックス（800）を示しており、これは、スライス・ヘッダの一部としてシグナリングされ得る。例示的なシンタックス（800）において、シンタックス要素 list\_entry\_IX[0] が、ビットストリームにおいて条件付きでシグナリングされる。詳細には、NumPocTotalCurr が 2 に等しく、num\_ref\_idx\_IX\_active\_minus1 が 0 に等しい場合、シンタックス要素 list\_entry\_IX[0] は、ビットストリームにおいてシグナリングされない。変数 num\_ref\_idx\_IX\_active\_minus1 は、スライスを復号化するために使用することができる R P L X の最大参照インデックスを示す。変数 num\_ref\_idx\_IX\_active\_minus1 は、デフォルト値（適用可能なピクチャ・パラメータ・セットにおいて示されるように、例えば、0 . . . 15 からの値）を有し得るか、あるいは、num\_ref\_idx\_IX\_active\_minus1 は、現スライスについてスライス・ヘッダにおいてシグナリングされる値を有し得る。  
10

**【 0 0 9 1 】**

図 8 に示されるように、R P L 変更情報がビットストリームにおいてシグナリングされると ref\_pic\_list\_modification\_flag\_IX が示す場合であっても、list\_entry\_IX[0] のシグナリングは、NumPocTotalCurr 及び num\_ref\_idx\_IX\_active\_minus1 に依存する。NumPocTotalCurr が 2 に等しく、num\_ref\_idx\_IX\_active\_minus1 が 1 に等しい場合、list\_entry\_IX[0] の値は、ref\_pic\_list\_modification\_flag\_IX に基づいて推定され得る。というのは、2 つの選択肢のみ可能であるからである（0 のデフォルト値又は 1 の非デフォルト値）。

**【 0 0 9 2 】**

したがって、図 8 は、リスト・エントリのシンタックス要素がシグナリングされるかどうかに関する条件を含む。R P L 0 では、その条件は、「if(ref\_pic\_list\_modification\_flag\_I0&&!(NumPocTotalCurr==2&&num\_ref\_idx\_I0\_active\_minus1==0))」である。R P L 1 では、その条件は、「if(ref\_pic\_list\_modification\_flag\_I1&&!(NumPocTotalCurr==2&&num\_ref\_idx\_I1\_active\_minus1==0))」である。

**【 0 0 9 3 】**

図 8 の例において、list\_entry\_I0[i] は、R P L 0 の現在位置に配置される RefPicListTemp0（一時的なバージョンの R P L）内の参照ピクチャのインデックスを指定する。list\_entry\_I0[i] シンタックス要素の長さは、Ceil(Log2(NumPocTotalCurr)) ビットである。list\_entry\_I0[i] の値は、0 ~ NumPocTotalCurr-1（0 と NumPocTotalCurr-1 を含む）である。NumPocTotalCurr が 2 に等しく、num\_ref\_idx\_I0\_active\_minus1 が 0 に等しい場合、シンタックス要素 list\_entry\_I0[0] は、ref\_pic\_list\_modification\_flag\_I0 に等しいと推定される。そうでなければ、シンタックス要素 list\_entry\_I0[i] が存在しない場合、0 に等しいと推定される。

**【 0 0 9 4 】**

図 8 の例において、list\_entry\_I1[i] は、R P L 1 の現在位置に配置される RefPicListTemp1（一時的なバージョンの R P L）内の参照ピクチャのインデックスを指定する。list\_entry\_I1[i] シンタックス要素の長さは、Ceil(Log2(NumPocTotalCurr)) ビットである。list\_entry\_I1[i] の値は、0 ~ NumPocTotalCurr-1（0 と NumPocTotalCurr-1 を含む）である。NumPocTotalCurr が 2 に等しく、num\_ref\_idx\_I1\_active\_minus1 が 0 に等しい場合、シンタックス要素 list\_entry\_I1[0] は、ref\_pic\_list\_modification\_flag\_I1 に等しいと推定される。そうでなければ、シンタックス要素 list\_entry\_I1[i] が存在しない場合、0 に等しいと推定される。

**【 0 0 9 5 】**

図 9 は、ref\_pic\_lists\_modification() シンタックス構造の別の例示的なシンタックス（900）を示しており、これは、スライス・ヘッダの一部としてシグナリングされ得る  
40  
50

。例示的なシンタックス(900)において、シンタックス要素list\_entry\_IX[0]が、ビットストリームにおいて条件付きでシグナリングされる。しかしながら、図8の例示的なシンタックスと比較すると、チェックされる条件が異なっている。また、list\_entry\_IX[ ]のシンタックス要素のシグナリングは、重み付き予測が使用されるかどうかに応じて調整され得る。

#### 【0096】

図9に従うと、重み付き予測が有効又無効にされるかは、リスト・エントリのシンタックス要素がビットストリームにおいてどのようにシグナリングされるかに影響を及ぼす。weighted\_pred\_flagが0に等しいPスライス又はweighted\_bipred\_flagが0に等しいBスライスでは、重み付きが無効にされる。図9の例示的なシンタックス(900)に従うと、重み付き予測が無効にされている場合、NumPocTotalCurrが2に等しく、num\_ref\_idx\_IX\_active\_minus1が1に等しいときには、list\_entry\_IX[0]及びlist\_entry\_IX[1]は送信されない。そのような場合、list\_entry\_IX[0]及びlist\_entry\_IX[1]は、1及び0にそれぞれ推定される。というのは、RPL変更は、他の可能性のみでは必要とされないからである(すなわち、list\_entry\_IX[0]及びlist\_entry\_IX[1]は、0及び1にそれぞれ等しい)。

#### 【0097】

したがって、図9は、リスト・エントリのシンタックス要素がシグナリングされるかどうかに関する条件を含む。RPL 0では、その条件は、「if(ref\_pic\_list\_modification\_flag\_10&&(NumPocTotalCurr==2&&num\_ref\_idx\_10\_active\_minus1==0)&&(NumPocTotalCurr==2&&num\_ref\_idx\_10\_active\_minus1==1&&((weighted\_pred\_flag!=1&&slice\_type==P)||((weighted\_bipred\_flag!=1&&slice\_type==B))))」である。RPL 1では、その条件は、「if(ref\_pic\_list\_modification\_flag\_11&&(NumPocTotalCurr==2&&num\_ref\_idx\_11\_active\_minus1==0)&&(NumPocTotalCurr==2&&num\_ref\_idx\_11\_active\_minus1==1&&weighted\_bipred\_flag!=1))」である。

#### 【0098】

さらに、NumPocTotalCurrが2に等しくない場合、又はnum\_ref\_idx\_IX\_active\_minus1が1に等しくない場合であっても、重み付き予測が無効にされるときには(Pスライスでは、weighted\_pred\_flagが0に等しく；Bスライスでは、weighted\_bipred\_flagが0に等しい)、list\_entry\_IX[i]シンタックス要素の長さは、Ceil(Log2(NumPocTotalCurr-i))ビットに制限される。この場合、リスト内に各参照ピクチャを1回配置することのみに有用であり、したがって、有用な可能性の数は、インデックスiが増えるにつれ、減少する。

#### 【0099】

図9の例において、list\_entry\_10[i]は、RPL 0の現在位置に配置されるRefPicListTemp0(一時的なバージョンのRPL)内の参照ピクチャのインデックスを指定する。重み付き予測が無効にされる(Pスライスでは、weighted\_pred\_flagが0に等しく；Bスライスでは、weighted\_bipred\_flagが0に等しい)ときには、list\_entry\_10[i]シンタックス要素の長さは、Ceil(Log2(NumPocTotalCurr-i))ビットである。そうでなければ、list\_entry\_10[i]シンタックス要素の長さは、Ceil(Log2(NumPocTotalCurr))ビットである。NumPocTotalCurrが2に等しく、num\_ref\_idx\_10\_active\_minus1が0に等しい場合、シンタックス要素list\_entry\_10[0]は、(図8の例のように)ref\_pic\_list\_modification\_flag\_10に等しいと推定される。そうでなければ、NumPocTotalCurrが2に等しく、num\_ref\_idx\_10\_active\_minus1が1に等しく、重み付き予測が無効にされる(weighted\_pred\_flagが0に等しく、現在のスライスがPスライスである、あるいは、weighted\_bipred\_flagが0に等しく、現在のスライスがBスライスである)場合、シンタックス要素list\_entry\_10[0]及びlist\_entry\_10[1]は、1及び0にそれぞれ等しいと推定される。そうでなければ、シンタックス要素list\_entry\_10[i]が存在しない場合、0に等しいと推定される。

#### 【0100】

重み付き予測が無効にされる(weighted\_pred\_flagが0に等しく、現在のスライスがP

10

20

30

40

50

スライスである、あるいは、weighted\_bipred\_flagが0に等しく、現在のスライスがBスライスである場合、list\_entry\_l0[i]の値は、0～NumPocTotalCurr-(i+1)（0とNumPocTotalCurr-(i+1)を含む）であり、エントリ値が解析された後、リストRefPicListTemp0から各エントリlist\_entry\_l0[i]を除去することにより、リストRefPicListTemp0は短くなる。そうでなければ、list\_entry\_l0[i]の値は、0～NumPocTotalCurr-1（0とNumPocTotalCurr-1を含む）である。

#### 【0101】

図9の例において、list\_entry\_l1[i]は、RPL-1の現在位置に配置されるRefPicListTemp1（一時的なバージョンのRPL）内の参照ピクチャのインデックスを指定する。  
重み付き予測が無効にされる（weighted\_bipred\_flagが0に等しい、というのは、Bスライスのみがリスト1を使用するからである）場合、list\_entry\_l1[i]シンタックス要素の長さは、Ceil(Log2(NumPocTotalCurr-i))ビットである。そうでなければ、list\_entry\_l1[i]シンタックス要素の長さは、Ceil(Log2(NumPocTotalCurr))ビットである。NumPocTotalCurrが2に等しく、num\_ref\_idx\_l1\_active\_minus1が0に等しい場合、シンタックス要素list\_entry\_l1[0]は、（図8の例のように）ref\_pic\_list\_modification\_flag\_l1に等しいと推定される。そうでなければ、NumPocTotalCurrが2に等しく、num\_ref\_idx\_l1\_active\_minus1が1に等しく、重み付き予測が無効にされる（weighted\_bipred\_flagが0に等しく、現在のスライスがBスライスである）場合、シンタックス要素list\_entry\_l1[0]及びlist\_entry\_l1[1]は、1及び0にそれぞれ等しいと推定される。そうでなければ、シンタックス要素list\_entry\_l1[i]が存在しない場合、0に等しいと推定される。

#### 【0102】

重み付き予測が無効にされる（weighted\_bipred\_flagが0に等しく、現在のスライスがBスライスである）場合、list\_entry\_l1[i]の値は、0～NumPocTotalCurr-(i+1)（0とNumPocTotalCurr-(i+1)を含む）であり、エントリ値が解析された後、リストRefPicListTemp1から各エントリlist\_entry\_l1[i]を除去することにより、リストRefPicListTemp1は短くなる。そうでなければ、list\_entry\_l1[i]の値は、0～NumPocTotalCurr-1（0とNumPocTotalCurr-1を含む）である。

#### 【0103】

D. RPL変更フラグの条件付きシグナリング及び解析のための一般化された技術

図10は、RPL変更フラグの条件付きシグナリングのための一般化された技術（1000）を示している。例えば、図3を参照して説明したビデオ・エンコーダを実装するコンピューティング・デバイスは、技術（1000）を実行することができる。

#### 【0104】

デバイスは、条件を評価する（1010）。例えば、条件は、参照ピクチャの総数を示す変数に少なくとも部分的に依存する。いくつかの例示的な実施例において、変数は、NumPocTotalCurrであり、エンコーダは、変数が1より大きいかどうかをチェックする。代替的に、エンコーダは、他の条件及び/又は追加の条件を評価してもよい。1010で評価される条件は、単一のファクタ（例えば、参照ピクチャの総数を示す変数の値）を含んでもよいし、1010で評価される条件は、複数のファクタ（例えば、参照ピクチャの総数を示す変数の値に加えて、1以上の他のファクタ）を含んでもよい。条件は、RPL変更構造の処理の一部として評価され得る（1010）。あるいは、条件は、スライス・ヘッダの処理の一部として評価されてもよい（1010）。

#### 【0105】

評価の結果に応じて、デバイスは、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従ってRPLが変更される（例えば、置換される、調整される）かどうかを示すフラグを、ビットストリームにおいて条件付きでシグナリングする（1020）。例えば、フラグは、ref\_pic\_list\_modification\_flag\_l0又はref\_pic\_list\_modification\_flag\_l1のうちの1つであり、スライス・ヘッダのRPL変更構造の一部として、条件付きでシグナリングされ得る。あるいは、条件が評価された（1010）後、評価の結果に応じて、（ビットストリームにおいて明示的にシグナリングされるシンタックス

要素に従って RPL が変更されるかどうかを示す 1 以上のフラグを含む ) RPL 変更構造が、ビットストリームにおいて条件付きでシグナリングされてもよい。

#### 【 0106 】

デバイスは、RPL 変更構造がシグナリングされるときに、スライスごとに、又は何らかの他の単位で、技術 (1000) を繰り返すことができる。

#### 【 0107 】

図 11 は、RPL 変更フラグの条件付き解析のための一般化された技術 (1100) を示している。例えば、図 4 を参照して説明したビデオ・デコーダを実装するコンピューティング・デバイスは、技術 (1100) を実行することができる。

#### 【 0108 】

デコーダは、条件を評価する (1110)。例えば、条件は、参照ピクチャの総数を示す変数に少なくとも部分的に依存する。いくつかの例示的な実施例において、変数は、NumPocTotalCurr であり、デコーダは、変数が 1 より大きいかどうかをチェックする。代替的に、デコーダは、他の条件及び / 又は追加の条件を評価してもよい。1110 で評価される条件は、単一のファクタ (例えば、参照ピクチャの総数を示す変数の値) を含んでもよいし、1110 で評価される条件は、複数のファクタ (例えば、参照ピクチャの総数を示す変数の値に加えて、1 以上の他のファクタ) を含んでもよい。条件は、RPL 変更構造の処理の一部として評価され得る (1110)。あるいは、条件は、スライス・ヘッダの処理の一部として評価されてもよい (1110)。

#### 【 0109 】

評価の結果に応じて、デバイスは、ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従って RPL が変更される (例えば、置換される、調整される) かどうかを示すフラグを、ビットストリームから条件付きで解析する (1120)。例えば、フラグは、ref\_pic\_list\_modification\_flag\_10 又は ref\_pic\_list\_modification\_flag\_11 のうちの 1 つであり、スライス・ヘッダの RPL 変更構造の一部として、条件付きでシグナリングされ得る。あるいは、条件が評価された (1110) 後、評価の結果に応じて、(ビットストリームにおいて明示的にシグナリングされるシンタックス要素に従って RPL が変更されるかどうかを示す 1 以上のフラグを含む ) RPL 変更構造が、ビットストリームから条件付きで解析されてもよい。

#### 【 0110 】

デバイスは、RPL 変更構造がシグナリングされるときに、スライスごとに、又は何らかの他の単位で、技術 (1100) を繰り返すことができる。

#### 【 0111 】

E . リスト・エントリの条件付きシグナリング及び解析のための一般化された技術

図 12 は、RPL 変更のリスト・エントリの条件付きシグナリングのための一般化された技術 (1200) を示している。例えば、図 3 を参照して説明したビデオ・エンコーダを実装するコンピューティング・デバイスは、技術 (1200) を実行することができる。

#### 【 0112 】

デバイスは、条件を評価する (1210)。例えば、条件は、参照ピクチャの総数を示す変数 (例えば、いくつかの実施例では、NumPocTotalCurr) に少なくとも部分的に依存する。あるいは、条件は、RPL のアクティブ参照ピクチャの数に少なくとも部分的に依存してもよい。あるいは、条件は、重み付き予測が無効にされるかどうかに少なくとも部分的に依存してもよい。異なるロジックを使用して、現在のスライスが P スライス又は B スライスであるかに応じて、且つ / あるいはどの RPL がシグナリング / 解析されているかどうかに応じて、重み付き予測が無効にされるかどうかをチェックすることができる。例えば、(P スライス又は B スライスにより使用され得る) 第 1 の RPL の条件をチェックするためのロジックは、(B スライスのみにより使用され得る) 第 2 の RPL の条件をチェックするためのロジックとは異なる。代替的に、エンコーダは、他の条件及び / 又は追加の条件を評価してもよい。

10

20

30

40

50

**【0113】**

評価の結果に応じて、デバイスは、RPLをどのように変更する（例えば、置換する、調整する）かを示すリスト・エントリの1以上のシンタックス要素を、ビットストリームにおいて条件付きでシグナリングする（1220）。例えば、リスト・エントリの1以上のシンタックス要素は、スライス・ヘッダのRPL変更構造の一部として、条件付きでシグナリングされる。

**【0114】**

いくつかの例示的な実施例において、（a）参照ピクチャの総数が2に等しく、（b）RPLのアクティブ参照ピクチャの数が1に等しい場合、リスト・エントリの1以上のシンタックス要素は、ビットストリームには存在せず、リスト・エントリのうちの1つに関する値が推定される。他の例示的な実施例において、この条件に加えて、（c）参照ピクチャの総数が2に等しく、（d）RPLのアクティブ参照ピクチャの数が2に等しく、（e）重み付き予測が無効にされる場合、リスト・エントリの1以上のシンタックス要素は、ビットストリームには存在せず、リスト・エントリのうちの2つに関する値が推定される。

10

**【0115】**

デバイスは、RPL変更構造がシグナリングされるときに、スライスごとに、又は何らかの他の単位で、技術（1200）を繰り返すことができる。

**【0116】**

図13は、RPL変更のリスト・エントリの条件付き解析のための一般化された技術（1300）を示している。例えば、図4を参照して説明したビデオ・デコーダを実装するコンピューティング・デバイスは、技術（1300）を実行することができる。

20

**【0117】**

デコーダは、条件を評価する（1310）。例えば、条件は、参照ピクチャの総数を示す変数（例えば、いくつかの実施例では、NumPocTotalCurr）に少なくとも部分的に依存する。あるいは、条件は、RPLのアクティブ参照ピクチャの数に少なくとも部分的に依存してもよい。あるいは、条件は、重み付き予測が無効にされるかどうかに少なくとも部分的に依存してもよい。異なるロジックを使用して、現在のスライスがPスライス又はBスライスであるかに応じて、且つ／あるいはどのRPLがシグナリング／解析されているかどうかに応じて、重み付き予測が無効にされるかどうかをチェックすることができる。代替的に、デコーダは、他の条件及び／又は追加の条件を評価してもよい。

30

**【0118】**

評価の結果に応じて、デバイスは、RPLをどのように変更する（例えば、置換する、調整する）かを示すリスト・エントリの1以上のシンタックス要素を、ビットストリームから条件付きで解析する（1320）。例えば、リスト・エントリの1以上のシンタックス要素は、スライス・ヘッダのRPL変更構造から、条件付きで解析される。

**【0119】**

いくつかの例示的な実施例において、（a）参照ピクチャの総数が2に等しく、（b）RPLのアクティブ参照ピクチャの数が1に等しい場合、リスト・エントリの1以上のシンタックス要素は、ビットストリームには存在せず、リスト・エントリのうちの1つに関する値が推定される。他の例示的な実施例において、この条件に加えて、（c）参照ピクチャの総数が2に等しく、（d）RPLのアクティブ参照ピクチャの数が2に等しく、（e）重み付き予測が無効にされる場合、リスト・エントリの1以上のシンタックス要素は、ビットストリームには存在せず、リスト・エントリのうちの2つに関する値が推定される。

40

**【0120】**

デバイスは、RPL変更構造がシグナリングされるときに、スライスごとに、又は何らかの他の単位で、技術（1300）を繰り返すことができる。

**【0121】**

F. リスト・エントリのシグナリング及び解析を調整するための一般化された技術

50

図14は、RPL変更のリスト・エントリのシグナリングを調整するための一般化された技術(1400)を示している。例えば、図3を参照して説明したビデオ・エンコーダを実装するコンピューティング・デバイスは、技術(1400)を実行することができる。

#### 【0122】

デバイスは、条件を評価する(1410)。例えば、条件は、重み付き予測が無効にされるかどうかに少なくとも部分的に依存する。異なるロジックを使用して、現在のスライスがPスライス又はBスライスであるかに応じて、且つ／あるいはどのRPLがシグナリング／解析されているかどうかに応じて、重み付き予測が無効にされるかどうかをチェックすることができる。例えば、(Pスライス又はBスライスにより使用され得る)第1のRPLの条件をチェックするためのロジックは、(Bスライスのみにより使用され得る)第2のRPLの条件をチェックするためのロジックとは異なる。代替的に、エンコーダは、他の条件及び／又は追加の条件を評価してもよい。

#### 【0123】

評価の結果に応じて、デバイスは、RPLをどのように変更する(例えば、置換する、調整する)かを示すリスト・エントリの1以上のシンタックス要素のビットストリームにおけるシグナリングを調整する(1420)。詳細には、1以上のシンタックス要素の少なくとも1つの長さ(ビット単位)が調整される。例えば、リスト・エントリのインデックス $i$ について、重み付き予測が無効にされる場合、1以上のシンタックス要素の少なくとも1つの長さ(ビット単位)は、 $i$ が増えるにつれ、減少する。いくつかの例示的な実施例において、重み付き予測が無効にされる場合、リスト・エントリ[ $i$ ]の所与のシンタックス要素の長さは、 $\text{Ceil}(\text{Log}_2(\text{NumPocTotalCurr}-i))$ である。そうでなければ(重み付き予測が有効にされる場合)、リスト・エントリ[ $i$ ]の所与のシンタックス要素の長さは、 $\text{Ceil}(\text{Log}_2(\text{NumPocTotalCurr}))$ ビットである。

#### 【0124】

デバイスは、RPL変更構造がシグナリングされるときに、スライスごとに、又は何らかの他の単位で、技術(1400)を繰り返すことができる。

#### 【0125】

図15は、RPL変更のリスト・エントリの解析を調整するための一般化された技術(1500)を示している。例えば、図4を参照して説明したビデオ・デコーダを実装するコンピューティング・デバイスは、技術(1500)を実行することができる。

#### 【0126】

デコーダは、条件を評価する(1510)。例えば、条件は、重み付き予測が無効にされるかどうかに少なくとも部分的に依存する。異なるロジックを使用して、現在のスライスがPスライス又はBスライスであるかに応じて、且つ／あるいはどのRPLがシグナリング／解析されているかどうかに応じて、重み付き予測が無効にされるかどうかをチェックすることができる。代替的に、デコーダは、他の条件及び／又は追加の条件を評価してもよい。

#### 【0127】

評価の結果に応じて、デバイスは、RPLをどのように変更する(例えば、置換する、調整する)かを示すリスト・エントリの1以上のシンタックス要素のビットストリームからの解析を調整する(1520)。詳細には、1以上のシンタックス要素の少なくとも1つの長さ(ビット単位)が調整される。例えば、リスト・エントリのインデックス $i$ について、重み付き予測が無効にされる場合、1以上のシンタックス要素の少なくとも1つの長さ(ビット単位)は、 $i$ が増えるにつれ、減少する。いくつかの例示的な実施例において、重み付き予測が無効にされる場合、リスト・エントリ[ $i$ ]の所与のシンタックス要素の長さは、 $\text{Ceil}(\text{Log}_2(\text{NumPocTotalCurr}-i))$ である。そうでなければ(重み付き予測が有効にされる場合)、リスト・エントリ[ $i$ ]の所与のシンタックス要素の長さは、 $\text{Ceil}(\text{Log}_2(\text{NumPocTotalCurr}))$ ビットである。

#### 【0128】

10

20

30

40

50

デバイスは、RPL 変更構造がシグナリングされるときに、スライスごとに、又は何らかの他の単位で、技術(1500)を繰り返すことができる。

#### 【0129】

##### G. 代替例

図7a、図7b、図10、及び図11は、条件に基づく、ref\_pic\_list\_modification\_flag\_10又はref\_pic\_list\_modification\_flag\_11等のフラグの条件付きシグナリング及び解析を示している。このように、(リスト・エントリのシンタックス要素等の)追加のRPL 変更情報のシグナリングが制御される。図7aを参照して説明したように、RPL 変更フラグのシグナリング及び解析は、ref\_pic\_lists\_modification()構造の一部として条件を評価することにより制御され得る。代替的に、図7bを参照して説明したように、RPL 変更構造(例えば、ref\_pic\_lists\_modification()構造)のシグナリング及び解析は、スライス・ヘッダ処理の一部として、又は他の方法で同じ条件を評価することにより制御されてもよい。例えば、変数NumPocTotalCurrが1より大きい場合、ref\_pic\_lists\_modification()構造がシグナリングされる。そうでなければ(変数NumPocTotalCurrが1以下である場合)、ref\_pic\_lists\_modification()構造はシグナリングされず、図7aを参照して上述したように、リスト・エントリの値が推定される。図10に拡張すると、条件が評価された後、評価の結果に応じて、RPL 変更シンタックス構造が、条件付きでシグナリングされる。図11に拡張すると、条件が評価された後、評価の結果に応じて、RPL 変更シンタックス構造が、条件付きで解析される。

#### 【0130】

10

例示の目的上、詳細な説明は、いくつかのパラメータ及び変数に関する特定の名前を伴う様々な例を含む。本明細書で説明したイノベーションは、そのような名前を有するパラメータ又は変数を伴う実施例に限定されない。代わりに、本明細書で説明したイノベーションは、様々なタイプのパラメータ及び変数を用いて実装されてもよい。

20

#### 【0131】

##### H.さらなる革新的特徴

特許請求の範囲に加えて、本明細書で説明した革新的特徴は、以下の表に示される特徴を含むが、それらの特徴に限定されるものではない。

#### 【0132】

【表 1 - 1】

#	特徴	
<b>A. RPLのリスト・エントリのシンタックス要素の条件付きシグナリング</b>		
A1	<p>ビデオ・エンコーダにより実行される方法であって、      条件を評価する評価ステップと、      前記評価ステップの結果に応じて、RPLをどのように変更するかを示すリスト・エン      トリの1以上のシンタックス要素を、ビットストリームにおいて条件付きでシグナリング      するステップと、      を含む、方法。</p>	10
A2	<p>ビデオ・デコーダにより実行される方法であって、      条件を評価する評価ステップと、      前記評価ステップの結果に応じて、RPLをどのように変更するかを示すリスト・エン      トリの1以上のシンタックス要素を、ビットストリームから条件付きで解析するステップ      と、      を含む、方法。</p>	20
A3	前記条件は、参照ピクチャの総数を示す変数に少なくとも部分的に依存する、特徴A1 又はA2に記載の方法。	
A4	前記変数は、NumPocTotalCurrである、特徴A3に記載の方法。	
A5	前記条件は、前記RPLのアクティブ参照ピクチャの数に少なくとも部分的に依存する、 特徴A1～A4のいずれかに記載の方法。	30
A6	前記条件は、重み付き予測が無効にされるかどうかに少なくとも部分的に依存する、特 徴A1～A5のいずれかに記載の方法。	
A7	異なるロジックを使用して、現在のスライスがPスライス又はBスライスであるかに応 じて、且つ／あるいはどのRPLがシグナリング／解析されているかに応じて、重 み付き予測が無効にされるかどうかをチェックする、特徴A6に記載の方法。	
A8	前記条件は、(a) 参照ピクチャの総数が2に等しく、(b) 前記RPLのアクティブ 参照ピクチャの数が1に等しいかに少なくとも部分的に依存する、特徴A1又はA 2に記載の方法。	40

【0133】

【表1 - 2】

A9	(a) 参照ピクチャの前記総数が2に等しく、(b) 前記RPLのアクティブ参照ピクチャの前記数が1に等しい場合、リスト・エントリの前記1以上のシンタックス要素は、前記ビットストリームには存在せず、前記リスト・エントリのうちの1つに関する値が推定される、特徴A8に記載の方法。	
A10	前記条件は、(c) 参照ピクチャの総数が2に等しく、(d) 前記RPLのアクティブ参照ピクチャの数が2に等しく、(e) 重み付き予測が無効にされるかどうかに少なくとも部分的に依存する、特徴A1又はA2に記載の方法。	10
A11	(c) 参照ピクチャの前記総数が2に等しく、(d) 前記RPLのアクティブ参照ピクチャの前記数が2に等しく、(e) 重み付き予測が無効にされる場合、リスト・エントリの前記1以上のシンタックス要素は、前記ビットストリームには存在せず、前記リスト・エントリのうちの2つに関する値が推定される、特徴A10に記載の方法。	
A12	リスト・エントリの前記1以上のシンタックス要素は、スライス・ヘッダのRPL変更構造の一部として、条件付きでシグナリングされる、特徴A1～A11のいずれかに記載の方法。	20
A13	前記RPLは、Pスライスに関連付けられたRPL_0である、特徴A1～A12のいずれかに記載の方法。	
A14	前記評価ステップと、複数のRPLのそれぞれについて、前記条件付きシグナリング又は前記条件付き解析とを繰り返すステップをさらに含み、前記複数のRPLは、Bスライスに関連付けられた、RPL_0及びRPL_1を含む、特徴A1～A12のいずれかに記載の方法。	30
A15	特徴A1～A14のいずれか1つに記載の方法を実行するよう適合されたコンピューティング・デバイス。	
A16	特徴A1～A14のいずれか1つに記載の方法をコンピューティング・デバイスに実行させるコンピュータ実行可能命令を記憶した有体のコンピュータ読み取り可能媒体。	

【0 1 3 4】

【表1 - 3】

B. RPLのリスト・エントリのシンタックス要素の長さの調整	
B1	ビデオ・エンコーダにより実行される方法であって、 条件を評価する評価ステップと、 前記評価ステップの結果に応じて、RPLをどのように変更するかを示すリスト・エン トリの1以上のシンタックス要素のビットストリームにおけるシグナリングを調整する調 整ステップであって、前記1以上のシンタックス要素の少なくとも1つの長さが調整され る、調整ステップと、 を含む、方法。 10
B2	ビデオ・デコーダにより実行される方法であって、 条件を評価する評価ステップと、 前記評価ステップの結果に応じて、RPLをどのように変更するかを示すリスト・エン トリの1以上のシンタックス要素のビットストリームからの解析を調整する調整ステップ であって、前記1以上のシンタックス要素の少なくとも1つの長さが調整される、調整ス テップと、 を含む、方法。 20
B3	前記条件は、重み付き予測が無効にされるかどうかに少なくとも部分的に依存する、特 徴B1又はB2に記載の方法。
B4	異なるロジックを使用して、現在のスライスがPスライス又はBスライスであるかに応 じて、且つ／あるいはどのRPLがシグナリング／解析されているかどうかに応じて、重 み付き予測が無効にされるかどうかをチェックする、特徴B3に記載の方法。 30
B5	前記リスト・エントリのインデックスiについて、重み付き予測が無効にされる場合、 前記1以上のシンタックス要素の前記少なくとも1つの前記長さは、iが増えるにつれ、 減少する、特徴B3に記載の方法。
B6	前記リスト・エントリのインデックスiについて、重み付き予測が無効にされる場合、 リスト・エントリ[i]の所与のシンタックス要素の前記長さは、 $\text{Ceil}(\text{Log}_2(\text{NumPocTotalCurr}-i))$ ビットであり、重み付き予測が有効にされる場合、リスト・ エントリ[i]の所与のシンタックス要素の前記長さは、 $\text{Ceil}(\text{Log}_2(\text{NumPocTotalCurr}))$ ビ ットである、特徴B3に記載の方法。 40
B7	特徴B1～B6のいずれか1つに記載の方法を実行するよう適合されたコンピューティ ング・デバイス。
B8	特徴B1～B6のいずれか1つに記載の方法をコンピューティング・デバイスに実行さ せるコンピュータ実行可能命令を記憶した有体のコンピュータ読み取り可能媒体。

【0135】

【表1 - 4】

C. 一般	
C1	<p>エンコーダにより実行される方法であって、 ビデオを符号化するステップと、 前記の符号化されたビデオを含むビットストリームの少なくとも一部を出力する出力ステップであって、本明細書で説明したイノベーションのうちの1つに従って、RPL情報をシグナリングするステップを含む出力ステップと、 を含む、方法。</p>
C2	<p>デコーダにより実行される方法であって、 符号化されたビデオを含むビットストリームの少なくとも一部を受信する受信ステップであって、本明細書で説明したイノベーションのうちの1つに従って、シグナリングされたRPL情報を解析するステップを含む受信ステップと、 前記符号化されたビデオを復号化するステップと、 を含む、方法。</p>
C3	特徴C1又はC2に記載の方法を実行するよう適合されたコンピューティング・デバイス。
C4	特徴C1又はC2に記載の方法をコンピューティング・デバイスに実行させるコンピュータ実行可能命令を記憶した有体のコンピュータ読み取り可能媒体。

## 【0136】

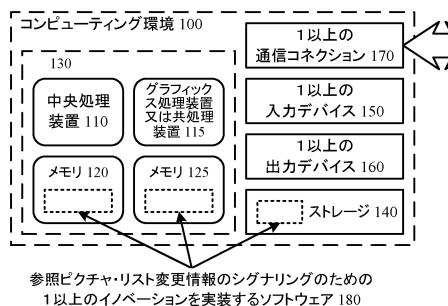
開示した本発明の原理を適用することができる多くの可能な実施形態の観点から、例示した実施形態は、本発明の好ましい例に過ぎないことを認識すべきであり、本発明の範囲を限定するものとして解釈すべきではない。むしろ、本発明の範囲は、請求項により定められる。したがって、我々は、このような請求項の範囲及び精神に含まれる全てを、我々の発明として特許請求する。

10

20

30

【図1】



【 図 2 a 】

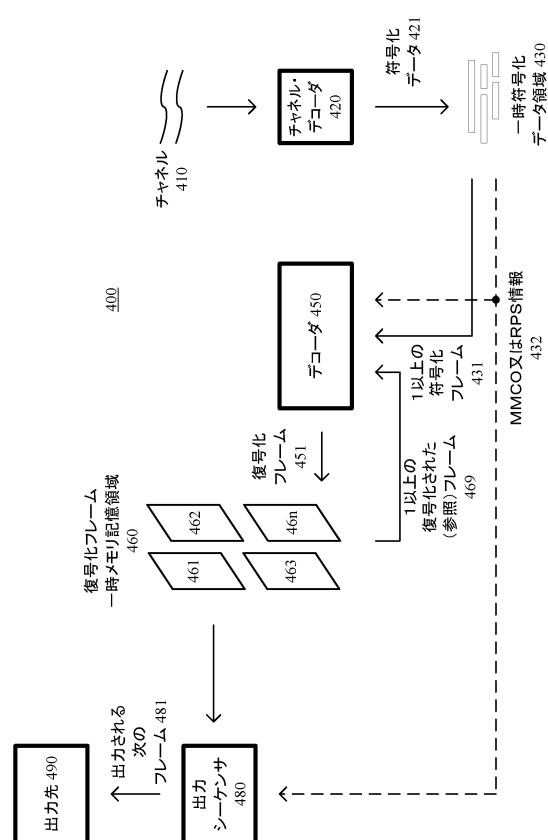
201

RTCツール 210  
エンコーダ 220  
デコーダ 270

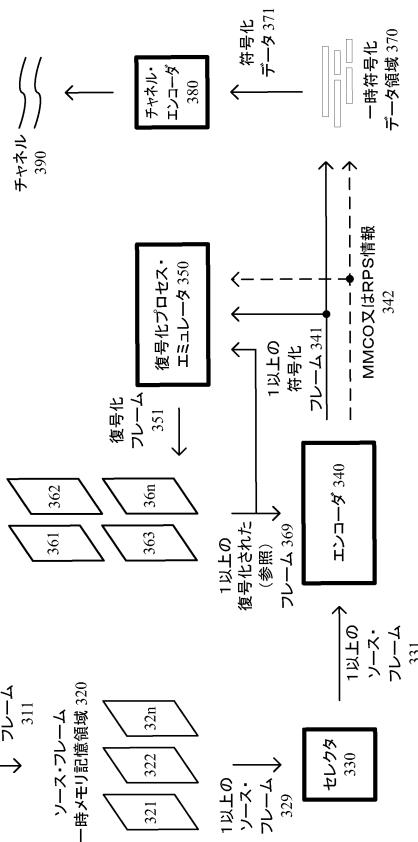
【図2b】

符号化ツール 212  
エンコーダ220

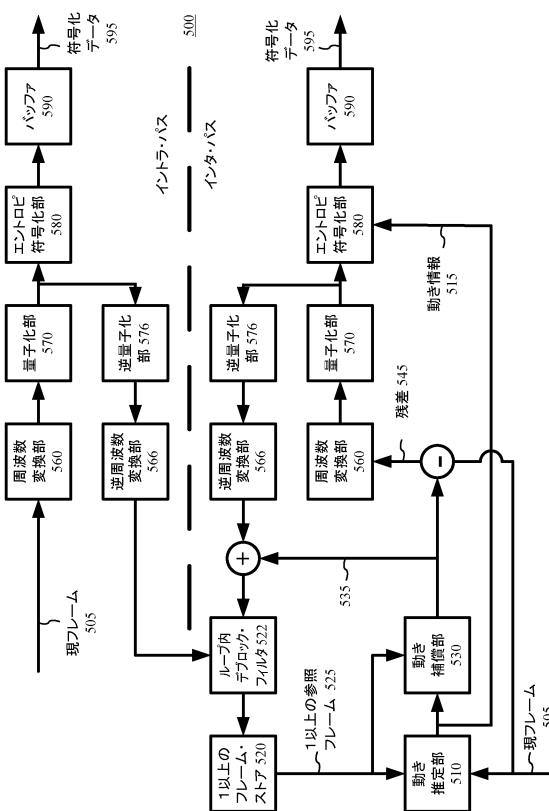
〔 図 4 〕



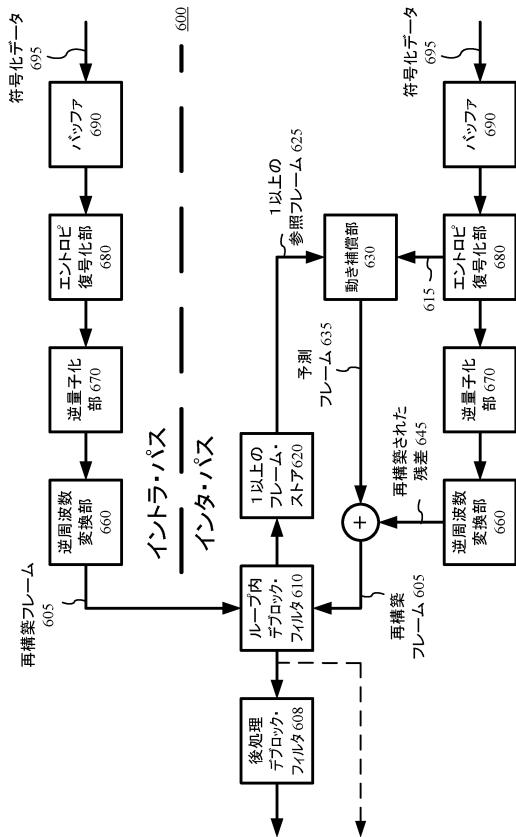
【図3】



( 5 )



【図6】



【図7 c】

760

ref_pic_lists_modification()	記述子
ref_pic_list_modification_flag_10	u(1)
if( ref_pic_list_modification_flag_10 )	
for( i = 0; i <= num_ref_idx_10_active_minus1; i++ )	
list_entry_10[i]	u(v)
if( slice_type == B ) {	
ref_pic_list_modification_flag_11	u(1)
if( ref_pic_list_modification_flag_11 )	
for( i = 0; i <= num_ref_idx_11_active_minus1; i++ )	
list_entry_11[i]	u(v)
}	
}	

【図8】

800

ref_pic_lists_modification()	記述子
ref_pic_list_modification_flag_10	u(1)
if( ref_pic_list_modification_flag_10 &&	
!( NumPocTotalCurr == 2 &&	
num_ref_idx_10_active_minus1 == 0 )	
for( i = 0; i <= num_ref_idx_10_active_minus1; i++ )	
list_entry_10[i]	u(v)
if( slice_type == B ) {	
ref_pic_list_modification_flag_11	u(1)
if( ref_pic_list_modification_flag_11 &&	
!( NumPocTotalCurr == 2 &&	
num_ref_idx_11_active_minus1 == 0 )	
for( i = 0; i <= num_ref_idx_11_active_minus1; i++ )	
list_entry_11[i]	u(v)
}	
}	

【図7 a】

700

ref_pic_lists_modification()	記述子
if( NumPocTotalCurr > 1 )	
ref_pic_list_modification_flag_10	u(1)
if( ref_pic_list_modification_flag_10 )	
for( i = 0; i <= num_ref_idx_10_active_minus1; i++ )	
list_entry_10[i]	u(v)
if( slice_type == B ) {	
if( NumPocTotalCurr > 1 )	
ref_pic_list_modification_flag_11	u(1)
if( ref_pic_list_modification_flag_11 )	
for( i = 0; i <= num_ref_idx_11_active_minus1; i++ )	
list_entry_11[i]	u(v)
}	
}	

【図7 b】

750

slice_segment_header()	記述子
...	
if( slice_type == P    slice_type == B ) {	
num_ref_idx_active_override_flag	u(1)
if( num_ref_idx_active_override_flag ) {	
num_ref_idx_10_active_minus1	ue(v)
if( slice_type == B )	
num_ref_idx_11_active_minus1	ue(v)
}	
if( list_modification_present_flag && NumPocTotalCurr > 1 )	
ref_pic_lists_modification()	
...	
}	

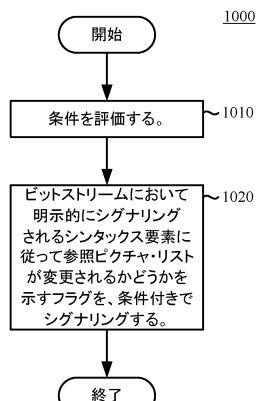
【図9】

900

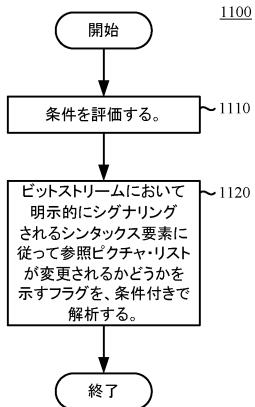
ref_pic_lists_modification()	記述子
ref_pic_list_modification_flag_10	u(1)
if( ref_pic_list_modification_flag_10 &&	
!( NumPocTotalCurr == 2 &&	
num_ref_idx_10_active_minus1 == 0 ) &&	
!( NumPocTotalCurr == 2 && num_ref_idx_10_active_minus1 == 1 &&	
( weighted_bipred_flag != 1 && slice_type == P )	
( weighted_bipred_flag != 1 && slice_type == B ) ) )	
for( i = 0; i <= num_ref_idx_10_active_minus1; i++ )	
list_entry_10[i]	u(v)
if( slice_type == B ) {	
ref_pic_list_modification_flag_11	u(1)
if( ref_pic_list_modification_flag_11 &&	
!( NumPocTotalCurr == 2 &&	
num_ref_idx_11_active_minus1 == 0 ) &&	
!( NumPocTotalCurr == 2 && num_ref_idx_11_active_minus1 == 1 &&	
weighted_bipred_flag != 1 ) )	
for( i = 0; i <= num_ref_idx_11_active_minus1; i++ )	
list_entry_11[i]	u(v)
}	
}	

【図10】

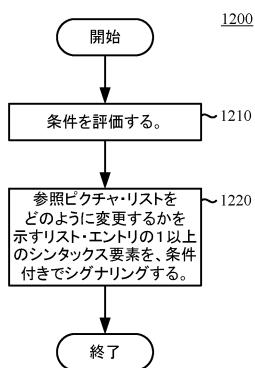
1000



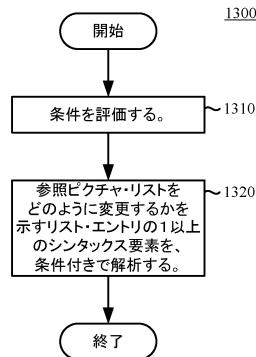
【図11】



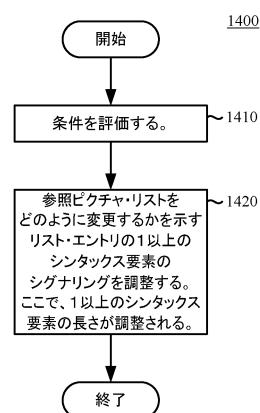
【図12】



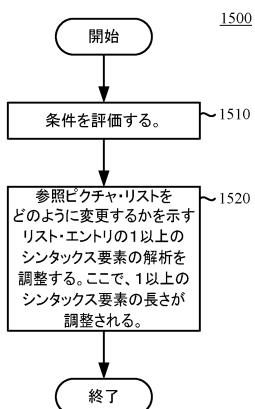
【図13】



【図14】



【図15】



---

フロントページの続き

(72)発明者 サリヴァン , ゲイリー ジェイ .

アメリカ合衆国 98052-6399 ワシントン州 レッドモンド ワン マイクロソフト  
ウェイ マイクロソフト コーポレーション エルシーエー - インターナショナル パテンツ 内

(72)発明者 カヌムリ , サンディープ

アメリカ合衆国 98052-6399 ワシントン州 レッドモンド ワン マイクロソフト  
ウェイ マイクロソフト コーポレーション エルシーエー - インターナショナル パテンツ 内

審査官 堀井 啓明

(56)参考文献 国際公開第2014/050989 (WO , A1 )

(58)調査した分野(Int.Cl. , DB名)

H04N19/00 - 19/98