

(19)日本国特許庁(JP)

**(12)特許公報(B2)**

(11)特許番号  
**特許第7536785号**  
**(P7536785)**

(45)発行日 令和6年8月20日(2024.8.20)

(24)登録日 令和6年8月9日(2024.8.9)

(51)国際特許分類

H 0 4 N	19/70 (2014.01)	F I	H 0 4 N	19/70
H 0 4 N	19/30 (2014.01)		H 0 4 N	19/30

請求項の数 11 (全74頁)

(21)出願番号	特願2021-562870(P2021-562870)
(86)(22)出願日	令和3年5月18日(2021.5.18)
(65)公表番号	特表2022-537244(P2022-537244)
A)	
(43)公表日	令和4年8月25日(2022.8.25)
(86)国際出願番号	PCT/US2021/032924
(87)国際公開番号	WO2021/236610
(87)国際公開日	令和3年11月25日(2021.11.25)
審査請求日	令和3年10月21日(2021.10.21)
(31)優先権主張番号	63/027,826
(32)優先日	令和2年5月20日(2020.5.20)
(33)優先権主張国・地域又は機関	米国(US)
(31)優先権主張番号	63/035,274
(32)優先日	令和2年6月5日(2020.6.5)

最終頁に続く

(73)特許権者	520353802 テンセント・アメリカ・エルエルシー アメリカ合衆国 9 4 3 0 6 カリフォルニア州 パロアルト パーク・ブルヴァード 2 7 4 7
(74)代理人	100110364 弁理士 実広 信哉
(74)代理人	100150197 弁理士 松尾 直樹
(72)発明者	ビヨンドウ・チエ アメリカ合衆国・カリフォルニア・ 9 4 3 0 6 ・パロ・アルト・パーク・ブルバード・ 2 7 4 7 ・テンセント・アメリカ・エルエルシー内 ステファン・ヴェンガー
(72)発明者	最終頁に続く

(54)【発明の名称】 符号化されたビデオストリームを復号するための方法、装置およびコンピュータプログラム

**(57)【特許請求の範囲】****【請求項1】**

少なくとも1つのプロセッサが実行する方法であって、

ピクチャを含むアクセスユニットを含む符号化されたビデオストリームを受信するステップと、

前記符号化されたビデオストリームのアクセス・ユニット・デリミタにおいて、前記アクセスユニットに、イントラ・ランダム・アクセス・ポイント (IRAP) ピクチャおよび段階的復号リフレッシュ (GDR) ピクチャのうちのいずれか一方が含まれているか否かを示す第1のフラグをシグナリングするステップと、

前記符号化されたビデオストリームのピクチャヘッダ内の、前記ピクチャが前記IRAPピクチャであるか否かを示す第2のフラグをシグナリングするステップと、

前記第1のフラグおよび前記第2のフラグの前記シグナリングに基づいて、前記ピクチャを現在のピクチャとして復号するステップと、を含み、

復号ピクチャバッファからのピクチャの出力および除去処理に使用されるフラグが存在せず、前記第1のフラグの値が1に等しいとき、前記第2のフラグの値が1に等しいことが必要とされるように、前記第1のフラグの値と前記第2のフラグの値とが整合される、

方法。

**【請求項2】**

前記符号化されたビデオストリームの前記ピクチャヘッダ内の、前記ピクチャが前記GDRピクチャであるか否かを示す第3のフラグをシグナリングするステップ

をさらに含み、

前記第1のフラグの値が1に等しく、前記第2のフラグの値が0に等しいとき、前記第3のフラグの値が1に等しいことが必要とされるように、前記第1のフラグの値と前記第3のフラグの値とが整合されている、

請求項1に記載の方法。

#### 【請求項3】

前記第3のフラグは、前記ピクチャが前記IRAPピクチャではないことを示す前記第2のフラグに基づいてシグナリングされる、請求項2に記載の方法。

#### 【請求項4】

前記第1のフラグは、前記ピクチャが前記IRAPピクチャおよび前記GDRピクチャのいずれか一方であることを示す値を有し、

前記第2のフラグは、前記ピクチャがIRAPピクチャであることを示す値を有し、

前記方法は、IRAPピクチャの前の任意のピクチャが出力されるか否かを示す、符号化されたビデオストリームのピクチャのスライスのスライスヘッダ内の第3のフラグをシグナリングするステップをさらに含む、

請求項1に記載の方法。

#### 【請求項5】

前記スライスのネットワーク抽象化層（NAL）ユニット型式を判定するステップをさらに含み、

前記第3のフラグは、判定された前記NALユニット型式に基づいてシグナリングされる、請求項4に記載の方法。

#### 【請求項6】

前記第3のフラグは、前記NALユニット型式がIDR\_W\_RADL、IDR\_N\_LP、またはCRA\_NUTに等しいと判定されることに基づいてシグナリングされる、請求項5に記載の方法。

#### 【請求項7】

前記符号化されたビデオストリームの前記ピクチャヘッダ内の、前記ピクチャが前記GDRピクチャであるか否かを示す第4のフラグをシグナリングするステップをさらに含み、

前記第1のフラグの値が1に等しく、前記第2のフラグの値が0に等しいとき、前記第4のフラグの値が1に等しいことが必要とされるように、前記第1のフラグの値と前記第4のフラグの値とは整合されている、

請求項5または6に記載の方法。

#### 【請求項8】

復号する前記ステップは、

参照ピクチャリストを構築するステップと、

前記参照ピクチャリスト内の利用できない参照ピクチャを生成するステップと、

前記参照ピクチャリスト内の参照ピクチャについて、以下の制約が適用されるビットストリーム適合性を確認するステップであって、

前記参照ピクチャリスト内にあると示されているエントリの数は、前記参照ピクチャリスト内にあると示されているアクティブエントリの数以上であり、

前記参照ピクチャリスト内のアクティブエントリによって参照される各ピクチャは、復号ピクチャバッファ（DPB）内に存在し、前記現在のピクチャの時間識別子値以下の時間識別子値を有し、

前記参照ピクチャリスト内のエントリによって参照される各ピクチャは前記現在のピクチャではなく、ピクチャ・ヘッダ・フラグによって参照ピクチャである可能性があることが示される、

ステップと、

を含む、請求項1～7のいずれか一項に記載の方法。

#### 【請求項9】

ビットストリーム適合性を確認する前記ステップは、前記現在のピクチャが独立した復

10

20

30

40

50

号器リフレッシュ (IDR) ピクチャ、クリーン・ランダム・アクセス (CRA) ピクチャ、または段階的復号リフレッシュ (GDR) ピクチャであるとの判定に基づいて実行される、請求項 8 に記載の方法。

【請求項 10】

請求項 1 ~ 9 のいずれか一項に記載の方法を行うように構成された装置。

【請求項 11】

少なくとも 1 つのプロセッサに、請求項 1 ~ 9 のいずれか一項に記載の方法を実行させるためのコンピュータプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

関連出願への相互参照

本出願は、2020 年 6 月 11 日に出願された米国仮特許出願第 63 / 037, 903 号、2020 年 6 月 8 日出願の米国仮特許出願第 63 / 036, 335 号、2020 年 6 月 5 日出願の米国仮特許出願第 63 / 035, 274 号、2020 年 5 月 20 日に出願された米国仮特許出願第 63 / 027, 826 号、および 2021 年 5 月 14 日に出願された米国特許出願第 17 / 320, 764 号の優先権を主張し、これらの開示はその全体が参考により本明細書に組み込まれる。

【0002】

本開示の実施形態は、ビデオの符号化および復号に関し、より具体的には、複数の層を有する符号化されたビデオストリームにおけるランダム・アクセス・ピクチャおよびその出力処理に関する。

10

【背景技術】

【0003】

動作補償を伴うインターピクチャ予測を使用したビデオの符号化および復号は、以前に使用されてきた。非圧縮デジタルビデオは、一連のピクチャを含むことができ、各ピクチャは、例えば  $1920 \times 1080$  の輝度サンプルおよび関連する色差サンプルの空間次元を有する。一連のピクチャは、例えば毎秒 60 ピクチャずなわち 60 Hz の固定または可変ピクチャレート（非公式にはフレームレートとしても知られる）を有し得る。非圧縮ビデオには、重要なビットレート要件がある。例えば、サンプルあたり 8 ビットの 1080p 60 4 : 2 : 0 ビデオ（60Hz のフレームレートにおける  $1920 \times 1080$  輝度サンプル解像度）には、1.5 Gbit / s に近い帯域幅が必要である。このようなビデオを 1 時間使用するには、600GB を超える記憶領域が必要である。

20

【0004】

ビデオの符号化および復号の目的の 1 つは、圧縮による入力ビデオ信号の冗長性の低減であり得る。圧縮は、前述の帯域幅または記憶領域の要件を、場合によっては 2 枠以上削減するのに役立ち得る。可逆圧縮と非可逆圧縮の両方、およびそれらの組み合わせを使用できる。可逆圧縮とは、元の信号の正確な複製を圧縮された元の信号から復元することができる技術を指す。非可逆圧縮を使用する場合、再構築された信号は元の信号と同一ではない可能性があるが、元の信号と再構築された信号との間の歪みは十分に小さいといい得るため、再構築された信号は目的の用途に役立つ。ビデオの場合、非可逆圧縮が広く採用されている。許容される歪みの量は用途によって異なる。例えば、特定の消費者ストリーミング用途のユーザは、テレビ投稿用途のユーザよりも高い歪みを許容し得る。達成可能な圧縮率は、より高い許容可能 / 受忍可能な歪みによって、より高い圧縮率が得られることを反映し得る。

30

【0005】

ビデオ符号化器および復号器は、例えば、そのうちのいくつかが以下に紹介される、動作補償、変換、量子化、エントロピー符号化など、いくつかの広範なカテゴリの手法を利用できる。

40

【0006】

以前は、ビデオ符号化器および復号器は、ほとんどの場合、符号化されたビデオシーケ

50

ンス (coded video sequence : CVS)、ピクチャグループ (Group of Pictures : GOP)、または同様のマルチ・ピクチャ・タイムフレームに対して定義され、一定のままであった所与のピクチャサイズで動作する傾向があった。例えば、MPEG - 2では、システム設計を使用して、シーンのアクティビティなどの要因に応じて水平解像度 (それによって、ピクチャサイズ) を変更したが、Iピクチャでのみ、したがって通常はGOP用であった。CVS内で異なる解像度を使用するための参照ピクチャのリサンプリングは、例えばITU - T Rec . H . 263 Annex Pで使用されている。しかしながら、ここではピクチャサイズは変化せず、参照ピクチャのみが再サンプリングされ、ピクチャキャンバスの一部のみが使用される (ダウンサンプリングの場合)、またはシーンの一部のみがキャプチャされる (アップサンプリングの場合) 可能性がある。さらに、H . 263 Annex Qは、個々のマクロブロックを上方または下方に (各次元で) 2倍だけ再サンプリングすることを可能にする。ここでも、ピクチャサイズは同じままである。マクロブロックのサイズはH . 263では固定されているため、シグナリングする必要はない。

#### 【0007】

予測ピクチャのピクチャサイズの変更は、最新のビデオ符号化においてより主流になった。例えば、VP9により、参照ピクチャの再サンプリングおよびピクチャ全体の解像度の変更が可能になる。同様に、VVC (例えば、Hendryら、「VVCのための適応解像度変更 (ARC) について ( “On adaptive resolution change (ARC) for VVC” )」、Joint Video Team文書JVT - M 0135 - v1、2019年1月9～19日を含み、その全体が本明細書に組み込まれる) に向けてなされたある提案により、異なる (より高いまたはより低い) 解像度への参照ピクチャ全体の再サンプリングが可能になる。そのような文書では、シーン・パラメータ・セット内で符号化され、ピクチャ・パラメータ・セット内のピクチャ毎の構文要素によって参照される異なる候補解像度が提案されている。

#### 【0008】

Brossら、「多用途ビデオ符号化 (ドラフト9)」、Joint Video Experts Team文書JVET - R 2001 - vA、2020年4月は、その全体が本明細書に組み込まれる。

#### 【発明の概要】

#### 【課題を解決するための手段】

#### 【0009】

符号化されたビデオストリーム内では、ネットワーク抽象化層 (network abstraction layer : NAL) ユニットヘッダ、パラメータセット、ピクチャヘッダ、またはスライスヘッダなどの高レベルの構文構造でランダム・アクセス・ポイント情報を示すために広く使用されている。ランダムアクセス情報に基づいて、ランダム・アクセス・ピクチャに関連付けられた復号された先頭ピクチャが管理される。本開示では、ランダムアクセス処理に関連する復号ピクチャ管理を明確にするために、いくつかの関連する構文要素および制約が説明される。

#### 【0010】

ビデオ・ビットストリームがトリック・モード・プレイによってランダムにアクセスされるとき、イントラ・ランダム・アクセス・ポイント (intra random access point : IR AP) ピクチャは、ビットストリームの中間点へのランダムアクセス、およびランダム・アクセス・ポイントにおけるビデオ・ビットストリームの復号の成功を可能にことができる。可能な方法の1つは、ある程度の復元時間でシーンを段階的にリフレッシュすることである。VVCおよび他のビデオコーデックでは、段階的復号リフレッシュ (gradual de coding refresh : GDR) ピクチャおよびアクセスユニット (access unit : AU) は、段階的復号リフレッシュによるランダムアクセス動作の構文およびセマンティクスを指定するように定義される。本開示では、GDRのシグナリングおよび復号処理を正しく指定するために、その構文、セマンティクス、および制約が記載される。

#### 【0011】

PまたはBスライスにおけるインター予測のために1つまたは複数の参照ピクチャリストが構築される場合、1つまたは複数のピクチャは、ランダムアクセスまたは意図しないピ

クチャ損失のために利用できない可能性がある。復号器のクラッシュまたは意図しない挙動を回避するために、ピクセルおよびパラメータの初期設定値を有する利用できないピクチャを生成することが望ましい。利用できないピクチャを生成した後、参照ピクチャリスト内のすべての参照ピクチャの検証を確認する必要がある場合がある。

#### 【 0 0 1 2 】

本開示の実施形態は、複数の層を有する符号化されたビデオストリームにおけるランダム・アクセス・ピクチャおよびその出力処理に関する。本開示の実施形態は、複数の層を有する符号化されたビデオストリームにおけるランダム・アクセス・ピクチャおよびその先頭ピクチャ出力指示に関する。本開示の実施形態は、複数の層を有する符号化されたビデオストリームにおける段階的復号リフレッシュおよび復元ポイントを用いてランダム・アクセス・ピクチャをシグナリングすることに関する。本開示の実施形態は、複数の層を有する符号化されたビデオストリームにおける参照ピクチャリスト構築および利用できないピクチャ生成に関する。本開示の実施形態は、ビデオ・ビットストリームにおける適応ピクチャサイズのシグナリングのための技術を含む。

10

#### 【 0 0 1 3 】

本開示の1つまたは複数の実施形態は、少なくとも1つのプロセッサによって実行される方法を含む。この方法は、ピクチャを含むアクセスユニットを含む符号化されたビデオストリームを受信するステップと、符号化されたビデオストリームのアクセス・ユニット・デリミタにおいて、アクセスユニットに、イントラ・ランダム・アクセス・ポイント (IRAP) ピクチャおよび段階的復号リフレッシュ (GDR) ピクチャのうちのいずれか一方が含まれているか否かを示す第1のフラグをシグナリングするステップと、符号化されたビデオストリームのピクチャヘッダ内に、ピクチャがIRAPピクチャであるか否かを示す第2のフラグをシグナリングするステップと、第1のフラグおよび第2のフラグのシグナリングに基づいて、ピクチャを現在のピクチャとして復号するステップと、を含み、第1のフラグの値および第2のフラグの値が位置合わせされる。

20

#### 【 0 0 1 4 】

一実施形態によれば、本方法は、符号化されたビデオストリームのピクチャヘッダ内に、ピクチャがGDRピクチャであるか否かを示す第3のフラグをシグナリングするステップをさらに含み、第1のフラグの値および第3のフラグの値が位置合わせされる。

30

#### 【 0 0 1 5 】

一実施形態によれば、第3のフラグは、ピクチャがIRAPピクチャではないことを示す第2のフラグに基づいてシグナリングされる。

#### 【 0 0 1 6 】

一実施形態によれば、第1のフラグは、ピクチャがIRAPピクチャおよびGDRピクチャのうちのいずれか1つであることを示す値を有し、第2のフラグは、ピクチャがIRAPピクチャであることを示す値を有し、本方法は、符号化されたビデオストリームのピクチャのスライスのスライスヘッダ内で、IRAPピクチャより前の任意のピクチャが出力されるか否かを示す第3のフラグをシグナリングするステップをさらに含む。

#### 【 0 0 1 7 】

一実施形態によれば、本方法は、スライスのネットワーク抽象化層 (NAL) ユニット型式を判定するステップであって、第3のフラグが、判定されたNALユニット型式に基づいてシグナリングされる、ステップをさらに含む。

40

#### 【 0 0 1 8 】

一実施形態によれば、第3のフラグは、IDR\_W\_RADL、IDR\_N\_LP、またはCRA\_NUTに等しいと判定されているNALユニット型式に基づいてシグナリングされる。

#### 【 0 0 1 9 】

一実施形態によれば、本方法は、符号化されたビデオストリームのピクチャヘッダ内に、ピクチャがGDRピクチャであるか否かを示す第4のフラグをシグナリングするステップをさらに含み、第1のフラグの値および第4のフラグの値が位置合わせされる。

#### 【 0 0 2 0 】

50

一実施形態によれば、第3のフラグは、ピクチャがIRAPピクチャではないことを示す第2のフラグに基づいてシグナリングされる。

#### 【0021】

一実施形態によれば、復号するステップは、参照ピクチャリストを構築するステップと、参照ピクチャリスト内の利用できない参照ピクチャを生成するステップと、参照ピクチャリスト内の参照ピクチャについて、ビットストリーム適合性を確認するステップであって、参照ピクチャリスト内にあると示されるエントリの数は、参照ピクチャリスト内にあると示されるアクティブエントリの数以上であり、参照ピクチャリスト内のアクティブエントリによって参照される各ピクチャは、復号ピクチャバッファ（DPB）内に存在し、現在のピクチャの時間識別子値以下の時間識別子値を有し、参照ピクチャリスト内のエントリによって参照される各ピクチャは、現在のピクチャではなく、参照ピクチャである可能性があることをピクチャ・ヘッダ・フラグによって示されるという制約が適用される、ステップと、を含む。

#### 【0022】

一実施形態によれば、ビットストリーム適合性を確認するステップは、現在のピクチャが独立した復号器リフレッシュ（independent decoder refresh：IDR）ピクチャ、クリーン・ランダム・アクセス（clean random access：CRA）ピクチャ、または段階的復号リフレッシュ（GDR）ピクチャであるとの判定に基づいて実行される。

#### 【0023】

1つまたは複数の実施形態によれば、システムが提供される。本システムは、ピクチャを含むアクセスユニットを含む符号化されたビデオストリームを受信するように構成された少なくとも1つのプロセッサと、コンピュータコードを格納するメモリであって、コンピュータコードは、少なくとも1つのプロセッサに対して、符号化されたビデオストリームのアクセス・ユニット・デリミタにおいて、アクセスユニットが、イントラ・ランダム・アクセス・ポイント（IRAP）ピクチャおよび段階的復号リフレッシュ（GDR）ピクチャのうちのいずれか一方が含まれているか否かを示す第1のフラグをシグナリングするよう構成された第1のシグナリングコードと、少なくとも1つのプロセッサに対して、符号化されたビデオストリームのピクチャヘッダ内の、ピクチャがIRAPピクチャであるか否かを示す第2のフラグをシグナリングするよう構成された第2のシグナリングコードと、少なくとも1つのプロセッサに、第1のフラグおよび第2のフラグのシグナリングに基づいて、ピクチャを現在のピクチャとして復号させるよう構成された復号コードと、を含み、第1のフラグの値と第2のフラグの値とが位置合わせされる、メモリと、を備えている。

#### 【0024】

一実施形態によれば、コンピュータコードは、少なくとも1つのプロセッサに対して、符号化されたビデオストリームのピクチャヘッダ内の、ピクチャがGDRピクチャであるか否かを示す第3のフラグをシグナリングするよう構成された第3のシグナリングコードであって、第1のフラグの値と第3のフラグの値とは位置合わせされている、第3のシグナリングコードをさらに含む。

#### 【0025】

一実施形態によれば、第3のフラグは、ピクチャがIRAPピクチャではないことを示す第2のフラグに基づいてシグナリングされる。

#### 【0026】

一実施形態によれば、第1のフラグは、ピクチャがIRAPピクチャおよびGDRピクチャのうちのいずれか1つであることを示す値を有し、第2のフラグは、ピクチャがIRAPピクチャであることを示す値を有し、コンピュータコードは、少なくとも1つのプロセッサに、符号化されたビデオストリームのピクチャのスライスのスライスヘッダ内の、IRAPピクチャより前の任意のピクチャが出力されているか否かを示す第3のフラグをシグナリングするよう構成された第3のシグナリングコードをさらに含む。

#### 【0027】

一実施形態によれば、コンピュータコードは、少なくとも1つのプロセッサに、スライ

スのネットワーク抽象化層（NAL）ユニット型式を判定せるように構成されたコードを判定することをさらに含み、第3のフラグは、判定されたNALユニット型式に基づいてシグナリングされる。

#### 【0028】

一実施形態によれば、第3のフラグは、IDR\_W\_RADL、IDR\_N\_LP、またはCRA\_NUTに等しいと判定されているNALユニット型式に基づいてシグナリングされる。

#### 【0029】

一実施形態によれば、コンピュータコードは、少なくとも1つのプロセッサに、符号化されたビデオストリームのピクチャヘッダ内の、ピクチャがGDRピクチャであるか否かを示す第4のフラグをシグナリングせしめるように構成された第4のシグナリングコードをさらに含み、第1のフラグの値および第4のフラグの値は位置合わせされる。10

#### 【0030】

一実施形態によれば、第3のフラグは、ピクチャがIRAPピクチャではないことを示す第2のフラグに基づいてシグナリングされる。

#### 【0031】

一実施形態によれば、復号コードは、少なくとも1つのプロセッサに参照ピクチャリストを構築せしめるように構成された構築コードと、少なくとも1つのプロセッサが参照ピクチャリストに利用できない参照ピクチャを生成するように構成された生成コードと、少なくとも1つのプロセッサに、参照ピクチャリスト内の参照ピクチャのビットストリーム適合性を確認せしめるように構成された確認コードであつて、以下の制約、すなわち、参照ピクチャリストにあることが示されているエントリの数は、参照ピクチャリストにあることが示されているアクティブエントリの数以上であり、参照ピクチャリストのアクティブエントリによって参照される各ピクチャは、復号ピクチャバッファ（DPB）に存在し、現在のピクチャの時間識別子値以下の時間識別子値を有し、参照ピクチャリストのエントリによって参照される各ピクチャは現在のピクチャではなく、ピクチャ・ヘッダ・フラグによって参照ピクチャである可能性があることが示される、が適用される、確認コードと、を含む。20

#### 【0032】

1つまたは複数の実施形態によれば、コンピュータ命令を記憶する非一時的コンピュータ可読媒体が提供される。コンピュータ命令は、ピクチャを含むアクセスユニットを含む符号化されたビデオストリームを受信する少なくとも1つのプロセッサによって実行された場合、少なくとも1つのプロセッサに、符号化されたビデオストリームのアクセス・ユニット・デリミタにおいて、アクセスユニットが、イントラ・ランダム・アクセス・ポイント（IRAP）ピクチャおよび段階的復号リフレッシュ（GDR）ピクチャのうちのいずれか一方が含まれているか否かを示す第1のフラグをシグナリングするステップと、符号化されたビデオストリームのピクチャヘッダ内の、ピクチャがIRAPピクチャであるか否かを示す第2のフラグをシグナリングするステップと、第1のフラグおよび第2のフラグのシグナリングに基づいて、ピクチャを現在のピクチャとして復号するステップと、を実行させ、第1のフラグの値と第2のフラグの値は位置合わせされる。30

#### 【0033】

開示された主題のさらなる特徴、性質、および様々な利点は、以下の詳細な説明および添付の図面からより明らかになるであろう。

#### 【図面の簡単な説明】

#### 【0034】

【図1】一実施形態による通信システムの簡略化されたブロック図の概略図である。

【図2】一実施形態による通信システムの簡略化されたブロック図の概略図である。

【図3】一実施形態による復号器の簡略化されたブロック図の概略図である。

【図4】一実施形態による符号化器の簡略化されたブロック図の概略図である。

【図5A】比較技術の実施形態によるARCパラメータをシグナリングするための第1の構成の概略図である。40

【図 5 B】比較技術の実施形態によるARCパラメータをシグナリングするための第2の構成の概略図である。

【図 6 A】一実施形態によるARCパラメータをシグナリングするための第1の構成の概略図である。

【図 6 B】一実施形態によるARCパラメータをシグナリングするための第2の構成の概略図である。

【図 6 C】一実施形態によるARCパラメータをシグナリングするための第3の構成の概略図である。

【図 7 A】一実施形態によるタイル・グループ・ヘッダの抜粋の概略図である。

【図 7 B】一実施形態によるシーケンス・パラメータ・セットの抜粋の概略図である。 10

【図 8】適応的な解像度変更を伴うスケーラビリティのための予測構造の一例である。

【図 9 A】一実施形態による構文表の一例を示す図である。

【図 9 B】一実施形態による構文表の一例を示す図である。

【図 10】一実施形態による、アクセスユニット当たりのPOCサイクルおよびアクセス・ユニット・カウント値の構文解析および復号の簡略ブロック図の概略図である。

【図 11】一実施形態による、多層サブピクチャを含むビデオ・ビットストリーム構造の概略図である。

【図 12】一実施形態による、拡張解像度を有する選択されたサブピクチャの表示の概略図である。

【図 13】一実施形態による、多層サブピクチャを含むビデオ・ビットストリームの復号および表示処理のブロック図である。 20

【図 14】一実施形態による、サブピクチャの拡張層を有する360ビデオ表示の概略図である。

【図 15 A】一実施形態に係る分割サブピクチャのレイアウトの一例を示す。

【図 15 B】一実施形態による、1つのサブピクチャの対応するサブピクチャサイズおよび位置情報の一例を示す図である。

【図 16】図15A～図15Bに示すサブピクチャの対応するピクチャ予測構造を示す。

【図 17】一実施形態に係る、一又は複数の層で符号化され得る複数のサブ領域に分割された入力ピクチャの一例を示す。

【図 18】図17に示すサブ領域の、ローカル領域の空間スケーラビリティ形式を有する、対応する層およびピクチャ予測構造を示す図である。 30

【図 19 A】一実施形態によるビデオ・パラメータ・セットの抜粋の概略図である。

【図 19 B】一実施形態によるシーケンス・パラメータ・セットの抜粋の概略図である。

【図 20】一実施形態によるサブピクチャ・レイアウト情報の構文表の一例である。

【図 21】一実施形態による、出力層および各出力層セットのプロファイル／階層／レベル情報を示す構文表の一例である。

【図 22】一実施形態による、出力層セット毎の出力層モードを示す構文表の一例である。

【図 23】出力層セット毎の各層の現在のサブピクチャを示す構文表の一例である。

【図 24】ビデオ・パラメータ・セットRBSPの構文表の一例である。

【図 25】出力層セットモードで出力層セットを示す構文表の一例である。 40

【図 26】ピクチャヘッダ内のIRAPピクチャまたはGDRピクチャと関連付けられたランダム・アクセス・ポイントを示すための構文表の一例である。

【図 27】AUデリミタ内のランダムアクセスAUを示す構文表の一例である。

【図 28】ピクチャヘッダにおけるIRAPピクチャの存在を示すための構文表の一例である。

。

【図 29】スライスヘッダ内の先頭ピクチャの出力を示す構文表の一例である。

【図 30】AUデリミタにおけるIRAPピクチャおよびGDRピクチャの存在を示すための構文表の一例である。

【図 31】GDRピクチャおよびピクチャヘッダ内のその復元ポイントを示すための構文表の一例である。 50

【図32】参照ピクチャリストの構築および利用できないピクチャの生成の簡略ブロック図の概略図である。

【図33】実施形態によるコンピュータコードのブロック図である。

【図34】実施形態を実施するのに適したコンピュータシステムの図である。

【発明を実施するための形態】

【0035】

図1は、本開示の実施形態による通信システム(100)の簡略化されたブロック図を示す。システム(100)は、ネットワーク(150)を介して相互接続された少なくとも2つの端末(110、120)を含み得る。データの一方向送信の場合、第1の端末(110)は、ネットワーク(150)を介して他の端末(120)に送信するために、ローカル位置でビデオデータを符号化することができる。第2の端末(120)は、ネットワーク(150)から他の端末の符号化されたビデオデータを受信し、符号化されたデータを復号し、復元されたビデオデータを表示することができる。一方向データ伝送は、メディア・サービング・アプリケーションなどにおいて一般的であり得る。

10

【0036】

図1は、例えば、ビデオ会議中に発生する可能性がある符号化されたビデオの双方向送信をサポートするために提供される第2の対の端末(130、140)を示す。データの双方向送信の場合、各端末(130、140)は、ネットワーク(150)を介して他の端末に送信するために、ローカル位置でキャプチャされたビデオデータを符号化することができる。各端末(130、140)はまた、他の端末によって送信された符号化されたビデオデータを受信し、符号化されたデータを復号し、復元されたビデオデータをローカル・ディスプレイ・デバイスに表示し得る。

20

【0037】

図1では、端末(110～140)は、サーバ、パーソナルコンピュータ、およびスマートフォン、ならびに／または任意の他の種類の端末として示されている場合がある。例えば、端末(110～140)は、ラップトップコンピュータ、タブレットコンピュータ、メディアプレーヤ、および／または専用ビデオ会議機器であってもよい。ネットワーク(150)は、例えば有線および／または無線通信ネットワークを含む、端末(110～140)間で符号化されたビデオデータを伝達する任意の数のネットワークを表す。通信ネットワーク(150)は、回路交換チャネルおよび／またはパケット交換チャネルでデータを交換することができる。代表的なネットワークには、電気通信ネットワーク、ローカル・エリア・ネットワーク、ワイド・エリア・ネットワークおよび／またはインターネットが含まれる。本議論の目的のために、ネットワーク(150)のアーキテクチャおよびトポロジーは、以下に本明細書で説明されない限り、本開示の動作にとって重要ではない場合がある。

30

【0038】

図2は、開示された主題のための用途の例として、ストリーミング環境におけるビデオ符号化器および復号器の配置を示す。開示された主題は、例えば、ビデオ会議、デジタルTV、CD、DVD、メモリスティックなどを含むデジタル媒体への圧縮ビデオの格納などを含む、他のビデオ対応アプリケーションにも等しく適用可能であり得る。

40

【0039】

図2に示すように、ストリーミングシステム(200)は、ビデオソース(201)および符号化器(203)を含むことができるキャプチャサブシステム(213)を含み得る。ビデオソース(201)は、例えば、デジタルカメラであってもよく、非圧縮ビデオ・サンプル・ストリーム(202)を作成するように構成されてもよい。非圧縮ビデオ・サンプル・ストリーム(202)は、符号化されたビデオ・ビットストリームと比較して高いデータ量を提供することができ、カメラ(201)に結合された符号化器(203)によって処理することができる。符号化器(203)は、以下により詳細に説明されるように、開示された主題の態様を可能にするまたは実装するためのハードウェア、ソフトウェア、またはそれらの組み合わせを含み得る。符号化されたビデオ・ビットストリーム(204)は、サンプルストリームと比較してより低いデータ量を含んでいてもよく、将来の使用のためにストリー

50

ミングサーバ(205)に格納され得る。1つまたは複数のストリーミングクライアント(206)は、ストリーミングサーバ(205)にアクセスして、符号化されたビデオ・ビットストリーム(204)の複製であり得るビデオ・ビットストリーム(209)を取得することができる。

#### 【0040】

実施形態では、ストリーミングサーバ(205)は、メディア対応ネットワーク要素(Media - Aware Network Element : MANE)としても機能することができる。例えば、ストリーミングサーバ(205)は、潜在的に異なるビットストリームをストリーミングクライアント(206)のうちの1つまたは複数に合わせるために、符号化されたビデオ・ビットストリーム(204)をブルーニングするように構成することができる。実施形態では、MANEは、ストリーミングシステム(200)内のストリーミングサーバ(205)から別々に提供されてもよい。10

#### 【0041】

ストリーミングクライアント(206)は、ビデオ復号器(210)およびディスプレイ(212)を含むことができる。ビデオ復号器(210)は、例えば、符号化されたビデオ・ビットストリーム(204)の入力複製であるビデオ・ビットストリーム(209)を復号し、ディスプレイ(212)または別のレンダリングデバイス(図示せず)上にレンダリングすることができる出力ビデオ・サンプル・ストリーム(211)を生成することができる。一部のストリーミングシステムでは、ビデオ・ビットストリーム(204、209)を特定のビデオ符号化 / 圧縮規格に従って符号化できる。そのような規格の例には、ITU - T勧告H.265が含まれるが、これに限定されない。一例では、開発中のビデオ符号化規格は、多用途ビデオ符号化(Versatile Video Coding : VVC)として非公式に知られている。本開示の実施形態は、VVCの文脈で使用され得る。20

#### 【0042】

図3は、本開示の一実施形態による、ディスプレイ(212)に取り付けられたビデオ復号器(210)の例示的な機能ブロック図を示す。

#### 【0043】

ビデオ復号器(210)は、チャネル(312)、受信機(310)、バッファメモリ(315)、エントロピー復号器 / 解析器(320)、スケーラ / 逆変換ユニット(351)、イントラ予測ユニット(352)、動作補償予測ユニット(353)、アグリゲータ(355)、ループ・フィルタ・ユニット(356)、参照ピクチャメモリ(357)、および現在のピクチャメモリ(358)を含むことができる。少なくとも一実施形態では、ビデオ復号器(210)は、集積回路、一連の集積回路、および / または他の電子回路を含むことができる。ビデオ復号器(210)はまた、関連するメモリを有する1つまたは複数のCPU上で実行されるソフトウェアにおいて部分的または全体的に具現化されてもよい。30

#### 【0044】

この実施形態および他の実施形態では、受信機(310)は、復号器(210)によって復号されるべき、1つまたは複数の符号化されたビデオシーケンス、1つの符号化されたビデオシーケンスを一度に受信してもよく、各符号化されたビデオシーケンスの復号は他の符号化されたビデオシーケンスから独立している。符号化されたビデオシーケンスは、符号化されたビデオデータを記憶する記憶デバイスへのハードウェア / ソフトウェアリンクであり得るチャネル(312)から受信され得る。受信機(310)は、それぞれの使用エンティティ(図示せず)に転送され得る他のデータ、例えば、符号化された音声データおよび / または補助データストリームとともに、符号化されたビデオデータを受信し得る。受信機(310)は、符号化されたビデオシーケンスを他のデータから分離することができる。ネットワークジッタに対抗するために、受信機(310)とエントロピー復号器 / 解析器(320)(以降、「解析器」)との間にバッファメモリ(315)が結合され得る。受信機(310)が十分な帯域幅および制御可能性のストア / フォワードデバイスから、または非同期ネットワークからデータを受信しているとき、バッファ(315)は使用されなくてもよく、または小さくてもよい。インターネットなどのベスト・エフォート・パケット・ネット40

トワークで使用するために、バッファ（315）が必要とされる場合があり、比較的大きくすることができ、適応サイズにすることができる。

#### 【0045】

ビデオ復号器（210）は、エントロピー符号化されたビデオシーケンスからシンボル（321）を再構築するための解析器（320）を含み得る。これらのシンボルのカテゴリは、例えば、復号器（210）の動作を管理するために使用される情報、および潜在的に、図2に示すように復号器に結合され得るディスプレイ（212）などのレンダリングデバイスを制御するための情報を含む。レンダリングデバイスの制御情報は、補足拡張情報（Supplementary Enhancement Information : SEI）メッセージまたはビデオユーザビリティ情報（Video Usability Information : VUI）パラメータ・セット・フラグメント（図示せず）の形であり得る。解析器（320）は、受信された符号化されたビデオシーケンスを解析／エントロピー復号することができる。符号化されたビデオシーケンスの符号化は、ビデオ符号化技術またはビデオ符号化規格に従うことができ、可変長符号化、ハフマン符号化、文脈依存性の有無にかかわらず算術符号化などを含む、当業者に周知の原理に従い得る。解析器（320）は、グループに対応する少なくとも1つのパラメータに基づいて、ビデオ復号器内のピクセルのサブグループのうちの少なくとも1つのサブ・グループ・パラメータの組を、符号化されたビデオシーケンスから抽出することができる。サブグループは、ピクチャグループ（GOP）、ピクチャ、タイル、スライス、マクロブロック、符号化ユニット（Coding Units : CU）、ブロック、変換ユニット（Transform Units : TU）、予測ユニット（Prediction Units : PU）などを含むことができる。解析器（320）はまた、変換係数、量子化パラメータ値、動作ベクトルなどのような符号化されたビデオシーケンス情報から抽出することができる。10

#### 【0046】

解析器（320）は、シンボル（321）を作成するために、バッファ（315）から受信されたビデオシーケンスに対してエントロピー復号／解析操作を実行することができる。

#### 【0047】

シンボル（321）の再構築には、符号化されたビデオピクチャまたはその一部（インターピクチャおよびイントラピクチャ、インターブロックおよびイントラブロックなど）の型式、およびその他の要因に応じて、複数の異なるユニットが含まれ得る。どのユニットが関与し、どのように関与するかは、解析器（320）によって符号化されたビデオシーケンスから解析されたサブグループ制御情報によって制御され得る。解析器（320）と以下の複数のユニットとの間のそのようなサブグループ制御情報の流れは、分かりやすくするために示されていない。30

#### 【0048】

すでに述べた機能ブロックのほかに、復号器210は、概念的には、以下で説明するようにならびにいくつかの機能ユニットに細分化され得る。商業的な制約の下で動作する実際の実装では、これらのユニットの多くは互いに密接に相互作用し、少なくとも部分的には互いに統合され得る。しかしながら、開示された主題を説明する目的で、以下の機能ユニットへの概念的細分化が適切である。

#### 【0049】

1つのユニットは、スケーラ／逆変換ユニット（351）であり得る。スケーラ／逆変換ユニット（351）は、量子化変換係数、ならびにどの変換を使用するか、ブロックサイズ、量子化係数、量子化スケーリング行列などをシンボル（複数可）（321）として含む制御情報を、解析器（320）から受信し得る。スケーラ／逆変換ユニット（351）は、アグリゲータ（355）に入力され得るサンプル値を備えるブロックを出力し得る。40

#### 【0050】

場合によっては、スケーラ／逆変換（351）の出力サンプルは、イントラ符号化されたブロック、すなわち、以前に再構築されたピクチャからの予測情報を使用していないが、現在のピクチャの以前に再構築された部分からの予測情報を使用することができるブロックに関連し得る。そのような予測情報を、イントラピクチャ予測ユニット（352）によっ50

て提供され得る。場合によっては、イントラピクチャ予測ユニット(352)は、現在のピクチャメモリ(358)からの現在の(部分的に再構築された)ピクチャからフェッチされた周囲のすでに再構築された情報を使用して、再構築中のブロックと同じサイズおよび形状のブロックを生成する。アグリゲータ(355)は、場合によっては、イントラ予測ユニット(352)が生成した予測情報を、スケーラ/逆変換ユニット(351)からの出力サンプル情報に、サンプル単位で付加する。

#### 【0051】

他の場合には、スケーラ/逆変換ユニット(351)の出力サンプルは、インター符号化された、潜在的に動作補償されたブロックに関連し得る。そのような場合、動作補償予測ユニット(353)は、参照ピクチャメモリ(357)にアクセスして、予測に使用されるサンプルをフェッチすることができる。フェッチされたサンプルをブロックに関連するシンボル(321)に従って動作補償した後、これらのサンプルは、出力サンプル情報を生成するために、アグリゲータ(355)によってスケーラ/逆変換ユニット(351)の出力に追加され得る(この場合、残差サンプルまたは残差信号と呼ばれる)。動作補償予測ユニット(353)が予測サンプルをフェッチする参照ピクチャメモリ(357)内のアドレスは、動作ベクトルによって制御することができる。動作ベクトルは、例えば、X、Y、および参照ピクチャ成分を有し得るシンボル(321)の形態で、動作補償予測ユニット(353)に利用可能であり得る。動作補償はまた、サブサンプルの正確な動作ベクトルが使用されているときに参照ピクチャメモリ(357)からフェッチされたサンプル値の補間、動作ベクトル予測機構などをも含み得る。

10

#### 【0052】

アグリゲータ(355)の出力サンプルは、ループ・フィルタ・ユニット(356)における様々なループフィルタリング技術の対象となり得る。ビデオ圧縮技術には、符号化されたビデオ・ビットストリームに含まれるパラメータによって制御され、解析器(320)からのシンボル(321)としてループ・フィルタ・ユニット(356)で利用できるインループフィルタ技術を含めることができるが、符号化されたピクチャまたは符号化されたビデオシーケンスの以前の(復号順で)部分の復号中に取得されたメタ情報に応答したり、以前に再構築およびループフィルタされたサンプル値に応答したりすることができる。

20

#### 【0053】

ループ・フィルタ・ユニット(356)の出力は、ディスプレイ(212)のようなレンダリングデバイスに出力するとともに、将来のインターピクチャ予測に使用するために参照ピクチャメモリ(357)に格納され得るサンプルストリームとすることができます。

30

#### 【0054】

完全に再構築されると、特定の符号化ピクチャは、将来の予測のための参照ピクチャとして使用することができる。符号化されたピクチャが完全に再構築され、符号化されたピクチャが(例えば、解析器(320)によって)参照ピクチャとして識別されると、現在の参照ピクチャは参照ピクチャメモリ(357)の一部になり得、次の符号化されたピクチャの再構築を開始する前に、新鮮な現在のピクチャメモリを再割り当てすることができる。

#### 【0055】

ビデオ復号器(210)は、ITU-T Rec. H.265などの規格に文書化され得る所定のビデオ圧縮技術に従って復号動作を実行することができる。符号化されたビデオシーケンスは、ビデオ圧縮技術または規格の構文と、ビデオ圧縮技術文書または規格、特にその中のプロフィール文書に準拠しているという意味において、使用されているビデオ圧縮技術または規格によって指定された構文に準拠している場合がある。また、いくつかのビデオ圧縮技術または規格に準拠するために、符号化されたビデオシーケンスの複雑さは、ビデオ圧縮技術または規格のレベルによって定義される範囲内であり得る。場合によっては、レベルは、最大ピクチャサイズ、最大フレームレート、最大再構築サンプルレート(例えば毎秒メガサンプルで測定される)、最大参照ピクチャサイズなどを制限する。レベルによって設定される制限は、場合によっては、仮想参照復号器(Hypothetical Reference D

40

50

ecoder : HRD ) 仕様および符号化されたビデオシーケンスにおいてシグナリングされたHRDバッファ管理のためのメタデータによってさらに制限され得る。

#### 【 0 0 5 6 】

一実施形態では、受信機(310)は、符号化されたビデオとともに追加の(冗長な)データを受信し得る。追加のデータは、符号化されたビデオシーケンスの一部として含まれ得る。データを適切に復号するために、および/または元のビデオデータをより正確に再構築するために、ビデオ復号器(210)によって追加のデータが使用され得る。追加のデータは、例えば、時間的、空間的、またはSNR拡張層、冗長スライス、冗長ピクチャ、前方誤り訂正コードなどの形式であり得る。

#### 【 0 0 5 7 】

図4は、本開示の一実施形態による、ビデオソース(201)に関連付けられたビデオ符号化器(203)の例示的な機能ブロック図を示す。

#### 【 0 0 5 8 】

ビデオ符号化器(203)は、例えば、ソース符号化器(430)、符号化エンジン(432)、(ローカル)復号器(433)、参照ピクチャメモリ(434)、予測器(435)、送信機(440)、エントロピー符号化器(445)、コントローラ(450)、およびチャネル(460)である符号化器を含むことができる。

#### 【 0 0 5 9 】

符号化器(203)は、符号化器(203)によって符号化されるビデオ画像をキャプチャすることができるビデオソース(201)(符号化器の一部ではない)からビデオサンプルを受信することができる。

#### 【 0 0 6 0 】

ビデオソース(201)は、符号化器(203)によって符号化されるソース・ビデオ・シーケンスを、任意の適切なビット深度(例えば、8ビット、10ビット、12ビット、...)であり得、任意の色空間(例えば、BT.601 Y CrCb、RGB、...)および適切なサンプリング構造(例えば、Y CrCb 4:2:0、Y CrCb 4:4:4)であり得るデジタル・ビデオ・サンプル・ストリームの形態で提供し得る。メディア・サービング・システムでは、ビデオソース(201)は、以前に準備されたビデオを格納する記憶デバイスであり得る。ビデオ会議システムでは、ビデオソース(203)は、ローカル画像情報をビデオシーケンスとしてキャプチャするカメラであり得る。ビデオデータは、順番に見たときに動作を与える複数の個別のピクチャとして提供され得る。ピクチャ自体は、ピクセルの空間アレイとして編成することができ、各ピクセルは、使用中のサンプリング構造、色空間などに応じて、1つまたは複数のサンプルを含み得る。当業者であれば、ピクセルとサンプルとの関係を容易に理解することができる。以下、サンプルに着目して説明する。

#### 【 0 0 6 1 】

一実施形態によれば、符号化器(203)は、用途によって要求されるように、リアルタイムで、または任意の他の時間制約の下で、ソース・ビデオ・シーケンスのピクチャを符号化し、符号化されたビデオシーケンス(443)に圧縮し得る。適切な符号化速度を強制することは、コントローラ(450)の1つの機能である。コントローラ(450)はまた、後述するように他の機能ユニットを制御してもよく、これらのユニットに機能的に結合されてもよい。結合は、明確にするために示されていない。コントローラ(450)によって設定されたパラメータには、レート制御関連パラメータ(ピクチャスキップ、量子化、レート歪み最適化手法のラムダ値など)、ピクチャサイズ、ピクチャグループ(GOP)レイアウト、最大動作ベクトル検索範囲などが含まれ得る。当業者は、特定のシステム設計用に最適化されたビデオ符号化器(203)に関係し得るので、コントローラ(450)の他の機能を容易に識別することができる。

#### 【 0 0 6 2 】

一部のビデオ符号化器は、熟練した人が「符号化ループ」として容易に認識できる方法で動作する。過度に単純化された説明として、符号化ループは、ソース符号化器(430)の符号化部分(符号化される入力ピクチャおよび参照ピクチャに基づいてシンボルを作成

10

20

30

40

50

する責任がある)からなり得、符号化器(203)に埋め込まれた(ローカル)復号器(433)は、シンボルを再構築してサンプルデータを作成する(リモート)復号器も作成する(シンボルと符号化されたビデオ・ビットストリームとの間の圧縮が、特定のビデオ圧縮技術で損失がない場合)。再構築されたサンプルストリームは、参照ピクチャメモリ(434)に入力される。シンボルストリームの復号は、復号器の場所(ローカルまたはリモート)に関係なくビット正確な結果をもたらすため、参照ピクチャ・メモリ・コンテンツもローカル符号化器とリモート符号化器との間でビットが正確である。言い換えると、符号化器の予測部は、復号中に予測を使用するときに復号器が「見る」とまったく同じサンプル値を参照ピクチャサンプルとして「見る」。参照ピクチャの同期性のこの基本原理(および、例えばチャネルエラーのために同期性を維持できない場合に生じるドリフト)は、当業者に知られている。

10

#### 【0063】

「ローカル」復号器(433)の動作は、「リモート」復号器(210)の動作と同じであり得、これは、図3に関連して上で詳細にすでに説明されている。しかしながら、シンボルが利用可能であり、エントロピー符号化器(445)および解析器(320)による符号化されたビデオシーケンスへのシンボルの符号化/復号は無損失であり得るため、チャネル(312)、受信機(310)、バッファ(315)、および解析器(320)を含む復号器(210)のエントロピー復号部分は、ローカル復号器(433)に完全に実装されない場合がある。

20

#### 【0064】

この時点でなされ得る観測は、復号器内に存在する構文解析/エントロピー復号を除く任意の復号器技術は、対応する符号化器内に実質的に同一の機能形態で存在する必要があり得るということである。このため、開示された主題は復号器動作に焦点を合わせている。符号化器技術の説明は、それらが包括的に説明された復号器技術の逆であり得るので省略することができる。特定の領域においてのみ、より詳細な説明が必要とされ、以下に提供される。

#### 【0065】

動作の一部として、ソース符号化器(430)は、「参照フレーム」として指定されたビデオシーケンスからの1つまたは複数の以前に符号化されたフレームを参照して入力フレームを予測的に符号化する動作補償予測符号化を実行し得る。このようにして、符号化エンジン(432)は、入力フレームのピクセルブロックと、入力フレームへの予測参照として選択され得る参照フレームのピクセルブロックとの間の差異を符号化する。

30

#### 【0066】

ローカルビデオ復号器(433)は、ソース符号化器(430)によって作成されたシンボルに基づいて、参照フレームとして指定され得るフレームの符号化されたビデオデータを復号し得る。符号化エンジン(432)の動作は、不可逆処理であることが有利であり得る。符号化されたビデオデータがビデオ復号器(図4には示されていない)で復号され得るとき、再構築されたビデオシーケンスは、通常、いくつかのエラーを伴うソース・ビデオ・シーケンスのレプリカであり得る。ローカルビデオ復号器(433)は、参照フレームに対してビデオ復号器によって実行され得る復号処理を複製し、再構築された参照フレームを参照ピクチャメモリ(434)に記憶させることができる。このようにして、符号化器(203)は、遠端ビデオ復号器によって得られる(送信エラーがない)再構築参照フレームとして共通のコンテンツを有する再構築参照フレームの複製をローカルに格納し得る。

40

#### 【0067】

予測器(435)は、符号化エンジン(432)の予測検索を実行し得る。すなわち、符号化される新しいフレームについて、予測器(435)は、(候補参照ピクセルブロックとしての)サンプルデータまたは参照ピクチャの動作ベクトル、ブロック形状などの、新しいピクチャの適切な予測参照として機能する特定のメタデータについて参照ピクチャメモリ(434)を検索することができる。予測器(435)は、適切な予測参照を見つけるために、サンプルブロック - ピクセルブロック毎に動作し得る。いくつかの場合において、予測

50

器（435）によって得られた検索結果によって決定されるように、入力ピクチャは、参照ピクチャメモリ（434）に記憶された複数の参照ピクチャから引き出された予測参照を有し得る。

#### 【0068】

コントローラ（450）は、例えば、ビデオデータを符号化するために使用されるパラメータおよびサブグループパラメータの設定を含む、ビデオ符号化器（430）の符号化動作を管理し得る。

#### 【0069】

前述のすべての機能ユニットの出力は、エントロピー符号化器（445）においてエントロピー符号化を受けることができる。エントロピー符号化器は、例えばハフマン符号化、可変長符号化、算術符号化などの当業者に知られている技術に従ってシンボルを可逆圧縮することにより、様々な機能ユニットにより生成されたシンボルを符号化されたビデオシーケンスに変換する。

10

#### 【0070】

送信機（440）は、エントロピー符号化器（445）によって作成された符号化されたビデオシーケンスをバッファリングして、符号化されたビデオデータを格納する記憶デバイスへのハードウェア／ソフトウェアリンクであり得る通信チャネル（460）を介した送信に備えることができる。送信機（440）は、ビデオ符号化器（430）からの符号化されたビデオデータを、送信される他のデータ、例えば符号化音声データおよび／または補助データストリーム（ソースは図示せず）とマージすることができる。

20

#### 【0071】

コントローラ（450）は、符号化器（203）の動作を管理し得る。符号化中に、コントローラ（450）は、各々の符号化されたピクチャに特定の符号化されたピクチャ型式を割り当て得、これは、それぞれのピクチャに適用され得る符号化技法に影響を及ぼし得る。例えば、ピクチャは、イントラピクチャ（Iピクチャ）、予測ピクチャ（Pピクチャ）、または双方向予測ピクチャ（Bピクチャ）として割り当てられることが多い。

#### 【0072】

イントラピクチャ（Iピクチャ）は、シーケンス内の他のフレームを予測のソースとして使用せずに符号化および復号できるものである。いくつかのビデオコーデックは、例えば、独立復号器リフレッシュ（IDR）ピクチャを含む異なる型式のイントラピクチャを可能にする。当業者は、Iピクチャのこれらの変形ならびにそれらのそれぞれの用途および特徴を認識している。

30

#### 【0073】

予測ピクチャ（Pピクチャ）は、各ブロックのサンプル値を予測するために最大で1つの動作ベクトルおよび参照インデックスを使用するイントラ予測またはインター予測を使用して符号化および復号され得るものであり得る。

#### 【0074】

双方向予測ピクチャ（Bピクチャ）は、各ブロックのサンプル値を予測するために最大で2つの動作ベクトルおよび参照インデックスを使用するイントラ予測またはインター予測を使用して符号化および復号され得るものであり得る。同様に、複数の予測ピクチャは、単一のブロックの再構築のために3つ以上の参照ピクチャおよび関連するメタデータを使用することができる。

40

#### 【0075】

ソースピクチャは、通常、空間的に複数のサンプルブロック（例えば、それぞれ $4 \times 4$ 、 $8 \times 8$ 、 $4 \times 8$ 、または $16 \times 16$ サンプルのブロック）に細分化され、ブロック毎に符号化され得る。ブロックは、ブロックのそれぞれのピクチャに適用される符号化割り当てによって決定されるように、他の（すでに符号化された）ブロックを参照して予測的に符号化され得る。例えば、Iピクチャのブロックは非予測的に符号化されてもよく、またはそれらは同じピクチャの既に符号化されたブロックを参照して予測的に符号化されてもよい（空間予測またはイントラ予測）。Pピクチャのピクセルブロックは、以前に符号化された1つの

50

参照ピクチャを参照して、空間的予測を介して、または時間的予測を介して、非予測的に符号化され得る。Bピクチャのブロックは、1つまたは2つの以前に符号化された参照ピクチャを参照して、空間的予測を介して、または時間的予測を介して、非予測的に符号化され得る。

#### 【0076】

ビデオ符号化器(203)は、ITU-T Rec. H.265などの所定のビデオ符号化技術または規格に従って符号化動作を実行し得る。その動作において、ビデオ符号化器(203)は、入力ビデオシーケンスの時間的および空間的冗長性を活用する予測符号化操作を含む、さまざまな圧縮操作を実行し得る。したがって、符号化されたビデオデータは、使用されているビデオ符号化技術または規格によって指定された構文に準拠することができる。

10

#### 【0077】

一実施形態では、送信機(440)は、符号化されたビデオとともに追加のデータを送信し得る。ビデオ符号化器(430)は、そのようなデータを、符号化されたビデオシーケンスの一部として含み得る。追加のデータには、時間的/空間的/SNR拡張層、冗長なピクチャやスライスなどの冗長データの他の形式、補足拡張情報(SEI)メッセージ、視覚ユーザビリティ情報(VUI)パラメータセットフラグメントなどが含まれ得る。

#### 【0078】

本開示の実施形態の特定の態様をより詳細に説明する前に、本明細書の残りの部分で参考されるいくつかの用語を以下に導入する。

#### 【0079】

以下、「サブピクチャ」とは、場合によっては、意味的にグループ化され、変更された解像度で独立して符号化され得るサンプル、ブロック、マクロブロック、符号化ユニット、または同様のエンティティの矩形配置を指す。1つまたは複数のサブピクチャがピクチャを形成することができる。1つまたは複数の符号化されたサブピクチャは、符号化されたピクチャを形成し得る。1つまたは複数のサブピクチャをピクチャにアセンブルすることができ、1つまたは複数のサブピクチャをピクチャから抽出することができる。ある特定の環境では、1つまたは複数の符号化されたサブピクチャは、サンプル・レベルへの符号化されたピクチャへのトランスコードを行わずに、圧縮された領域においてアセンブルされ得る。そして、同じまたはある特定の他のケースでは、1つまたは複数の符号化されたサブピクチャが、圧縮された領域における符号化されたピクチャから抽出され得る。

20

#### 【0080】

「適応解像度変更」(ARC)は、以降、例えば、参照ピクチャ再サンプリングによって、符号化されたビデオシーケンス内のピクチャまたはサブピクチャの解像度の変更を可能にする機構を称する。以下、「ARCパラメータ」は、適応的解像度変更を実行するために必要とされる制御情報を指し、これは、例えば、フィルタパラメータ、倍率、出力および/または参照ピクチャの解像度、様々な制御フラグなどを含み得る。

30

#### 【0081】

上記の説明は、単一の意味的に独立した符号化されたビデオピクチャの符号化および復号に焦点を当てている。独立したARCパラメータを有する複数のサブピクチャの符号化/復号の含意およびその含意される追加の複雑さを説明する前に、ARCパラメータをシグナリングするための実施形態を説明するものとする。

40

#### 【0082】

図6A～図6Cを参照すると、ARCパラメータをシグナリングするためのいくつかの新規の例示的な実施形態が示されている。各実施形態で述べたように、それらは、符号化効率、複雑さ、およびアーキテクチャの観点から特定の利点を有する。ビデオ符号化規格または技術は、これらの実施形態のうちの1つまたは複数を実装することができ、ARCパラメータをシグナリングするための比較技術から既知の実施形態も含むことができる。比較技術の実施形態は、図5A～図5Bに示す例を含む。新規な実施形態は、相互に排他的ではなく、アプリケーションのニーズ、関連する標準技術、または符号化器の選択に基づいていざれかが使用され得るように、比較技術の実施形態も含む標準または技術に含まれ得るこ

50

とが考えられる。

#### 【 0 0 8 3 】

ARCパラメータのクラスは、(1) X次元およびY次元で分離または結合されたアップ/ダウンサンプル係数、または(2)時間次元を追加した所与の数のピクチャの一定速度ズームイン/アウトを示すアップ/ダウンサンプル係数を含むことができる。上記の2つのいずれかは、要素を含む表を指すことができる1つまたは複数の構文要素の符号化または復号を含み得る。そのような構文要素は、実施形態では長さが短くてもよい。

#### 【 0 0 8 4 】

「解像度」は、入力ピクチャ、出力ピクチャ、参照ピクチャ、符号化ピクチャ、結合または個別の、サンプル、ロック、マクロロック、CU、または任意の他の適切な粒度の単位でのX次元またはY次元の解像度を指すことができる。2つ以上の分解能がある場合(例えば、入力ピクチャ用のもの、参照ピクチャ用のものなど)、特定の場合には、値の1つのセットを値の別のセットから推測することができる。解像度は、例えば、フラグの使用によってゲーティングすることができる。解像度のより詳細な例は、以下でさらに提供される。

10

#### 【 0 0 8 5 】

「ワーピング」座標は、H.263 Annex Pで使用されるものと同様に、上述したように適切な粒度であってもよい。H.263 Annex Pは、そのようなワーピング座標を符号化する1つの効率的な方法を定義しているが、他の潜在的により効率的な方法も考えられる。例えば、Annex Pのワーピング座標の可変長可逆的な「ハフマン」スタイルの符号化は、適切な長さの二値符号化に置き換えることができ、二値符号語の長さは、例えば、最大ピクチャサイズから導出することができ、場合によっては特定の係数を乗算し、特定の値をオフセットして、最大ピクチャサイズの境界外の「ワーピング」を可能にする。

20

#### 【 0 0 8 6 】

アップサンプル・フィルタ・パラメータまたはダウンサンプル・フィルタ・パラメータに関して、最も簡単な場合には、アップサンプリングおよび/またはダウンサンプリング用の単一のフィルタのみが存在してもよい。しかしながら、特定の場合には、フィルタパラメータのシグナリングによって実施することができるフィルタ設計のより多くの柔軟性を可能にすることが有利であり得る。そのようなパラメータは、可能なフィルタ設計のリスト内のインデックスを介して選択されてもよく、フィルタは、(例えば、適切なエンタロピー符号化技術を使用してフィルタ係数のリストを介して)完全に指定されてもよく、および/またはフィルタは、上述の機構のいずれかに従ってシグナリングされるアップ/ダウンサンプル比を介して暗黙的に選択されてもよい。

30

#### 【 0 0 8 7 】

以下では、符号語で示される有限集合のアップ/ダウンサンプル係数(X次元およびY次元の両方で使用される同じ係数)を符号化する場合を例にして説明する。その符号語は、有利には、例えば、H.264およびH.265などのビデオ符号化仕様における特定の構文要素に共通のExt-Golomb符号を使用することによって可変長符号化ができる。アップ/ダウンサンプル係数への値の1つの適切なマッピングは、例えば、以下の表1に従うことができる。

40

#### 【 0 0 8 8 】

50

## 【表1】

表1

符号語	Ext-Golomb 符号	オリジナル/ターゲット解像度
0	1	1/1
1	010	1/1.5 (50%アップスケール)
2	011	1.5/1 (50%ダウンスケール)
3	00100	1/2 (100%アップスケール)
4	00101	2/1 (100%ダウンスケール)

10

## 【0089】

アプリケーションのニーズ、およびビデオ圧縮技術または規格で利用可能なアップスケール機構およびダウンスケール機構の能力に従って、多くの同様のマッピングを考案することができる。表は、より多くの値に拡張することができる。値はまた、例えばMANEによって、再サンプリング係数がビデオ処理エンジン（何よりもまず復号器および符号化器）自体の外部で関心のあるものであった場合に特定の利点を有し得るExt - Golomb符号（例えば、二値符号化を使用する）以外のエントロピー符号化機構によって表され得る。解像度変更が必要とされない（おそらく）最も一般的なケースでは、最も一般的なケースで二値コードを使用するよりも符号化効率の利点を有することができる短い（例えば、表1の第2行に示されているように、単一のビットのみ）Ext - Golomb符号を選択することができることに留意されたい。

20

## 【0090】

表内のエントリの数、ならびにそれらのセマンティクスは、完全にまたは部分的に構成可能であり得る。例えば、表の基本的な概要は、シーケンスまたは復号器パラメータセットなどの「高」パラメータセットで伝達されてもよい。代替的または追加的に、1つまたは複数のそのような表は、ビデオ符号化技術または規格で定義されてもよく、例えば復号器またはシーケンス・パラメータ・セットを介して選択されてもよい。

30

## 【0091】

上記のように符号化されたアップサンプル / ダウンサンプル係数 (ARC情報) が、ビデオ符号化技術または標準構文にどのように含まれ得るかについて、以下に説明する。アップ / ダウンサンプルフィルタを制御する1つまたはいくつかの符号語にも同様の考慮事項が適用され得る。以下では、フィルタまたは他のデータ構造に比較的大量のデータが必要とされる場合に関する説明も提供される。

30

## 【0092】

図5Aを参照すると、H.263 Annex Pは、ピクチャヘッダ(501)内、具体的にはH.263 PLUSPTYPE(503)ヘッダ拡張内の4つのワーピング座標の形態のARC情報(502)を含む。このような設計は、(a)利用可能なピクチャヘッダがあり、かつ(b)ARC情報の頻繁な変更が予想される場合に、賢明となり得る。しかしながら、H.263スタイルのシグナリングを使用するときのオーバーヘッドは非常に高くなる可能性があり、ピクチャヘッダは一時的な性質であり得るため、スケーリング係数はピクチャ境界に関係しない可能性がある。

40

## 【0093】

図5Bを参照すると、JVCET-M 135-v1は、シーケンス・パラメータ・セット(507)内に位置するターゲット解像度を含む表(506)をインデックスする、ピクチャ・パラメータ・セット(504)内に位置するARC参照情報(505)(インデックス)を含む。シーケンス・パラメータ・セット(507)内の表(506)内の可能な解決策の配置は、機能交換中の相互運用交渉ポイントとしてSPS(507)を使用することによって正当化され得

50

る。解像度は、適切なピクチャ・パラメータ・セット(504)を参照することによって、ピクチャ毎に表(506)内の値によって設定された制限内で変化し得る。

#### 【0094】

図6A～図6Cを参照すると、本開示の以下の実施形態は、ビデオ・ビットストリーム内のARC情報を、例えば、本開示の復号器に伝達することができる。これらの実施形態の各々は、上述の比較技術を超える特定の利点を有する。実施形態は、同じビデオ符号化技術または規格に同時に存在してもよい。

#### 【0095】

図6Aを参照する実施形態では、再サンプリング(ズーム)係数などのARC情報(509)は、例えばスライスヘッダ、GOBヘッダ、タイルヘッダ、またはタイル・グループ・ヘッダなどのヘッダ(508)に存在し得る。一例として、図6Aは、タイル・グループ・ヘッダとしてヘッダ(508)を示す。そのような構成は、例えば表1に示すように、単一の可変長ue(v)または数ビットの固定長符号語など、ARC情報が小さい場合に適切であり得る。ARC情報をタイル・グループ・ヘッダ内に直接有することは、ARC情報が、ピクチャ全体ではなく、例えばタイル・グループ・ヘッダに対応するタイルグループによって表されるサブピクチャに適用可能であり得るという追加の利点を有する。加えて、ビデオ圧縮技術または規格が(例えば、タイルグループベースの適応解像度の変化とは対照的に)全ピクチャ適応解像度変更のみを使用する場合でも、ARC情報をタイル・グループ・ヘッダ(例えば、H.263スタイルのピクチャヘッダに)に入れることは、エラー回復力の観点から特定の利点を有する。上述した説明では、ARC情報(509)が、タイル・グループ・ヘッダに存在する場合について説明したが、上述した説明は、ARC情報(509)が、例えば、スライスヘッダ、GOBヘッダ又はタイルヘッダに存在する場合にも同様に適用できることは言うまでもない。

10

20

30

#### 【0096】

図6Bを参照する同じまたは別の実施形態では、ARC情報(512)自体は、例えば、ピクチャ・パラメータ・セット、ヘッダ・パラメータ・セット、タイル・パラメータ・セット、適応パラメータセットなどの適切なパラメータセット(511)に存在し得る。一例として、図6Bは、適応パラメータセット(APS)としてパラメータセット(511)を示す。そのパラメータセットの範囲は、有利には、ピクチャ以下であり得る。例えば、パラメータセットの範囲はタイルグループであってもよい。ARC情報(512)の使用は、関連するパラメータセットの起動によって暗黙的であり得る。例えば、ビデオ符号化技術または規格がピクチャベースのARCのみを企図する場合、ピクチャ・パラメータ・セットまたは同等物が関連するパラメータセットとして適切であり得る。

40

#### 【0097】

図6Cを参照する同じまたは別の実施形態では、ARC参照情報(513)は、タイル・グループ・ヘッダ(514)または類似のデータ構造に存在し得る。ARC参照情報(513)は、単一ピクチャを超えるスコープを有するパラメータセット(516)において利用可能なARC情報(515)のサブセットを指すことができる。例えば、パラメータセット(516)は、シーケンス・パラメータ・セット(SPS)または復号器パラメータセット(DPS)であってもよい。

50

#### 【0098】

JVET - M 0135 - v1で使用されるタイル・グループ・ヘッダ、PPS、またはSPSからのPPSの間接的な追加レベルの暗示的なアクティブ化は、シーケンス・パラメータ・セットと同様に、ピクチャ・パラメータ・セットが機能交渉またはアナウンスに使用され得るため、不要であり得る。しかしながら、ARC情報が、例えばタイルグループによっても表されるサブピクチャに適用可能であるべきである場合、起動範囲がタイルグループに限定されたパラメータセット(例えば、適応パラメータセットまたはヘッダ・パラメータ・セット)がより良い選択であり得る。また、ARC情報が無視できないサイズである場合、例えば、多数のフィルタ係数などのフィルタ制御情報を含む場合、パラメータは、符号化効率の観点から直接ヘッダを使用するよりも良い選択であり得る。なぜなら、これらの設定は

50

、同じパラメータセットを参照することによって将来のピクチャまたはサブピクチャによって再利用可能であり得るからである。

#### 【0099】

シーケンス・パラメータ・セットまたは複数のピクチャにまたがるスコープを有する別のより高いパラメータセットを使用するとき、特定の考慮事項が適用され得る。

#### 【0100】

(1) ARC情報(515)を表に格納するためのパラメータセット(516)は、いくつかの場合、シーケンス・パラメータ・セットとすることができますが、他の場合、復号器パラメータセットとすることができますが有利である。復号器パラメータセットは、複数のCVS、すなわち符号化されたビデオストリーム、すなわちセッション開始からセッション終了までのすべての符号化されたビデオビットのアクティブ化範囲を有することができる。そのような範囲は、可能性のあるARC因子が、おそらくハードウェアに実装される復号器機能であり得、ハードウェア機能は、CVSによって変化しない傾向があるため、より適切であり得る(少なくともいくつかの娛樂システムでは、長さが1/2以下のグループ・オブ・ピクチャである)。それにもかかわらず、いくつかの実施形態は、特に以下の点(2)に関連して、本明細書に記載のシーケンス・パラメータ・セットにARC情報表を含むことができる。

10

#### 【0101】

(2) ARC参照情報(513)は、有利には、JVCT - M 0135 - v1のようにピクチャ・パラメータ・セット内ではなくヘッダ(514)(例えば、ピクチャ/スタイル/GO B/タイル・グループ・ヘッダ; 以降、タイル・グループ・ヘッダ)内に直接配置されてもよい。その理由は、符号化器が、例えばARC参照情報のようなピクチャ・パラメータ・セット内の単一の値を変更したい場合、符号化器は、新しいPPSを作成し、その新しいPPSを参照しなければならない場合があるからである。ARC参照情報のみが変化するが、例えばPPS内の量子化行列情報などの他の情報が残っている場合、そのような情報はかなりのサイズである可能性があり、新しいPPSを完成させるために再送信される必要がある。ARC参照情報は、変化する唯一の値であるARC情報表へのインデックスのような単一の符号語であってもよいので、例えばすべての量子化行列情報を再送信することは面倒で無駄である。したがって、ARC参照情報をヘッダ(例えば、ヘッダ(514))に直接配置することは、JVET - M 0135 - v1において提案されているように、PPSを介した間接化を回避することができるので、符号化効率の観点からかなり良好であり得る。また、ARC参照情報をPPSに入れることは、ピクチャ・パラメータ・セット・アクティブ化の範囲がピクチャであるため、ARC参照情報によって参照されるARC情報が、サブピクチャではなく、ピクチャ全体に適用される必要があるというさらなる欠点を有する。

20

30

#### 【0102】

同じまたは別の実施形態では、ARCパラメータのシグナリングは、図7A～図7Bに概説されているような詳細な例に従うことができる。図7A～図7Bは、構文図を示す。このような構文図の表記は、大雑把にはCスタイルプログラミングに従う。太字の線はビットストリームに存在する構文要素を示し、太字のない線は制御フローまたは変数の設定を示すことが多い。

40

#### 【0103】

ピクチャの(おそらく長方形の)部分に適用可能なヘッダの構文構造の例として、タイル・グループ・ヘッダ(600)は、条件付きで、可変長のExp-Golomb符号化構文要素dec\_pic\_size\_idx(602)(太字で示されている)を含むことができる。タイル・グループ・ヘッダ(600)におけるこの構文要素の存在は、適応型解像度(603)の使用によってゲーティングすることができる。ここで、適応解像度フラグの値は太字で示されておらず、これは、フラグが構文図内で発生する点でビットストリーム内に存在することを意味する。適応型解像度がこのピクチャまたはその一部に使用されているか否かは、ビットストリームの内部または外部の任意の高レベル構文構造でシグナリングすることができる。図7A～図7Bに示す例では、適応解像度は、以下に概説するようにシーケンス・パラメ

50

ータ・セット(610)でシグナリングされる。

#### 【0104】

図7Bは、シーケンス・パラメータ・セット(610)の抜粋を示す。示されている第一の構文要素は、adaptive\_pic\_resolution\_change\_flag(611)である。真の場合、そのようなフラグは適応解像度の使用を示すことができ、適応解像度は特定の制御情報を必要とする場合がある。この例では、このような制御情報は、シーケンス・パラメータ・セット(610)内のif()文(612)に基づくフラグの値と、タイル・グループ・ヘッダ(600)とに基づいて、条件付きで存在する。

#### 【0105】

適応解像度が使用されているとき、この例では、符号化はサンプル単位の出力解像度(613)である。この例示的な実施形態における出力解像度(613)は、出力ピクチャの解像度を共に定義することができる構文要素output\_pic\_width\_in\_luma\_samplesおよびoutput\_pic\_height\_in\_luma\_samplesの両方を指す。ビデオ符号化技術または規格の他の場所では、いずれかの値に対する特定の制限を定義することができる。例えば、レベル定義は、上記の2つの構文要素の値の積とすることができます総出力サンプルの数を制限することができる。また、特定のビデオ符号化技術もしくは規格、または例えばシステム規格などの外部技術もしくは規格は、番号付け範囲(例えば、一方または両方の寸法は、2の累乗で割り切れなければならない)、またはアスペクト比(例えば、幅と高さは4:3または16:9などの関係になければならない)を制限することができる。そのような制限は、ハードウェア実装を容易にするために、または他の理由のために導入され得る。

10

#### 【0106】

特定の用途では、符号化器は、出力ピクチャサイズであるサイズを暗黙的に仮定するのではなく、特定の参照ピクチャサイズを使用するように復号器に指示することが望ましい場合がある。この例では、構文要素reference\_pic\_size\_present\_flag(614)は、参照ピクチャ寸法(615)(この場合も、数字は、例示的な実施形態では幅と高さの両方を指す)の条件付き存在をゲートする。

20

#### 【0107】

図7Bは、可能な復号ピクチャの幅および高さの表をさらに示す。このような表は、例えば、表指示(616)(例えば、構文要素num\_dec\_pic\_size\_in\_luma\_samples\_minus1)によって表すことができる。構文要素の「minus1」は、その構文要素の値の解釈を指すことができる。例えば、構文要素の符号化値が0である場合、1つの表エントリが存在する。符号化された値が5である場合、6つの表エントリが存在する。表内の各「線」について、復号されたピクチャの幅および高さが次に表エントリ(617)として構文に含まれる。

30

#### 【0108】

提示された表エントリ(617)は、タイル・グループ・ヘッダ(600)内の構文要素dec\_pic\_size\_idx(602)を使用してインデックス付けすることができ、それにより、タイルグループ毎に異なる復号サイズ、実際にはズーム率を可能にする。

40

#### 【0109】

例えばVP9のようなあるビデオ符号化技術または規格は、空間スケーラビリティを可能にするために、(本開示の実施形態とは全く異なるようにシグナルされ得る)ある形式の参照ピクチャ再サンプリングを時間スケーラビリティと併せて実施することによって、空間スケーラビリティをサポートする。特に、特定の参照ピクチャは、ARCスタイル技術を使用してより高い解像度にアップサンプリングされ、空間拡張層のベースを形成することができる。そのようなアップサンプリングされたピクチャは、詳細を追加するために、高解像度で通常の予測機構を使用して洗練されることができる。

#### 【0110】

本開示の実施形態は、そのような環境で使用することができる。場合によっては、同じまたは別の実施形態では、ネットワーク抽象化層(NAL)ユニットヘッダ内の値、例えば

50

時間IDフィールドを使用して、時間層だけでなく空間層も示すことができる。そうすることは、特定のシステム設計において特定の利点を有する。例えば、NALユニットヘッダ時間ID値に基づいて時間層選択転送のために作成および最適化された既存の選択転送ユニット（Selected Forwarding Units : SFU）は、スケーラブル環境のために変更することなく使用することができる。これを可能にするために、本開示の実施形態は、符号化されたピクチャサイズと、NALユニットヘッダ内の時間IDフィールドによって示される時間層との間のマッピングを含むことができる。

#### 【0111】

いくつかのビデオ符号化技術では、アクセスユニット（AU）は、所与の時間インスタンスにおいてそれぞれのピクチャ / スライス / タイル / NALユニットビットストリームに取り込まれて構成された、符号化されたピクチャ、スライス、タイル、NALユニットなどを参照することができる。そのような時間的な例は、合成時間であり得る。

10

#### 【0112】

高効率ビデオ符号化（High Efficiency Video Coding : HEVC）、および特定の他のビデオ符号化技術では、復号ピクチャバッファ（decoded picture buffer : DPB）に格納された複数の参照ピクチャの中から選択された参照ピクチャを示すためにピクチャ順序カウント（picture order count : POC）値を使用することができる。アクセスユニット（AU）が1つまたは複数のピクチャ、スライス、またはタイルを含む場合、同じAUに属する各ピクチャ、スライス、またはタイルは、同じPOC値を有することができ、そこから、それらが同じ合成時間のコンテンツから作成されたことを導出することができる。言い換えば、2つのピクチャ / スライス / タイルが同じ所与のPOC値を搬送するシナリオにおいて、2つのピクチャ / スライス / タイルが同じAUに属し、同じ合成時間を有すると判定することができる。逆に、異なるPOC値を有する2つのピクチャ / タイル / スライスは、それらのピクチャ / スライス / タイルが異なるAUに属し、異なる合成時間を有することを示すことができる。

20

#### 【0113】

本開示の一実施形態では、アクセスユニットが異なるPOC値を有するピクチャ、スライス、またはタイルを含むことができるという点で、前述の硬直した関係を緩和することができる。AU内で異なるPOC値を許容することにより、POC値を使用して、同一の提示時間是有する、潜在的に独立して復号可能なピクチャ / スライス / タイルを識別することができる。したがって、本開示の実施形態は、以下により詳細に記載されるように、参照ピクチャ選択シグナリング（例えば、参照ピクチャ・セット・シグナリングまたは参照ピクチャ・リスト・シグナリング）を変更することなく、複数のスケーラブル層のサポートを可能にすることができます。

30

#### 【0114】

一実施形態では、POC値のみから、異なるPOC値を有する他のピクチャ / スライス / タイルに対して、ピクチャ / スライス / タイルが属するAUを識別できることが依然として望ましい。これは、以下に説明する実施形態で達成することができる。

#### 【0115】

同じまたは他の実施形態では、アクセス・ユニット・カウント（AUC）は、NALユニットヘッダ、スライスヘッダ、タイル・グループ・ヘッダ、SEIメッセージ、パラメータセットまたはAUデリミタなどの高レベル構文構造でシグナリングされてもよい。AUCの値は、どのNALユニット、ピクチャ、スライス、またはタイルが所与のAUに属するかを識別するために使用され得る。AUCの値は、別個の合成時間インスタンスに対応し得る。AUC値は、POC値の倍数に等しくてもよい。POC値を整数値で割ることにより、AUC値を算出することができる。場合によっては、分割演算は、復号器の実装に一定の負担をかける可能性がある。そのような場合、AUC値の番号付けスペースの小さな制限は、本開示の実施形態によって実行されるシフト演算による分割演算の置換を可能にすることができる。例えば、AUC値は、POC値範囲の最上位ビット（MSB）値に等しくてもよい。

40

#### 【0116】

50

同じ実施形態において、AU毎のPOCサイクルの値（例えば、構文要素poc\_cycle\_au）は、NALユニットヘッダ、スライスヘッダ、タイル・グループ・ヘッダ、SEIメッセージ、パラメータセットまたはAUデリミタなどの高レベル構文構造でシグナリングされてもよい。poc\_cycle\_au構文要素は、いくつの異なる連続するPOC値を同じAUに関連付けることができるかを示すことができる。例えば、poc\_cycle\_auの値が4に等しい場合、0 - 3に等しいPOC値を有するピクチャ、スライスまたはタイルは、0に等しいAUC値を有するAUに関連付けられ、4 - 7に等しいPOC値を有するピクチャ、スライスまたはタイルは、1に等しいAUC値を有するAUに関連付けられる。したがって、AUCの値は、POC値をpoc\_cycle\_auの値で除算することによって本開示の実施形態によって推測することができる。

10

#### 【0117】

同じまたは別の実施形態では、poc\_cycle\_auの値は、例えばビデオ・パラメータ・セット（VPS）に位置する、符号化されたビデオシーケンス内の空間層またはSNR層の数を識別する情報から導出されてもよい。以下、このような関係を簡単に説明する。上述したような導出は、VPSにおいて数ビットを節約することができ、したがって符号化効率を改善することができるが、ビデオ・パラメータ・セットの下位に階層的に適切な高レベル構文構造でpoc\_cycle\_auを明示的に符号化して、ピクチャなどのビットストリームの所与の小部分についてpoc\_cycle\_auを最小化することができることが有利であり得る。この最適化は、POC値（および／またはPOCを間接的に参照する構文要素の値）が低レベルの構文構造で符号化され得るため、上記の導出処理を通して節約できるよりも多くのビットを節約することができる。

20

#### 【0118】

同じまたは別の実施形態では、図9Aは、符号化されたビデオシーケンス内のすべてのピクチャ／スライスに使用されるpoc\_cycle\_auを示す、VPS（630）またはSPS内のvps\_poc\_cycle\_au（632）の構文要素をシグナリングするための構文表の例を示し、図9Bは、スライスヘッダ（640）内の現在のスライスのpoc\_cycle\_auを示す、slice\_poc\_cycle\_au（642）の構文要素をシグナリングするための構文表の例を示す。POC値がAU毎に一律に増加する場合、VPS（630）におけるvps\_contant\_poc\_cycle\_per\_au（634）を1に設定し、VPS（630）においてvps\_poc\_cycle\_au（632）をシグナリングする。この場合、slice\_poc\_cycle\_au（642）は明示的にシグナリングされず、POCの値をvps\_poc\_cycle\_au（632）で除算することにより、AU毎のAUCの値が算出される。POC値がAU毎に一律に増加しない場合、VPS（630）におけるvps\_contant\_poc\_cycle\_per\_au（634）は0に設定される。この場合、vps\_access\_unit\_cntはシグナリングされず、slice\_access\_unit\_cntは、スライスまたはピクチャ毎にスライスヘッダでシグナリングされる。各スライスまたはピクチャは、異なる値のslice\_access\_unit\_cntを有してもよい。AU毎のAUCの値は、POCの値をslice\_poc\_cycle\_au（642）で除算することにより算出される。

30

#### 【0119】

図10は、実施形態の関連する作業フローを説明するためのブロック図を示す。例えば、復号器（または符号化器）は、VPS／SPSを解析して、AU毎のPOCサイクルが一定であるか否かを識別する（652）。続いて、復号器（または符号化器）は、AU毎のPOCサイクルが符号化されたビデオシーケンス内で一定であるか否かに基づいて決定を行う（654）。すなわち、復号器（または符号化器）は、AU毎のPOCサイクルが一定であれば、シーケンスレベルpoc\_cycle\_au値とPOC値からアクセス・ユニット・カウント値を算出する（656）。あるいは、AU毎のPOCサイクルが一定でない場合、復号器（または符号化器）は、ピクチャレベルpoc\_cycle\_au値とPOC値からアクセス・ユニット・カウント値を算出する（658）。いずれの場合も、復号器（または符号化器）は、次に、例えば、VPS／SPSを解析し、AU毎のPOCサイクルが一定であるか否かを識別することによって処理を繰り返すことができる（662）。

40

#### 【0120】

50

同じまたは他の実施形態では、ピクチャ、スライス、またはタイルのPOCの値が異なっていても、同じAUC値を有するAUに対応するピクチャ、スライス、またはタイルは、同じ復号または出力時間インスタンスに関連付けられ得る。したがって、同じAU内のピクチャ、スライス、またはタイルにわたる解析／復号間の依存関係なしに、同じAUに関連付けられたピクチャ、スライス、またはタイルのすべてまたはサブセットを並列に復号することができ、同時に出力することができる。

#### 【0121】

同じまたは他の実施形態では、ピクチャ、スライス、またはタイルのPOCの値が異なっていても、同じAUC値を有するAUに対応するピクチャ、スライス、またはタイルは、同じ合成／表示時間インスタンスに関連付けられ得る。合成時間がコンテナ形式に含まれている場合、ピクチャが異なるAUに対応していても、ピクチャが同じ合成時間有する場合、それらのピクチャは同じ時間インスタンスで表示され得る。10

#### 【0122】

同じまたは他の実施形態では、各ピクチャ、スライス、またはタイルは、同じAU内で同じ時間識別子（例えば、構文要素temporal\_id）を有することができる。時間インスタンスに対応するピクチャ、スライス、またはタイルのすべてまたはサブセットは、同じ時間サブ層に関連付けられ得る。同じまたは他の実施形態では、各ピクチャ、スライス、またはタイルは、同じAU内の同じまたは異なる空間層id（例えば、syntax element layer\_id）を有することができる。時間インスタンスに対応するピクチャ、スライス、またはタイルのすべてまたはサブセットは、同じまたは異なる空間層に関連付けられ得る。20

#### 【0123】

図8は、適応的な解像度変更を伴うtemporal\_id、layer\_id、およびPOC値とAUC値の組み合わせを有するビデオシーケンス構造(680)の一例を示す。この例では、AUC=0の第1のAU内のピクチャ、スライス、またはタイルは、temporal\_id=0およびlayer\_id=0または1を有することができるが、AUC=1の第2のAU内のピクチャ、スライス、またはタイルは、それぞれtemporal\_id=1およびlayer\_id=0または1を有することができる。temporal\_idおよびlayer\_idの値に関わらず、POCの値はピクチャ毎に1ずつ増加する。この例では、poc\_cycle\_auの値は2に等しくすることができる。一実施形態では、poc\_cycle\_auの値は、(空間スケーラビリティ)層の数に等しく設定されてもよい。この例では、POCの値は2だけ増加し、AUCの値は1だけ増加する。一例として、図8は、第1のAU(AUC=0)内で、POC 0、TID 0、およびLID 0を有するIスライス(681)と、POC 1、TID 0、およびLID 1を有するBスライス(682)とを示す。第2のAU(AUC=1)内で、図8は、POC 2、TID 1、およびLID 0を有するBスライス(683)と、POC 3、TID 1、およびLID 1を有するBスライス(684)とを示す。第3のAU(AUC=3)内で、図8は、POC 4、TID 0、およびLID 0を有するBスライス(685)と、POC 5、TID 0、およびLID 1を有するBスライス(686)とを示す。30

#### 【0124】

上記の実施形態では、ピクチャ間または層間予測構造および参照ピクチャ指示のすべてまたはサブセットは、HEVCにおける既存の参照ピクチャセット(RPS)シグナリングまたは参照ピクチャリスト(RPL)シグナリングを使用することによってサポートされ得る。RPSまたはRPLでは、選択された参照ピクチャは、現在のピクチャと選択された参照ピクチャとの間のPOCの値またはPOCのデルタ値をシグナリングすることによって示される。本開示の実施形態では、RPSおよびRPLを使用して、シグナリングを変更せずにピクチャ間または層間予測構造を示すことができるが、以下の制限がある。参照ピクチャのtemporal\_idの値が現在のピクチャのtemporal\_idの値より大きい場合、現在のピクチャは、動作補償または他の予測のために参照ピクチャを使用しない場合がある。参照ピクチャのlayer\_idの値が現在のピクチャのlayer\_idの値より大きい場合、現在のピクチャは、動作補償または他の予測のために参照ピクチャを使用しない場合がある。40

#### 【0125】

同じ実施形態および他の実施形態では、時間的動作ベクトル予測のためのPOC差に基づく

10

20

30

40

50

く動作ベクトルのスケーリングは、アクセスユニット内の複数のピクチャにわたって無効にすることができる。したがって、各ピクチャは、アクセスユニット内で異なるPOC値を有し得るが、同じAU内で異なるPOCを有する参照ピクチャは、同じ時間インスタンスを有する参照ピクチャと考えられ得るので、動作ベクトルは、スケールされず、アクセスユニット内での時間動作ベクトル予測のために使用されない。したがって、実施形態では、動作ベクトルスケーリング関数は、参照ピクチャが現在のピクチャに関連付けられたAUに属するときに1を返すことができる。

#### 【0126】

同じ実施形態および他の実施形態では、参照ピクチャの空間解像度が現在のピクチャの空間解像度と異なる場合、時間動作ベクトル予測のためのPOC差に基づく動作ベクトルのスケーリングは、複数のピクチャにわたって任意選択的に無効にすることができる。動作ベクトルのスケーリングが可能である場合、動作ベクトルは、現在のピクチャと参照ピクチャとの間のPOC差および空間解像度比の両方にに基づいてスケーリングされ得る。

10

#### 【0127】

同じまたは別の実施形態では、動作ベクトルは、特にpoc\_cycle\_auが不均一な値を有する場合(vps\_contant\_poc\_cycle\_per\_au == 0の場合)、時間的な動作ベクトル予測のためにPOC差の代わりにAUC差に基づいてスケーリングされてもよい。そうでない場合(vps\_contant\_poc\_cycle\_per\_au == 1の場合)、AUC差分に基づく動作ベクトルのスケーリングは、POC差分に基づく動作ベクトルのスケーリングと同一であり得る。

20

#### 【0128】

同じまたは別の実施形態では、動作ベクトルがAUC差に基づいてスケーリングされるとき、現在のピクチャと同じAU(同じAUC値を有する)内の参照動作ベクトルは、AUC差に基づいてスケーリングされず、現在のピクチャと参照ピクチャとの間の空間解像度比に基づくスケーリングなしまたはスケーリングありの動作ベクトル予測に使用される。

#### 【0129】

同じ実施形態および他の実施形態において、AUC値は、AUの境界を識別するために使用され、AU粒度を有する入力および出力タイミングを必要とする仮想参照復号器(HRD)動作のために使用される。ほとんどの場合、AU内の最上位層を持つ復号されたピクチャが、表示のために出力され得る。AUC値およびlayer\_id値は、出力ピクチャの識別に用いることができる。

30

#### 【0130】

一実施形態では、ピクチャは1つまたは複数のサブピクチャを含むことができる。各サブピクチャは、ピクチャの局所領域または全領域をカバーすることができる。サブピクチャによってサポートされる領域は、別のサブピクチャによってサポートされる領域と重なっていてもよいし、重なっていないなくてもよい。1つまたは複数のサブピクチャによって構成される領域は、ピクチャの全領域をカバーしてもしなくてもよい。ピクチャがサブピクチャからなる場合、サブピクチャによってサポートされる領域は、ピクチャによってサポートされる領域と同一であり得る。

40

#### 【0131】

同じ実施形態において、サブピクチャは、符号化ピクチャに使用される符号化方法と同様の符号化方法によって符号化されてもよい。サブピクチャは、独立して符号化され得るか、または、別のサブピクチャまたは符号化されたピクチャに依存して符号化され得る。サブピクチャは、別のサブピクチャまたは符号化されたピクチャからの解析依存性を有しても、有しなくてもよい。

#### 【0132】

同じ実施形態では、符号化されたサブピクチャは、1つまたは複数の層に含まれてもよい。層内の符号化されたサブピクチャは、異なる空間解像度を有することができる。元のサブピクチャは、空間的に再サンプリング(アップサンプリングまたはダウンサンプリング)され、異なる空間解像度パラメータで符号化され、層に対応するビットストリームに

50

含まれ得る。

#### 【0133】

同じ実施形態または別の実施形態において、(W, H)を有するサブピクチャが符号化されて層0に対応する符号化ビットストリームに含まれることができる一方で、(W \* S<sub>w,k</sub>, H \* S<sub>h,k</sub>)を有する元の空間解像度を有するサブピクチャからアップサンプリングされた（またはダウンサンプリングされた）サブピクチャが符号化されて層kに対応する符号化ビットストリームに含まれることができ、S<sub>w,k</sub>, S<sub>h,k</sub>は水平方向および垂直方向のリサンプリング比を示す。S<sub>w,k</sub>, S<sub>h,k</sub>の値が1より大きい場合、再サンプリングはアップサンプリングに等しい。一方、S<sub>w,k</sub>, S<sub>h,k</sub>の値が1より小さい場合、再サンプリングはダウンサンプリングに等しい。

10

#### 【0134】

同じまたは別の実施形態では、層内の符号化されたサブピクチャは、同じサブピクチャまたは異なるサブピクチャ内の別の層内の符号化されたサブピクチャの視覚的品質とは異なる視覚的品質を有し得る。例えば、層n内のサブピクチャiは量子化パラメータQ<sub>i,n</sub>で符号化され、層m内のサブピクチャjは量子化パラメータQ<sub>j,m</sub>で符号化される。

#### 【0135】

同じまたは別の実施形態では、層内の符号化されたサブピクチャは、同じローカル領域の別の層内の符号化されたサブピクチャからの構文解析または復号の依存関係なしに、独立して復号可能であり得る。同じローカル領域の別のサブピクチャ層を参照することなく独立して復号可能であり得るサブピクチャ層は、独立したサブピクチャ層である。独立したサブピクチャ層内の符号化されたサブピクチャは、同じサブピクチャ層内の以前に符号化されたサブピクチャからの復号または構文解析の依存関係を有しても有しなくてもよいが、符号化されたサブピクチャは、別のサブピクチャ層内の符号化されたピクチャからのいかなる依存関係も有しなくてもよい。

20

#### 【0136】

同じ実施形態または別の実施形態では、層内の符号化されたサブピクチャは、同じローカル領域の別の層内の符号化されたサブピクチャからの任意の構文解析または復号依存性を伴って、依存して復号可能であり得る。同じローカル領域の別のサブピクチャ層を参照して従属的に復号可能であり得るサブピクチャ層は、従属サブピクチャ層である。依存サブピクチャ内の符号化されたサブピクチャは、同じサブピクチャに属する符号化されたサブピクチャ、同じサブピクチャ層内の以前に符号化されたサブピクチャ、または両方の参照サブピクチャを参照することができる。

30

#### 【0137】

同じまたは別の実施形態では、符号化されたサブピクチャは、1つまたは複数の独立したサブピクチャ層および1つまたは複数の依存サブピクチャ層を含む。しかしながら、符号化されたサブピクチャに対して少なくとも1つの独立したサブピクチャ層が存在してもよい。独立したサブピクチャ層は、0に等しい、NALユニットヘッダまたは別の高レベル構文構造に存在し得る層識別子（例えば、構文要素layer\_id）の値を有し得る。layer\_idが0に等しいサブピクチャ層は、基本サブピクチャ層であってもよい。

40

#### 【0138】

同じ実施形態または別の実施形態において、ピクチャは、1つまたは複数の前景サブピクチャおよび1つの背景サブピクチャを含むことができる。背景サブピクチャによってサポートされる領域は、ピクチャの領域と等しくてもよい。前景サブピクチャによってサポートされる領域は、背景サブピクチャによってサポートされる領域と重複してもよい。背景サブピクチャは基本サブピクチャ層であってもよく、前景サブピクチャはノンベース（拡張）サブピクチャ層であってもよい。1つまたは複数のノンベースサブピクチャ層は、復号のために同じベース層を参照することができる。layer\_idがaに等しい各ノンベースサブピクチャ層は、layer\_idがbに等しいノンベースサブピクチャ層を参照することができ、aはbより大きい。

#### 【0139】

50

同じ実施形態または別の実施形態において、ピクチャは、背景サブピクチャの有無にかかわらず、1つまたは複数の前景サブピクチャを含むことができる。各サブピクチャは、それ自身のベースサブピクチャ層および1つまたは複数のノンベース（拡張）層を有することができる。各基本サブピクチャ層は、1つまたは複数の非基本サブピクチャ層によって参照され得る。*layer\_id*がaに等しい各ノンベースサブピクチャ層は、*layer\_id*がbに等しいノンベースサブピクチャ層を参照することができ、aはbより大きい。

#### 【0140】

同じ実施形態または別の実施形態において、ピクチャは、背景サブピクチャの有無にかかわらず、1つまたは複数の前景サブピクチャを含むことができる。（ベースまたはノンベース）サブピクチャ層内の各符号化されたサブピクチャは、同じサブピクチャに属する1つまたは複数のノンベース層サブピクチャと、同じサブピクチャに属さない1つまたは複数のノンベース層サブピクチャとによって参照され得る。

10

#### 【0141】

同じ実施形態または別の実施形態において、ピクチャは、背景サブピクチャの有無にかかわらず、1つまたは複数の前景サブピクチャを含むことができる。層a内のサブピクチャは、同じ層内の複数のサブピクチャにさらに分割され得る。層b内の1つまたは複数の符号化されたサブピクチャは、層a内の分割サブピクチャを参照することができる。

#### 【0142】

同じ実施形態または別の実施形態において、符号化されたビデオシーケンス（CVS）は、符号化されたピクチャのグループであってもよい。CVSは、1つまたは複数の符号化されたサブピクチャシーケンス（coded sub-picture sequences : CSPS）を含むことができ、CSPSは、ピクチャの同じローカル領域をカバーする符号化されたサブピクチャのグループであり得る。CSPSは、符号化されたビデオシーケンスの時間解像度と同じまたは異なる時間解像度を有し得る。

20

#### 【0143】

同じまたは別の実施形態では、CSPSは符号化され、1つまたは複数の層に含まれてもよい。CSPSは、1つまたは複数のCSPS層を含むか、またはそれらからなり得る。CSPSに対応する1つまたは複数のCSPS層を復号することは、同じローカル領域に対応するサブピクチャのシーケンスを再構築することができる。

30

#### 【0144】

同じまたは別の実施形態では、CSPSに対応するCSPS層の数は、別のCSPSに対応するCSPS層の数と同一であっても異なっていてもよい。

#### 【0145】

同じまたは別の実施形態では、CSPS層は、別のCSPS層とは異なる時間分解能（例えば、フレームレート）を有し得る。元の（圧縮されていない）サブピクチャシーケンスは、時間的に再サンプリング（アップサンプリングまたはダウンサンプリング）され、異なる時間解像度パラメータで符号化され、層に対応するビットストリームに含まれ得る。

#### 【0146】

同じ実施形態または別の実施形態において、フレームレートFを有するサブピクチャシーケンスが符号化され、層0に対応する符号化ビットストリームに含まれてもよく、 $F * S_{t,k}$ を有する元のサブピクチャシーケンスからの時間的にアップサンプリングされた（またはダウンサンプリングされた）サブピクチャシーケンスが符号化され、層kに対応する符号化ビットストリームに含まれてもよく、 $S_{t,k}$ は層kの時間的サンプリング比を示す。 $S_{t,k}$ の値が1より大きい場合、時間的リサンプリング処理はフレームレートアップ変換に等しい。一方、 $S_{t,k}$ の値が1より小さい場合、時間的リサンプリング処理はフレームレートダウン変換に等しい。

40

#### 【0147】

同じまたは別の実施形態では、CSPS層aを有するサブピクチャが、動作補償または任意の層間予測のためにCSPS層bを有するサブピクチャによって参照されるとき、CSPS層aの空間解像度がCSPS層bの空間解像度と異なる場合、CSPS層a内の復号されたピクセルが再

50

サンプリングされ、参照に使用される。再サンプリング処理は、アップ・サンプリング・フィルタリングまたはダウン・サンプリング・フィルタリングを必要とする場合がある。

#### 【0148】

図11は、layer\_idが0に等しい背景ビデオCSPSおよび複数の前景CSPS層を含む例示的なビデオストリームを示す。符号化されたサブピクチャは、1つまたは複数の拡張CSPS層(704)を備え得るが、どの前景CSPS層にも属さない背景領域は、ベース層(702)を備え得る。ベース層(702)は背景領域および前景領域を含むことができ、一方、拡張CSPS層(704)は前景領域を含む。拡張CSPS層(704)は、同じ領域において、ベース層(702)よりも良好な視覚的品質を有し得る。拡張CSPS層(704)は、同じ領域に対応する、再構築されたピクセルおよびベース層(702)の動作ベクトルを参照することができる。

10

#### 【0149】

同じまたは別の実施形態では、ベース層(702)に対応するビデオ・ビットストリームはトラックに含まれ、各サブピクチャに対応するCSPS層(704)はビデオファイル内の分離されたトラックに含まれる。

#### 【0150】

同じまたは別の実施形態では、ベース層(702)に対応するビデオ・ビットストリームはトラックに含まれ、同じlayer\_idを有するCSPS層(704)は分離されたトラックに含まれる。この例では、層kに対応するトラックは、層kに対応するCSPS層(704)のみを含む。

20

#### 【0151】

同じまたは別の実施形態では、各サブピクチャの各CSPS層(704)は、別個のトラックに格納される。各トラックは、1つまたは複数の他のトラックからの構文解析または復号の依存関係を有しても、有しなくてもよい。

#### 【0152】

同じまたは別の実施形態では、各トラックは、サブピクチャのすべてまたはサブセットのCSPS層(704)の層iから層jに対応するビットストリームを含むことができ、 $0 < i < j = k$ であり、kはCSPSの最上位層である。

#### 【0153】

同じまたは別の実施形態では、ピクチャは、デプスマップ、アルファマップ、3Dジオメトリデータ、占有マップなどを含む1つまたは複数の関連するメディアデータを含むか、またはそれらからなる。そのような関連する時限メディアデータは、それぞれが1つのサブピクチャに対応する1つまたは複数のデータサブストリームに分割することができる。

30

#### 【0154】

同じまたは別の実施形態では、図12は、多層サブピクチャ方法に基づくテレビ会議の一例を示す。ビデオストリームには、背景ピクチャに対応する1つのベース層ビデオ・ビットストリームと、前景サブピクチャに対応する1つまたは複数の拡張層ビデオ・ビットストリームとが含まれる。各拡張層ビデオ・ビットストリームは、CSPS層に対応することができる。ディスプレイには、ベース層(712)に対応するピクチャが初期設定で表示される。ベース層(712)は、ピクチャ(PIP)内に1つまたは複数のユーザのピクチャを含むことができる。クライアントの制御によって特定のユーザが選択されると、選択されたユーザに対応する拡張CSPS層(714)が符号化され、拡張品質または空間解像度で表示される。

40

#### 【0155】

図13は、実施形態の動作のための図を示す。実施形態では、復号器は、あるベース層および1つまたは複数の拡張CSPS層(722)などの複数の層を含むビデオ・ビットストリームを復号することができる。続いて、復号器は、背景領域および1つまたは複数の前景サブピクチャを識別し(724)、特定のサブピクチャ領域が選択されるか否かに関する決定を行うことができる(726)。例えば、ユーザのPIPに対応する特定のサブピクチャ領域が選択された場合(はい)、復号器は、選択されたユーザに対応する拡張サブピ

50

クチャを復号して表示することができる(728)。例えば、復号器は、拡張CSPS層(714)に対応する画像を復号して表示し得る。特定のサブピクチャ領域が選択されていない場合(いいえ)、復号器は背景領域を復号して表示することができる(730)。例えば、復号器は、ベース層(712)に対応する画像を復号して表示してもよい。

#### 【0156】

同じまたは別の実施形態では、ネットワーク中間ボックス(ルータなど)は、その帯域幅に応じて、ユーザに送信する層のサブセットを選択することができる。ピクチャ/サブピクチャ編成は、帯域幅適応のために使用され得る。例えば、ユーザが帯域幅を持っていない場合、ルータは、それらの重要性に起因して、または使用された設定に基づいて、層のストリップまたはいくつかのサブピクチャを選択する。一実施形態では、そのような処理は、帯域幅に適応するために動的に行われてもよい。

10

#### 【0157】

図14は、360ビデオの例示的なユースケースを示す。球面360ピクチャ(742)が平面ピクチャ上に投影される場合、投影される球面360ピクチャ(742)は、ベース層(744)として複数のサブピクチャ(745)に分割され得る。サブピクチャ(745)のうちの特定のサブピクチャの拡張層(746)は、符号化され、クライアントへ送信され得る。復号器は、すべてのサブピクチャ(745)を含むベース層(744)と、サブピクチャ(745)のうちの選択された一方の拡張層(746)の両方を復号し得る。現在のビューポートがサブピクチャ(745)のうちの選択されたものと同一であるとき、表示されたピクチャは、拡張層(746)を有する復号されたサブピクチャ(745)を用いてより高い品質を有することができる。そうでなければ、ベース層(744)を有する復号されたピクチャをより低い品質で表示することができる。

20

#### 【0158】

同じまたは別の実施形態では、表示用の任意のレイアウト情報が補助情報(SEIメッセージまたはメタデータなど)としてファイルに存在してもよい。1つまたは複数の復号されたサブピクチャは、シグナリングされたレイアウト情報を応じて再配置および表示され得る。レイアウト情報は、ストリーミングサーバまたは放送局によってシグナリングされてもよいし、ネットワークエンティティまたはクラウドサーバによって再生成されてもよいし、ユーザのカスタマイズされた設定によって決定されてもよい。

30

#### 【0159】

一実施形態では、入力ピクチャが1つまたは複数の(長方形の)サブ領域に分割される場合、各サブ領域は独立した層として符号化され得る。ローカル領域に対応する各独立層は、一意のlayer\_id値を有することができる。各独立した層について、サブピクチャサイズおよび位置情報がシグナリングされ得る。例えば、ピクチャサイズ(幅、高さ)および左上隅のオフセット情報(x\_offset、y\_offset)がシグナリングされ得る。図15Aは、分割されたサブピクチャ(752)のレイアウトの一例を示し、図15Bは、サブピクチャ(752)のうちの1つの対応するサブピクチャサイズおよび位置情報の一例を示し、図16は、対応するピクチャ予測構造を示す。サブピクチャサイズ(複数可)およびサブピクチャ位置(複数可)を含むレイアウト情報は、パラメータセット(複数可)、スライスもしくはタイルグループのヘッダ、またはSEIメッセージなどの高レベル構文構造でシグナリングされ得る。

40

#### 【0160】

同じ実施形態において、独立した層に対応する各サブピクチャは、AU内にその一意のPOC値を有し得る。DPBに格納されたピクチャのうちの参照ピクチャをRPSまたはRPL構造の構文要素を用いて示す場合、層に対応する各サブピクチャのPOC値を用いてもよい。

#### 【0161】

同一または別の実施形態では、(層間)予測構造を示すために、layer\_idは使用されなくてもよく、POC(デルタ)値が使用されてもよい。

#### 【0162】

同じ実施形態では、層(またはローカル領域)に対応するNに等しいPOC値を有するサ

50

ピクチャは、動作補償予測のための同じ層（または同じローカル領域）に対応する、K + Nに等しいPOC値を有するサブピクチャの参照ピクチャとして使用されてもされなくてもよい。ほとんどの場合、数Kの値は、（独立した）層の最大数に等しくてもよく、これはサブ領域の数と同一であってもよい。

#### 【0163】

同じまたは別の実施形態では、図17～図18は、図15A～図15Bおよび図16の拡張ケースを示す。入力ピクチャが複数の（例えば4）サブ領域に分割される場合、各ローカル領域は1つまたは複数の層で符号化され得る。この場合、独立した層の数はサブ領域の数に等しくてもよく、1つまたは複数の層がサブ領域に対応してもよい。したがって、各サブ領域は、1または複数の独立した層および0または複数の従属層を用いて符号化され得る。

10

#### 【0164】

同じ実施形態では、図17を参照すると、入力ピクチャは、左上サブ領域（762）、右上サブ領域（763）、左下サブ領域（764）、および右下サブ領域（765）を含む4つのサブ領域に分割され得る。右上サブ領域（763）は、層1および層4である2つの層として符号化することができ、右下サブ領域（765）は、層3および層5である2つの層として符号化することができる。この場合、層4は、動作補償予測のために層1を参照することができ、層5は、動作補償のために層3を参照することができる。

#### 【0165】

同じまたは別の実施形態では、層境界にわたるインループフィルタリング（デブロックフィルタリング、適応インループフィルタリング、リシェーパ、バイラテラルフィルタリング、または任意のディープラーニングベースのフィルタリングなど）は、（任意選択的に）無効にすることができる。

20

#### 【0166】

同じまたは別の実施形態では、層境界にわたる動作補償予測またはブロック内複製は、（任意選択的に）無効にすることができる。

#### 【0167】

同じまたは別の実施形態では、サブピクチャの境界における動作補償予測またはインループフィルタリングのための境界パディングは、任意選択的に処理されてもよい。境界パディングが処理されるか否かを示すフラグは、パラメータセット（VPS、SPS、PPS、またはAPS）、スライスもしくはタイル・グループ・ヘッダ、またはSEIメッセージなどの高レベル構文構造でシグナリングすることができる。

30

#### 【0168】

同じまたは別の実施形態では、サブ領域（またはサブピクチャ）のレイアウト情報は、VPSまたはSPSでシグナリングされ得る。図19Aは、VPS（770）の構文要素の例を示し、図19Bは、SPS（780）の構文要素の例を示す。この例では、VPS（770）において、vps\_sub\_picture\_dividing\_flag（772）がシグナリングされる。フラグは、入力ピクチャが複数のサブ領域に分割されているか否かを示すことができる。vps\_sub\_picture\_dividing\_flag（772）の値が0に等しいとき、現在のVPSに対応する符号化されたビデオシーケンス内の入力ピクチャは複数のサブ領域に分割されなくてもよい。この場合、入力ピクチャサイズは、SPS（680）でシグナリングされる符号化ピクチャサイズ(pic\_width\_in\_luma\_samples(786)、pic\_height\_in\_luma\_samples(788))と等しくてもよい。vps\_sub\_picture\_dividing\_flag（772）の値が1に等しいとき、入力ピクチャは複数のサブ領域に分割され得る。この場合、構文要素vps\_full\_pic\_width\_in\_luma\_samples(774)およびvps\_full\_pic\_height\_in\_luma\_samples(776)がVPS（770）でシグナリングされる。vps\_full\_pic\_width\_in\_luma\_samples(774)およびvps\_full\_pic\_height\_in\_luma\_samples(776)の値は、それぞれ入力ピクチャの幅および高さに等しくてもよい。

40

#### 【0169】

同じ実施形態において、vps\_full\_pic\_width\_in\_luma\_samples(774)およびvps\_full\_pic\_height\_in\_luma\_samples(776)の値は、復号に用いられなくて

50

もよく、合成および表示に用いられてもよい。

#### 【0170】

同じ実施形態において、`vps_sub_picture_dividing_flag`(772)の値が1に等しいとき、構文要素`pic_offset_x`(782)および`pic_offset_y`(784)は、特定の層(複数可)に対応するSPS(780)においてシグナリングされ得る。この場合、SPS(780)でシグナリングされる符号化ピクチャサイズ(`pic_width_in_luma_samples`(786)、`pic_height_in_luma_samples`(788))は、特定の層に対応するサブ領域の幅および高さに等しくてもよい。また、サブ領域の左上隅の位置(`pic_offset_x`(782)、`pic_offset_y`(784))は、SPS(780)でシグナリングされてもよい。

#### 【0171】

同じ実施形態では、サブ領域の左上隅の位置情報(`pic_offset_x`(782)、`pic_offset_y`(784))は復号に使用されなくてもよく、合成および表示に使用されてもよい。

#### 【0172】

同じまたは別の実施形態では、入力ピクチャのすべてまたはサブセットサブ領域のレイアウト情報(サイズおよび位置)、および層間の依存関係情報は、パラメータセットまたはSEIメッセージでシグナリングされ得る。図20は、サブ領域のレイアウト、層間の依存関係、およびサブ領域と1つまたは複数の層との間の関係の情報を示す構文要素の一例を示す図である。この例では、構文要素`num_sub_region`(791)は、現在の符号化されたビデオシーケンス内の(長方形の)サブ領域の数を示す。構文要素`num_layers`(792)は、現在の符号化されたビデオシーケンス内の層数を示す。`num_layers`(792)の値は、`num_sub_region`(791)の値以上であってもよい。任意のサブ領域が単一の層として符号化される場合、`num_layers`(792)の値は、`num_sub_region`(791)の値と等しくてもよい。1つまたは複数のサブ領域が複数の層として符号化されるとき、`num_layers`(792)の値は、`num_sub_region`(791)の値よりも大きくてもよい。構文要素`direct_dependency_flag[i][j]`(793)は、第j層から第i層までの依存関係を示す。構文要素`num_layers_for_region[i]`(794)は、i番目のサブ領域に関連付けられた層の数を示す。構文要素`sub_region_layer_id[i][j]`(795)は、i番目のサブ領域に対応付けられた第j層の`layer_id`を示す。構文要素`sub_region_offset_x[i]`(796)および`sub_region_offset_y[i]`(797)は、それぞれi番目のサブ領域の左上隅の水平および垂直位置を示す。構文要素`sub_region_width[i]`(798)および`sub_region_height[i]`(799)は、それぞれi番目のサブ領域の幅および高さを示す。

#### 【0173】

一実施形態では、プロファイル階層レベル情報の有無にかかわらず出力される複数の層のうちの1つを示すように設定された出力層を指定する1つまたは複数の構文要素は、高レベル構文構造(例えば、VPS、DPS、SPS、PPS、APS、またはSEIメッセージ)でシグナリングされ得る。図21を参照すると、VPSを参照する符号化されたビデオシーケンスにおける出力層セット(`output layer set : OLS`)の数を示す構文要素`num_output_layer_sets`(804)が、VPSでシグナルされ得る。出力層セット毎に、構文要素`output_layer_flag`(810)は出力層の数と同じ回数シグナリングされ得る。

#### 【0174】

同じ実施形態において、1に等しい構文要素`output_layer_flag`(810)は、第i層が出力されることを指定する。0と等しい構文要素`output_layer_flag`(810)は、第i層を出力しないことを指定する。

#### 【0175】

同じまたは別の実施形態では、各出力層セットのプロファイル階層レベル情報を指定する1つまたは複数の構文要素は、高レベル構文構造(例えば、VPS、DPS、SPS、PPS、APS、またはSEIメッセージ)でシグナリングされ得る。さらに図21を参照すると、VPSを参照する符号化されたビデオシーケンス内のOLS毎のプロファイル階層レベル情報の数を示す構文要素`num_profile_tier_level`(806)は、VPS内でシグナリングされ得る。

10

20

30

40

50

出力層セット毎に、プロファイル階層レベル情報の構文要素のセット、またはプロファイル階層レベル情報内のエントリのうちの特定のプロファイル階層レベル情報を示すインデックスが、出力層の数と同じ回数シグナリングされ得る。

#### 【0176】

同じ実施形態において、構文要素profile\_tier\_level\_idx [ i ] [ j ] (812) は、i番目のOLSの第j層に適用されるprofile\_tier\_level( ) (808) 構文構造の、VPSにおけるprofile\_tier\_level( ) (808) 構文構造のリストへのインデックスを指定する。

#### 【0177】

プロファイル、階層、およびレベル（およびその対応する情報）は、ビットストリームに対する制限、したがって、ビットストリームを復号するために必要な能力に対する制限を指定することができる。プロファイル、階層、およびレベル（およびその対応する情報）はまた、個々の復号器実装間の相互運用ポイントを示すために使用され得る。プロファイルは、例えば規格のビットストリーム構文全体のサブセットであってもよい。各プロファイル（およびその対応する情報）は、プロファイルに準拠するすべての復号器によってサポートされ得るアルゴリズム特徴および制限のサブセットを指定することができる。階層およびレベルは、各プロファイル内で指定されてもよく、階層のレベルは、ビットストリーム内の構文要素の値に課される制約の指定されたセットであってもよい。階層の各レベル（およびその対応する情報）は、本開示の構文要素が取り得る値の制限および／または値の算術組み合わせの制限のセットを指定することができる。同じ階層およびレベル定義のセットをすべてのプロファイルで使用することができるが、個々の実装は異なる階層をサポートし、階層内ではサポートされるプロファイル毎に異なるレベルをサポートすることができる。任意の所与のプロファイルについて、階層のレベルは、特定の復号器処理負荷およびメモリ能力に対応することができる。より低い階層に指定されたレベルは、より高い階層に指定されたレベルよりも制約され得る。

10

#### 【0178】

同じまたは別の実施形態では、図22を参照すると、構文要素num\_profile\_tier\_level (806) および／またはnum\_output\_layer\_sets (804) は、最大層数が1より大きい場合 ( $vps\_max\_layers\_minus1 > 0$ ) にシグナリングされ得る。

20

#### 【0179】

同じまたは別の実施形態では、図22を参照すると、第i出力層セットに対する出力層シグナリングのモードを示す構文要素vps\_output\_layers\_mode [ i ] (822) は、VPS内に存在し得る。

30

#### 【0180】

同じ実施形態において、0に等しい構文要素vps\_output\_layers\_mode [ i ] (822) は、第i出力層セットで最上位層のみが出力されることを指定する。1に等しい構文要素vps\_output\_layers\_mode [ i ] (822) は、すべての層が第i出力層セットで出力されることを指定する。2に等しい構文要素vps\_output\_layers\_mode [ i ] (822) は、出力される層が、第i出力層が設定された1に等しいvps\_output\_layer\_flag [ i ] [ j ] を有する層であることを指定する。より多くの値が予約されてもよい。

40

#### 【0181】

同じ実施形態において、構文要素output\_layer\_flag [ i ] [ j ] (810) は、第i出力層セットの構文要素vps\_output\_layers\_mode [ i ] (822) の値に応じてシグナリングされてもされなくてもよい。

#### 【0182】

同じまたは別の実施形態では、図22を参照すると、第i出力層セットについてフラグvps\_ptl\_signal\_flag [ i ] (824) が存在してもよい。vps\_ptl\_signal\_flag [ i ] (824) の値に応じて、第iの出力層セットのプロファイル階層レベル情報はシグナリングされてもされなくてもよい。

#### 【0183】

同じまたは別の実施形態では、図23を参照すると、現在のCVS内のサブピクチャの数m

50

`ax_subpics_minus1`は、高レベル構文構造（例えば、VPS、DPS、SPS、PPS、APS、またはSEIメッセージ）でシグナリングされ得る。

#### 【0184】

同じ実施形態において、図23を参照すると、サブピクチャの数が1より大きい (`max_subpics_minus1 > 0`) 場合、`i`番目のサブピクチャに対するサブピクチャ識別子`sub_pic_id[i]` (821) がシグナリングされ得る。

#### 【0185】

同じまたは別の実施形態では、各出力層セットの各層に属するサブピクチャ識別子を示す1つまたは複数の構文要素は、VPSでシグナリングされ得る。図23を参照すると、識別子`sub_pic_id_layer[i][j][k]` (826) は、第*i*出力層セットの第*j*層に存在する第*k*サブピクチャを示す。復号器は、識別子`sub_pic_id_layer[i][j][k]` (826) の情報を用いて、特定の出力層セットの層毎にどのサブピクチャを復号して出力できるかを認識することができる。10

#### 【0186】

一実施形態では、ピクチャヘッダ (picture header : PH) は、符号化されたピクチャのすべてのスライスに適用される構文要素を含む構文構造である。ピクチャユニット (picture unit : PU) は、指定された分類規則に従って互いに関連付けられ、復号順で連続し、正確に1つの符号化されたピクチャを含むNALユニットのセットである。PUは、ピクチャヘッダ (PH) と、符号化されたピクチャを構成する1または複数のビデオ符号化層 (video coding layer : VCL) NALユニットとを含み得る。20

#### 【0187】

一実施形態では、SPS (RBSP) は、0に等しいTemporalIdを有する少なくとも1つのAUに含まれるか、または外部手段を介して提供されることによって、参照される前に復号処理に利用可能であり得る。

#### 【0188】

一実施形態では、SPS (RBSP) は、SPSを参照する1つまたは複数のPPSを含むCVS内の0に等しいTemporalIdを有する少なくとも1つのAUに含まれることによって、または外部手段を介して提供されることによって、参照される前に復号処理に利用可能であり得る。30

#### 【0189】

一実施形態では、SPS (RBSP) は、SPSを参照する1つまたは複数のPPSを含むCVS内のSPS NALユニットを参照するPPS NALユニットの最も低い`inuh_layer_id`値に等しい`nuh_layer_id`を有する少なくとも1つのPUに含まれることによって、または外部手段を介して提供されることによって、1つまたは複数のPPSによって参照される前に復号処理に利用可能とすることができる。30

#### 【0190】

一実施形態では、SPS (RBSP) は、0に等しいTemporalIdおよびSPS NALユニットを参照するPPS NALユニットの最も低い`inuh_layer_id`値に等しい`nuh_layer_id`を有する少なくとも1つのPUに含まれることによって、または外部手段を介して提供されることによって、1つまたは複数のPPSによって参照される前に復号処理に利用可能とすることができる。40

#### 【0191】

一実施形態では、SPS (RBSP) は、0に等しいTemporalIdと、SPSを参照する1つまたは複数のPPSを含むCVS内のSPS NALユニットを参照するPPS NALユニットの最も低い`inuh_layer_id`値に等しい`nuh_layer_id`とを有する少なくとも1つのPUに含まれることによって、1つまたは複数のPPSによって参照される前に復号処理に利用可能であり得るか、または外部手段を介して提供されるか、または外部手段を介して提供され得る。

#### 【0192】

同じまたは別の実施形態では、識別子`pps_seq_parameter_set_id`は、参照されるSPSの識別子`sps_seq_parameter_set_id`の値を指定する。識別子`pps_seq_para`50

`meter_set_id`の値は、符号化層ビデオシーケンス (coded layer video sequence : CLVS)において符号化ピクチャによって参照されるすべてのPPSにおいて同じであり得る。

#### 【0193】

同じまたは別の実施形態では、CVS内の識別子`sps_seq_parameter_set_id`の特定の値を有するすべてのSPS NALユニットは同じ内容を有することができる。

#### 【0194】

同じまたは別の実施形態では、`nuh_layer_id`値に関係なく、SPS NALユニットは、識別子`sps_seq_parameter_set_id`の同じ値空間を共有することができる。

#### 【0195】

同じまたは別の実施形態では、SPS NALユニットの`nuh_layer_id`値は、SPS NALユニットを参照するPPS NALユニットの最も低い`nuh_layer_id`値に等しくてもよい。 10

#### 【0196】

一実施形態では、`m`に等しい`nuh_layer_id`を有するSPSが`n`に等しい`nuh_layer_id`を有する1つまたは複数のPPSによって参照される場合、`m`に等しい`nuh_layer_id`を有する層は、`n`に等しい`nuh_layer_id`を有する層または`m`に等しい`nuh_layer_id`を有する層の（直接的または間接的な）参照層と同じであり得る。

#### 【0197】

一実施形態では、PPS (RBSP) は、PPS NALユニットの`TemporalId`と等しい`TemporalId`を有する少なくとも1つのAUに含まれることによって、または外部手段を介して提供されることによって、参照される前に復号処理に利用可能であり得る。 20

#### 【0198】

一実施形態では、PPS (RBSP) は、PPSを参照する1つまたは複数のPH（または符号化スライスNALユニット）を含むCVS内のPPS NALユニットの`TemporalId`に等しい`TemporalId`を有する少なくとも1つのAUに含まれることによって、または外部手段を介して提供されることによって、参照される前に復号処理に利用可能とすることができます。

#### 【0199】

一実施形態では、PPS (RBSP) は、PPSを参照する1つまたは複数のPH（または符号化スライスNALユニット）を含む、または外部手段を介して提供される、CVS内のPPS NALユニットを参照する符号化スライスNALユニットの最低`nuh_layer_id`値に等しい`nuh_layer_id`を有する少なくとも1つのPUに含まれることによって、1つまたは複数のPH（または符号化スライスNALユニット）によって参照される前に復号処理に利用可能とすることができます。 30

#### 【0200】

一実施形態では、PPS (RBSP) は、PPSを参照する1つまたは複数のPH（または符号化スライスNALユニット）を含む、または外部手段を介して提供される、CVS内のPPS NALユニットを参照する、PPS NALユニットの`TemporalId`に等しい`TemporalId`および符号化スライスNALユニットの最も低い`nuh_layer_id`値に等しい`nuh_layer_id`を有する少なくとも1つのPUに含まれることによって、1つまたは複数のPH（または符号化スライスNALユニット）によって参照される前に復号処理に利用可能とすることができます。

#### 【0201】

同じまたは別の実施形態では、PH内の識別子`ph_pic_parameter_set_id`は、使用中の参照PPSの識別子`pps_pic_parameter_set_id`の値を指定する。`pps_seq_parameter_set_id`の値は、CLVS内の符号化ピクチャによって参照されるすべてのPPSにおいて同じであり得る。 40

#### 【0202】

同じまたは別の実施形態では、PU内の識別子`pps_pic_parameter_set_id`の特定の値を有するすべてのPPS NALユニットは同じコンテンツを有し得る。

#### 【0203】

同じまたは別の実施形態では、`nuh_layer_id`値に関係なく、PPS NALユニットは、識別子`pps_pic_parameter_set_id`の同じ値空間を共有することができる。 50

**【0204】**

同じまたは別の実施形態では、PPS NALユニットのnuh\_layer\_id値は、PPS NALユニットを参照するNALユニットを参照する符号化スライスNALユニットの最も低いnuh\_layer\_id値に等しくてもよい。

**【0205】**

一実施形態では、mに等しいnuh\_layer\_idを有するPPSが、nに等しいnuh\_layer\_idを有する1つまたは複数の符号化スライスNALユニットによって参照されるとき、mに等しいnuh\_layer\_idを有する層は、nに等しいnuh\_layer\_idを有する層またはmに等しいnuh\_layer\_idを有する層の（直接または間接）参照層と同じであり得る。

**【0206】**

一実施形態では、PPS（RBSP）は、PPS NALユニットのTemporalIdと等しいTemporalIdを有する少なくとも1つのAUに含まれることによって、または外部手段を介して提供されることによって、参照される前に復号処理に利用可能とすることができる。

10

**【0207】**

一実施形態では、PPS（RBSP）は、PPSを参照する1つまたは複数のPH（または符号化スライスNALユニット）を含むCVS内のPPS NALユニットのTemporalIdに等しいTemporalIdを有する少なくとも1つのAUに含まれることによって、または外部手段を介して提供されることによって、参照される前に復号処理に利用可能とすることができる。

**【0208】**

一実施形態では、PPS（RBSP）は、PPSを参照する1つまたは複数のPH（または符号化スライスNALユニット）を含む、または外部手段を介して提供される、CVS内のPPS NALユニットを参照する符号化スライスNALユニットの最低nuh\_layer\_id値に等しいnuh\_layer\_idを有する少なくとも1つのPUに含まれることによって、1つまたは複数のPH（または符号化スライスNALユニット）によって参照される前に復号処理に利用可能とすることができる。

20

**【0209】**

一実施形態では、PPS（RBSP）は、PPSを参照する1つまたは複数のPH（または符号化スライスNALユニット）を含む、または外部手段を介して提供される、CVS内のPPS NALユニットを参照する、PPS NALユニットのTemporalIdに等しいTemporalIdおよび符号化スライスNALユニットの最低nuh\_layer\_id値に等しいnuh\_layer\_idを有する少なくとも1つのPUに含まれることによって、1つまたは複数のPH（または符号化スライスNALユニット）によって参照される前に復号処理に利用可能であり得る。

30

**【0210】**

同じまたは別の実施形態では、PH内の識別子ph\_pic\_parameter\_set\_idは、使用中の参照PPSの識別子pps\_pic\_parameter\_set\_idの値を指定する。識別子pps\_seq\_parameter\_set\_idの値は、CLVS内の符号化ピクチャによって参照されるすべてのPPSにおいて同じであり得る。

**【0211】**

同じまたは別の実施形態では、PU内のpps\_pic\_parameter\_set\_idの特定の値を有するすべてのPPS NALユニットは同じコンテンツを有し得る。

40

**【0212】**

同じまたは別の実施形態では、nuh\_layer\_id値に関係なく、PPS NALユニットは、識別子pps\_pic\_parameter\_set\_idの同じ値空間を共有することができる。

**【0213】**

同じまたは別の実施形態では、PPS NALユニットのnuh\_layer\_id値は、PPS NALユニットを参照するNALユニットを参照する符号化スライスNALユニットの最も低いnuh\_layer\_id値に等しくてもよい。

**【0214】**

一実施形態では、mに等しいnuh\_layer\_idを有するPPSが、nに等しいnuh\_layer\_idを有する1つまたは複数の符号化スライスNALユニットによって参照されるとき、mに

50

等しいnuh\_layer\_idを有する層は、nに等しいnuh\_layer\_idを有する層またはmに等しいnuh\_layer\_idを有する層の（直接または間接）参照層と同じであり得る。

#### 【0215】

出力層は、出力される出力層セットの層であってもよい。出力層セット（OLS）は、指定された層のセットであってもよく、層のセット内の1つまたは複数の層が出力層であるように指定される。出力層セット（OLS）層インデックスは、OLS内の層の、OLS内の層のリストに対するインデックスである。

#### 【0216】

サブ層は、TemporalId変数の特定の値を有するVCL NALユニットおよび関連する非VCL NALユニットを含むサブ層の、時間スケーラブルなビットストリームの時間スケーラブルな層であってもよい。サブ層表現は、特定のサブ層および下位のサブ層のNALユニットを含むビットストリームのサブセットであり得る。

10

#### 【0217】

VPS RBSPは、TemporalIdが0である少なくとも1つのAUに含まれるか、または外部手段を介して提供されることによって、参照される前に復号処理に利用可能であり得る。CVS内のvps\_video\_parameter\_set\_idの特定の値を有するすべてのVPS NALユニットは、同じコンテンツを有することができる。

#### 【0218】

図24～図25を参照して、例示的なVPS RBSPの構文要素を以下に説明する。

20

#### 【0219】

構文要素vps\_video\_parameter\_set\_id(842)は、他の構文要素による参照のためにVPSの識別子を提供する。構文要素vps\_video\_parameter\_set\_id(842)の値は0より大きくてよい。

#### 【0220】

構文要素vps\_max\_layers\_minus1(802)+1は、VPSを参照する各CVS内の最大許容層数を指定する。

#### 【0221】

構文要素vps\_max\_sublayers\_minus1(846)+1は、VPSを参照する各CVS内の層に存在し得る時間サブ層の最大数を指定する。構文要素vps\_max\_sublayers\_minus1(846)の値は、0以上6以下の範囲とすることができます。

30

#### 【0222】

1に等しい構文要素vps\_all\_layers\_same\_num\_sublayers\_flag(848)は、時間サブ層の数が、VPSを参照する各CVS内のすべての層について同じであることを指定する。0に等しい構文要素vps\_all\_layers\_same\_num\_sublayers\_flag(848)は、VPSを参照する各CVS内の層が同じ数の時間サブ層を有しても有しなくてもよいことを指定する。存在しない場合、vps\_all\_layers\_same\_num\_sublayers\_flag(848)の値は1に等しいと推測され得る。

#### 【0223】

1に等しい構文要素vps\_all\_independent\_layers\_flag(850)は、CVS内のすべての層が層間予測を使用せずに独立して符号化されることを指定する。0に等しい構文要素vps\_all\_independent\_layers\_flag(850)は、CVS内の層のうちの1つまたは複数が層間予測を使用できることを指定する。存在しない場合、vps\_all\_independent\_layers\_flag(850)の値は1に等しいと推測され得る。

40

#### 【0224】

構文要素vps\_layer\_id[i](852)は、第i層のnuh\_layer\_id値を指定する。mおよびnの任意の2つの負でない整数値について、mがn未満であるとき、vps\_layer\_id[m]の値はvps\_layer\_id[n]未満であり得る。

#### 【0225】

1に等しい構文要素vps\_independent\_layer\_flag[i](854)は、インデックスiを有する層が層間予測を使用しないことを指定する。0に等しい構文要素vps\_independ

50

`ent_layer_flag[i]` (854) は、インデックス*i*を有する層が層間予測を使用することができるなどを指定し、0以上*i*-1以下の範囲内の*j*の構文要素`vps_direct_ref_layer_flag[i][j]`がVPSに存在する。存在しない場合、構文要素`vps_independent_layer_flag[i]` (854) の値は1に等しいと推論され得る。

#### 【0226】

0に等しい構文要素`vps_direct_ref_layer_flag[i][j]` (856) は、インデックス*j*を有する層がインデックス*i*を有する層に対する直接参照層ではないことを指定する。1に等しい構文要素`vps_direct_ref_layer_flag[i][j]` (856) は、インデックス*j*を有する層がインデックス*i*を有する層のための直接参照層であることを指定する。0以上、`vps_max_layers_minus1`以下の範囲の*i*および*j*について、構文要素`vps_direct_ref_layer_flag[i][j]` (856) が存在しない場合、その構文要素は0と等しいと推論され得る。構文要素`vps_independent_layer_flag[i]` (854) が0に等しい場合、構文要素`vps_direct_ref_layer_flag[i][j]` (856) の値が1に等しくなるように、0以上*i*-1以下の範囲内に少なくとも1つの*j*の値があつてもよい。

#### 【0227】

変数`NumDirectRefLayers[i]`、`DirectRefLayerIdx[i][d]`、`NumRefLayers[i]`、`RefLayerIdx[i][r]`、および`LayerUsedAsRefLayerFlag[j]`は、以下のように導出することができる。

```
for (i = 0 ; i <= vps_max_layers_minus1 ; i++) {
    for (j = 0 ; j <= vps_max_layers_minus1 ; j++) {
        dependencyFlag[i][j] = vps_direct_ref_layer_flag[i][j]
        for (k = 0 ; k < i ; k++)
            if (vps_direct_ref_layer_flag[i][k] && dependencyFlag[k][j])
                dependencyFlag[i][j] = 1
    }
    LayerUsedAsRefLayerFlag[i] = 0
}
for (i = 0 ; i <= vps_max_layers_minus1 ; i++) {
    for (j = 0 , d = 0 , r = 0 ; j <= vps_max_layers_minus1 ; j++) { (37)
        if (vps_direct_ref_layer_flag[i][j]) {
            DirectRefLayerIdx[i][d++] = j
            LayerUsedAsRefLayerFlag[j] = 1
        }
        if (dependencyFlag[i][j])
            RefLayerIdx[i][r++] = j
    }
    NumDirectRefLayers[i] = d
    NumRefLayers[i] = r
} (40)
```

#### 【0228】

`vps_layer_id[i]` (852) に等しい`nuh_layer_id`を有する層の層インデックスを指定する変数`GeneralLayerIdx[i]`は、以下のように導出することができる。

```
for (i = 0 ; i <= vps_max_layers_minus1 ; i++) (38)
    GeneralLayerIdx[vps_layer_id[i]] = i
```

#### 【0229】

両方とも0から`vps_max_layers_minus1` (846) までの範囲内の*i*および*j*の任意の2つの異なる値について、`dependencyFlag[i][j]`が1に等しいとき、第*i*層に適用される`chroma_format_idc`および`bit_depth_minus8`の値は、第*j*層に適用される`chroma_format_idc`および`bit_depth_minus8`の値にそれぞれ等しくなり得ることがビッ

10

20

30

40

50

トストリーム適合性の要件であり得る。

#### 【0 2 3 0】

1に等しい構文要素max\_tid\_ref\_present\_flag [ i ] ( 858 ) は、構文要素max\_tid\_il\_ref\_pics\_plus1 [ i ] ( 860 ) が存在することを指定する。0に等しい構文要素max\_tid\_ref\_present\_flag [ i ] ( 858 ) は、構文要素max\_tid\_il\_ref\_pics\_plus1 [ i ] ( 860 ) が存在しないことを指定する。

#### 【0 2 3 1】

0に等しい構文要素max\_tid\_il\_ref\_pics\_plus1 [ i ] ( 860 ) は、第i層の非IRAPピクチャによって層間予測が使用されないことを指定する。0より大きい構文要素max\_tid\_il\_ref\_pics\_plus1 [ i ] ( 860 ) は、第i層のピクチャを復号するために、max\_tid\_il\_ref\_pics\_plus1 [ i ] - 1より大きいTemporalIdを有するピクチャが層間参照ピクチャ (ILRP) として使用されないことを指定する。存在しない場合、構文要素max\_tid\_il\_ref\_pics\_plus1 [ i ] ( 860 ) の値は7に等しいと推測され得る。

#### 【0 2 3 2】

1に等しい構文要素each\_layer\_is\_an\_ols\_flag ( 862 ) は、各OLSが1つの層のみを含み、VPSを参照するCVS内の各層自体がOLSであり、含まれる単一の層が唯一の出力層であることを指定する。0に等しい構文要素each\_layer\_is\_an\_ols\_flag ( 862 ) は、OLSが複数の層を含み得ることを指定する。構文要素vps\_max\_layers\_minus1が0に等しい場合、構文要素each\_layer\_is\_an\_ols\_flag ( 862 ) の値は1に等しいと推測され得る。そうでない場合、構文要素vps\_all\_independent\_layers\_flag ( 854 ) が0に等しいとき、各構文要素each\_layer\_is\_an\_ols\_flag ( 862 ) の値は0に等しいと推測され得る。

#### 【0 2 3 3】

0に等しい構文要素ols\_mode\_idc ( 864 ) は、VPSによって指定されたOLSの総数がvps\_max\_layers\_minus1 + 1に等しいことを指定し、i番目のOLSは0以上i以下の層インデックスを有する層を含み、各OLSに対して、OLSの最上位層のみが出力される。

#### 【0 2 3 4】

1に等しい構文要素ols\_mode\_idc ( 864 ) は、VPSによって指定されたOLSの総数がvps\_max\_layers\_minus1 + 1に等しいことを指定し、i番目のOLSは0以上i以下の層インデックスを有する層を含み、各OLSについて、OLS内のすべての層が出力される。

#### 【0 2 3 5】

2に等しい構文要素ols\_mode\_idc ( 864 ) は、VPSによって指定されたOLSの総数が明示的にシグナリングされ、各OLSについて出力層が明示的にシグナリングされ、他の層がOLSの出力層の直接的または間接的な参照層である層であることを指定する。

#### 【0 2 3 6】

構文要素ols\_mode\_idc ( 864 ) の値は、0以上2以下の範囲であってもよい。構文要素ols\_mode\_idc ( 864 ) の値3は、ITU-T | ISO / IECによる将来の使用のために予約することができる。

#### 【0 2 3 7】

構文要素vps\_all\_independent\_layers\_flag ( 850 ) が1に等しく、各\_layer\_is\_an\_ols\_flag ( 862 ) が0に等しい場合、構文要素ols\_mode\_idc ( 864 ) の値は2に等しいと推測され得る。

#### 【0 2 3 8】

構文要素num\_output\_layer\_sets\_minus1 ( 866 ) + 1は、構文要素ols\_mode\_idc ( 864 ) が2に等しいときにVPSによって指定されるOLSの総数を指定する。

#### 【0 2 3 9】

VPSによって指定されたOLSの総数を指定する変数TotalNumOIssは、以下のように導出することができる。

```
if ( vps_max_layers_minus1 == 0 )
    TotalNumOIss = 1
```

10

20

30

40

50

```

else if ( each_layer_is_an_ols_flag | | ols_mode_idc == 0 | | ols_mode
_idc == 1 )
    TotalNumOIss = vps_max_layers_minus1 + 1
else if ( ols_mode_idc == 2 )
    TotalNumOIss = num_output_layer_sets_minus1 + 1

```

## 【 0 2 4 0 】

1に等しい構文要素ols\_output\_layer\_flag [ i ] [ j ] ( 868 ) は、ols\_mode\_idc ( 864 ) が2に等しい場合、nuh\_layer\_idがvps\_layer\_id [ j ] に等しい層がi番目のOLSの出力層であることを指定する。0に等しい構文要素ols\_output\_layer\_flag [ i ] [ j ] ( 868 ) は、構文要素ols\_mode\_idc ( 864 ) が2に等しい場合、nuh\_layer\_idがvps\_layer\_id [ j ] に等しい層はi番目のOLSの出力層ではないことを指定する。

## 【 0 2 4 1 】

i番目のOLSにおける出力層の数を指定する変数NumOutputLayersInOls [ i ] と、i番目のOLSにおける第j層におけるサブ層の数を指定する変数NumSubLayersInLayerInOLS [ i ] [ j ] と、i番目のOLSにおける第j出力層のnuh\_layer\_id値を指定する変数OutputLayerIdInOls [ i ] [ j ] と、第k層が少なくとも1つのOLSにおける出力層として使用されるか否かを指定する変数LayerUsedAsOutputLayerFlag [ k ] とは、以下のように導出されてもよい。

```

NumOutputLayersInOls [ 0 ] = 1
OutputLayerIdInOls [ 0 ] [ 0 ] = vps_layer_id [ 0 ]
NumSubLayersInLayerInOLS [ 0 ] [ 0 ] = vps_max_sub_layers_minus1 + 1
LayerUsedAsOutputLayerFlag [ 0 ] = 1
for ( i = 1 , i < vps_max_layers_minus1 ; i ++ ) {
    if ( each_layer_is_an_ols_flag | | ols_mode_idc < 2 )
        LayerUsedAsOutputLayerFlag [ i ] = 1
    else /* ( !each_layer_is_an_ols_flag & & ols_mode_idc == 2 ) */ 
        LayerUsedAsOutputLayerFlag [ i ] = 0
}
for ( i = 1 ; i < TotalNumOIss ; i ++ )
    if ( each_layer_is_an_ols_flag | | ols_mode_idc == 0 ) {
        NumOutputLayersInOls [ i ] = 1
        OutputLayerIdInOls [ i ] [ 0 ] = vps_layer_id [ i ]
        for ( j = 0 ; j < i & & ( ols_mode_idc == 0 ) ; j ++ )
            NumSubLayersInLayerInOLS [ i ] [ j ] = max_tid_il_ref_pics_plus1
[i]
        NumSubLayersInLayerInOLS [ i ] [ i ] = vps_max_sub_layers_minus1
+ 1
    } else if ( ols_mode_idc == 1 ) {
        NumOutputLayersInOls [ i ] = i + 1
        for ( j = 0 ; j < NumOutputLayersInOls [ i ] ; j ++ ) {
            OutputLayerIdInOls [ i ] [ j ] = vps_layer_id [ j ]
            NumSubLayersInLayerInOLS [ i ] [ j ] = vps_max_sub_layers_minus
1 + 1
        }
    } else if ( ols_mode_idc == 2 ) {
        for ( j = 0 ; j < vps_max_layers_minus1 ; j ++ ) {
            layerIncludedInOlsFlag [ i ] [ j ] = 0
            NumSubLayersInLayerInOLS [ i ] [ j ] = 0
        }
        for ( k = 0 , j = 0 ; k < = vps_max_layers_minus1 ; k ++ ) ( 40 )
    }
}

```

10

20

30

40

50

```

    if ( ols_output_layer_flag [ i ] [ k ] ) {
        layerIncludedInOlsFlag [ i ] [ k ] = 1
        LayerUsedAsOutputLayerFlag [ k ] = 1
        OutputLayerIdx [ i ] [ j ] = k
        OutputLayerIdInOls [ i ] [ j + + ] = vps_layer_id [ k ]
        NumSubLayersInLayerInOLS [ i ] [ j ] = vps_max_sub_layers_minus
s1 + 1
    }
    NumOutputLayersInOls [ i ] = j
    for ( j = 0 ; j < NumOutputLayersInOls [ i ] ; j + + ) {
        idx = OutputLayerIdx [ i ] [ j ]
        for ( k = 0 ; k < NumRefLayers [ idx ] ; k + + ) {
            layerIncludedInOlsFlag [ i ] [ RefLayerIdx [ idx ] [ k ] ] = 1
            if ( NumSubLayersInLayerInOLS [ i ] [ RefLayerIdx [ idx ] [ k ] ] <
                max_tid_il_ref_pics_plus1 [ OutputLayerIdInOls [ i ] [ j ] ] )
                NumSubLayersInLayerInOLS [ i ] [ RefLayerIdx [ idx ] [ k ] ] =
                    max_tid_il_ref_pics_plus1 [ OutputLayerIdInOls [ i ] [ j ] ]
        }
    }
}
【 0 2 4 2 】
10

```

0以上、vps\_max\_layers\_minus1以下の範囲のiの各値について、LayerUsedAsRefLayerFlag [ i ] およびLayerUsedAsOutputLayerFlag [ i ] の値は、いずれも0に等しくなくてもよい。言い換えると、少なくとも1つのOLSの出力層でも他の層の直接参照層でもない層は存在しなくてもよい。

#### 【 0 2 4 3 】

各OLSについて、出力層である少なくとも1つの層が存在し得る。すなわち、0以上TotalNumOls - 1以下の範囲の任意のiについて、NumOutputLayersInOls [ i ] の値を1以上としてもよい。

#### 【 0 2 4 4 】

30  
i番目のOLSの層数を指定する変数NumLayersInOls [ i ] 、およびi番目のOLSの第j層のnuh\_layer\_id値を指定する変数LayerIdInOls [ i ] [ j ] は、以下のように導出され得る。

```

NumLayersInOls [ 0 ] = 1
LayerIdInOls [ 0 ] [ 0 ] = vps_layer_id [ 0 ]
for ( i = 1 ; i < TotalNumOls ; i + + ) {
    if ( each_layer_is_an_ols_flag ) {
        NumLayersInOls [ i ] = 1
        LayerIdInOls [ i ] [ 0 ] = vps_layer_id [ i ]
    } else if ( ols_mode_idc == 0 || ols_mode_idc == 1 ) {
        NumLayersInOls [ i ] = i + 1
        for ( j = 0 ; j < NumLayersInOls [ i ] ; j + + )
            LayerIdInOls [ i ] [ j ] = vps_layer_id [ j ]
    } else if ( ols_mode_idc == 2 ) {
        for ( k = 0 , j = 0 ; k <= vps_max_layers_minus1 ; k + + )
            if ( layerIncludedInOlsFlag [ i ] [ k ] )
                LayerIdInOls [ i ] [ j + + ] = vps_layer_id [ k ]
        NumLayersInOls [ i ] = j
    }
}
40
50

```

**【 0 2 4 5 】**

nuh\_layer\_idがLayerIdInOls [ i ] [ j ] に等しい層のOLS層インデックスを指定する変数OlsLayerIdx [ i ] [ j ] は、以下のように導出される。

```
for ( i = 0 ; i < TotalNumOIss ; i + + )
    for j = 0 ; j < NumLayersInOls [ i ] ; j + +
        OlsLayerIdx [ i ] [ LayerIdInOls [ i ] [ j ] ] = j
```

**【 0 2 4 6 】**

各OLSにおける最下層は、独立した層であってもよい。すなわち、0以上TotalNumOIss - 1以下の範囲のi毎に、vps\_independent\_layer\_flag [ GeneralLayerIdx [ LayerIdInOls [ i ] [ 0 ] ] ] の値が1になるようにしてもよい。各層は、VPSによって指定される少なくとも1つのOLSに含まれてもよい。言い換えれば、0以上、vps\_max\_layers\_minus1以下の範囲内のkに対して、vps\_layer\_id [ k ] のうちの1つに等しいnuh\_layer\_id、nuhLayerIdの特定の値を有する各層について、iおよびjの値の少なくとも1つの対が存在してもよく、ここで、iは0以上、TotalNumOIss - 1以下の範囲内にあり、jはNumLayersInOls [ i ] - 1以下の範囲内にあり、その結果、LayerIdInOls [ i ] [ j ] の値はnuhLayerIdに等しい。

**【 0 2 4 7 】**

一実施形態では、復号処理は、構文要素PictureOutputFlagを設定するために現在のピクチャ（例えば、syntax要素CurrPic）に対して以下のように動作することができる。

**【 0 2 4 8 】**

以下の条件のうちの1つが真である場合、PictureOutputFlagは0に等しく設定される。

( 1 ) 現在のピクチャはRASLピクチャであり、関連付けられたIRAPピクチャのNoOutputBeforeRecoveryFlagは1に等しい。

( 2 ) gdr\_enabled\_flagが1に等しく、現在のピクチャが、1に等しいNoOutputBeforeRecoveryFlagを有するGDRピクチャである。

( 3 ) gdr\_enabled\_flagが1に等しく、現在のピクチャが、NoOutputBeforeRecoveryFlagが1に等しいGDRピクチャと関連付けられ、現在のピクチャのPicOrderCntValが、関連付けられたGDRピクチャのRpPicOrderCntValよりも小さい。

( 4 ) sps\_video\_parameter\_set\_idが0より大きく、ols\_mode\_idcが0に等しく、現在のAUが以下の条件：( a ) PicAが1に等しいPictureOutputFlagを有する、( b ) PicAが現在のピクチャよりも大きいnuh\_layer\_id nuhLidを有する、( c ) PicAがOLSの出力層に属する（すなわち、OutputLayerIdInOls [ TargetOlsIdx ] [ 0 ] はnuhLidに等しい）のすべてを満たすピクチャ（例えば、構文要素picA）を含む。

( 5 ) sps\_video\_parameter\_set\_idは0より大きく、ols\_mode\_idcは2に等しく、ols\_output\_layer\_flag [ TargetOlsIdx ] [ GeneralLayerIdx [ nuh\_layer\_id ] ] は0に等しい。

**【 0 2 4 9 】**

上記の条件のいずれも真でない場合、構文要素PictureOutputFlagは、構文要素pic\_output\_flagと等しく設定され得る。

**【 0 2 5 0 】**

現在のピクチャのすべてのスライスが復号された後、現在の復号されたピクチャは「短期参照に使用」としてマークされてもよく、RefPicList [ 0 ] またはRefPicList [ 1 ] 内の各ILRPエントリは「短期参照に使用」としてマークされてもよい。

**【 0 2 5 1 】**

同一または他の実施形態において、各層が出力層セットである場合、構文要素ols\_mode\_idc ( 864 ) の値に関わらず、構文要素PictureOutputFlagはpic\_output\_flagと等しく設定される。

**【 0 2 5 2 】**

同一または別の実施形態において、構文要素PictureOutputFlagは、sps\_video\_parameter\_set\_idが0より大きい場合に0に等しく設定され、各\_layer\_is\_an\_ols\_f

10

20

30

40

50

`lag ( 862 )` が0に等しく、`ols_mode_idc ( 864 )` が0に等しく、現在のAUは、以下の条件：PicAが1に等しいPictureOutputFlagを有する、PicAが現在のピクチャのそれよりも大きい`nuh_layer_id` `nuhLid`を有する、およびPicAがOLS（すなわち、`OutputLayerIdInOls [ TargetOlsIdx ] [ 0 ]` は`nuhLid`に等しい）の出力層に属する、のすべてを満たすピクチャpicAを含む。

#### 【 0 2 5 3 】

同一または別の実施形態において、構文要素PictureOutputFlagは、`sps_video_parameter_set_id`が0より大きい場合に0に等しく設定され、`each_layer_is_an_ols_flag`が0に等しく、`ols_mode_idc`が2に等しく、`ols_output_layer_flag [ TargetOlsIdx ] [ GeneralLayerIdx [ nuh_layer_id ] ]` が0に等しい。

10

#### 【 0 2 5 4 】

イントラ・ランダム・アクセス・ポイント（IRAP）ピクチャは、クローズドグループのピクチャ構造をサポートする瞬時復号リフレッシュ（IDR）ピクチャであり得る、または、オープングループのピクチャ構造をサポートするクリーン・ランダム・アクセス（CRA）ピクチャであり得る、ランダムアクセスのための符号化されたピクチャであり得る。段階的復号リフレッシュ（GDR）ピクチャは、ピクチャの部分リフレッシュを伴う緩やかなランダムアクセスのためのピクチャであってもよい。

#### 【 0 2 5 5 】

本開示の実施形態は、IRAPピクチャまたはGDRピクチャを示す構文要素を含み得る。例えば、図26を参照すると、ピクチャヘッダ（1）が提供され得る。ピクチャヘッダ（1）において、フラグ`ph_gdr_or_irap_pic_flag`（2）がシグナリングされてもよい。フラグは、ピクチャヘッダ（1）に関連付けられている現在のPU内にIRAPピクチャまたはGDRピクチャが存在することを示す。

20

#### 【 0 2 5 6 】

同じまたは別の実施形態では、図26に示すように、`ph_gdr_or_irap_pic_flag`（2）が1に等しい場合にのみ、フラグ`ph_no_output_of_prior_pics_flag`（3）を条件付きでシグナリングすることができる。`ph_no_output_of_prior_pics_flag`（3）の値は、DPBからのピクチャの出力および除去処理に使用され得る。フラグの値は、ビットストリーム内の最初のAUではないCVSS AU内のピクチャの復号後のDPB内の以前に復号されたピクチャの出力に影響を及ぼし得る。

30

#### 【 0 2 5 7 】

1に等しいフラグ`ph_gdr_or_irap_pic_flag`が、現在のピクチャがGDRまたはIRAPピクチャであることを指定するように、`ph_gdr_or_irap_pic_flag`のセマンティクスによって指定される制約は「一方向」であり得るので、IRAPピクチャは0に等しい`ph_gdr_or_irap_pic_flag`を有し得るという潜在的な問題が存在する。0に等しいフラグ`ph_gdr_or_irap_pic_flag`は、現在のピクチャがGDRピクチャではなく、IRAPピクチャであってもよいことを指定する。IRAPピクチャの`ph_gdr_or_irap_pic_flag`の値が0に等しい場合、`ph_no_output_of_prior_pics_flag`の値は、シグナリングまたは推論規則なしで、DPB動作のために使用され得る。

40

#### 【 0 2 5 8 】

潜在的な問題に対処するために、一実施形態において、`ph_gdr_or_irap_pic_flag`（2）のセマンティクスの制約を「双方向」として指定することができ、そのため、現在のピクチャがIRAPピクチャであるときに`ph_no_output_of_prior_pics_flag`（3）をシグナリングする必要があり得る。1に等しいフラグ`ph_gdr_or_irap_pic_flag`（2）は、現在のピクチャがGDRまたはIRAPピクチャであることを指定する。0に等しいフラグ`ph_gdr_or_irap_pic_flag`（2）は、現在のピクチャがGDRピクチャでもIRAPピクチャでもないことを指定する。

#### 【 0 2 5 9 】

同じまたは別の実施形態では、`ph_no_output_of_prior_pics_flag`（3）値の推論規則は、存在しない場合、以下のように指定され得る：フラグ`ph_no_output_of_p`

50

`prior_pics_flag(3)` は、ビットストリーム内の最初のAUではないCVSS AU内のピクチャの復号後のDPB内の以前に復号されたピクチャの出力に影響を及ぼす。存在する場合、`ph_no_output_of_prior_pics_flag(3)` の値は、AU内のすべてのピクチャについて同じでなければならないことがビットストリーム適合性の要件であり得る。

#### 【0260】

AU内のピクチャのピクチャヘッダ(1)内に`ph_no_output_of_prior_pics_flag(3)` が存在する場合、AUの`ph_no_output_of_prior_pics_flag(3)` 値は、AU内のピクチャの`ph_no_output_of_prior_pics_flag(3)` 値である。存在しない場合、`ph_no_output_of_prior_pics_flag(3)` の値は0に等しいと推測され得る。

#### 【0261】

図27を参照すると、AUデリミタ(10)を使用して、AUの開始、AUがIRAPまたはGDR AUであるか否か、およびAUデリミタNALユニットを含むAU内の符号化ピクチャに存在するスライスの型式を示すことができる。ビットストリームがただ1つの層を含むとき、AUデリミタ(10)に関連付けられた標準的な復号処理は存在しない場合がある。

#### 【0262】

AUデリミタ(10)において、`aud_irap_or_gdr_au_flag(12)` は、IRAPまたはGDR AUの存在を示すことができ、図27に示すようにシグナリングされることがある。1であるフラグ`aud_irap_or_gdr_au_flag(12)` は、AU delimiterを含むAUがIRAPまたはGDR AUであることを指定してもよい。また、フラグ`aud_irap_or_gdr_au_flag(12)` が0であることは、AU delimiter(10)を含むAUがIRAPでもGDR AUでもないことを示してもよい。

#### 【0263】

同じまたは別の実施形態において、IRAPまたはGDR AUのフラグ`aud_irap_or_gdr_au_flag(12)` は、ビットストリームが複数の層を有する場合に存在してもよく、`sps_video_parameter_set_id` は0より大きい。ビデオ符号化技術または規格は、複数層ビットストリームのためのAUデリミタの存在を要求し得る。

#### 【0264】

同じまたは別の実施形態では、図26～図27を参照すると、`aud_irap_or_gdr_au_flag(12)` が存在し、`aud_irap_or_gdr_au_flag(12)` の値が1に等しいとき、`ph_gdr_or_irap_pic_flag(2)` の値は1に等しいことが必要とされ得る。AU delimiter(10)における`aud_irap_or_gdr_au_flag(12)` が1である場合、各PUは、GDRまたはIRAPピクチャを有し得るからである。

#### 【0265】

同一または別の実施形態において、`pps_mixed_nalu_types_in_pic_flag` が1であるとき、`ph_no_output_of_prior_pics_flag` の値は存在しなくてもよく、`pps_mixed_nalu_types_in_pic_flag` が存在すると判定(例えば復号器によって)されたとき、`ph_no_output_of_prior_pics_flag(3)` の値は無視されてもよい。

#### 【0266】

同じまたは別の実施形態では、`ph_no_output_of_prior_pics_flag(3)` は、ビットストリーム内の最初のAUではないCVSS AU内のピクチャの復号後のDPB内の以前に復号されたピクチャの出力に影響を及ぼし得る。存在しない場合、`ph_no_output_of_prior_pics_flag(3)` の値は1に等しいと推測され得る。

#### 【0267】

同じまたは別の実施形態では、`ph_no_output_of_prior_pics_flag(3)` の値が推論規則なしで使用されるという問題を解決するために、`ph_no_output_of_prior_pics_flag(3)` が存在せず、`ph_gdr_or_irap_pic_flag(2)` が1に等しい場合、図28に示すように、ピクチャヘッダ(1)において`ph_gdr_or_irap_pic_flag(2)` を`ph_irap_pic_flag(6)` で置き換えることができる。1に等しいフラグ`ph_irap_pic_flag(6)` は、現在のピクチャがIRAPピクチャであることを指定することができる。0に等しいフラグ`ph_irap_pic_flag(6)` は、現在のピクチャがIRAPピクチャではなく

10

20

30

40

50

いことを指定し得る。

#### 【 0 2 6 8 】

同じまたは別の実施形態では、`ph_no_output_of_prior_pics_flag(3)` の値が推論規則なしで使用されるという問題を解決するために、`ph_no_output_of_prior_pics_flag(3)` が存在せず、`ph_gdr_or_irap_pic_flag(2)` が1に等しいとき、図29に示すように、ピクチャヘッダ(1)内の`ph_no_output_of_prior_pics_flag(3)`は、スライスヘッダ(20)内の`sh_no_output_of_prior_pics_flag(23)`に置き換えられてもよい。

#### 【 0 2 6 9 】

同じ実施形態において、`sh_no_output_of_prior_pics_flag(23)`は、現在のVCL NALのNALユニット型式がIDR\_W\_RADL、IDR\_N\_LP、またはCRA\_NUTに等しい場合にのみ、スライスヘッダ(20)内に条件付きで存在してもよい。IDR\_W\_RADLは、ビットストリーム内に存在する関連するRASLピクチャを有さないが、ビットストリーム内に関連するRADLピクチャを有し得るIDRピクチャの符号化スライスセグメントを含むNALユニット型式である。IDR\_N\_LPは、ビットストリーム内に存在する関連する先頭ピクチャを有しないIDRピクチャの符号化スライスセグメントを含むNALユニット型式であり得る。CRA\_NUTは、CRAピクチャの符号化スライスセグメントを含むNALユニット型式である。

#### 【 0 2 7 0 】

同じまたは別の実施形態では、`sh_no_output_of_prior_pics_flag(23)`は、ビットストリーム内の最初のAUではないCVSS AU内のピクチャの復号後のDPB内の以前に復号されたピクチャの出力に影響を及ぼし得る。

#### 【 0 2 7 1 】

同じまたは別の実施形態では、存在する場合、`sh_no_output_of_prior_pics_flag(23)`の値はAU内のすべてのピクチャについて同じであるべきであることがビットストリーム適合性の要件であり得る。AU内のピクチャのスライスヘッダ(20)に`sh_no_output_of_prior_pics_flag(23)`が存在する場合、AUの`sh_no_output_of_prior_pics_flag(23)`値は、AU内のピクチャの`sh_no_output_of_prior_pics_flag(23)`値であってもよい。

#### 【 0 2 7 2 】

同一または別の実施形態において、ピクチャ・パラメータ・セット内の`pps_mixed_nalu_types_in_pic_flag`が1に等しいとき、`sh_no_output_of_prior_pics_flag(23)`の値は存在しなくてもよい。存在する場合、`sh_no_output_of_prior_pics_flag(23)`の値は無視され得る。

#### 【 0 2 7 3 】

同じまたは別の実施形態では、図30に示すように、`aud_irap_au_flag(16)`がAUデリミタ(10)に存在してもよい。1であるフラグ`aud_irap_au_flag(16)`は、AUデリミタ(10)を含むAUがIRAP AUであることを特定してもよい。フラグ`aud_irap_au_flag(16)`が0であることは、AUデリミタ(10)を含むAUがIRAP AUでないことを示してもよい。

#### 【 0 2 7 4 】

また、本実施の形態において、`aud_irap_au_flag(16)`が存在するとき、ピクチャヘッダ(1)の`ph_irap_pic_flag(6)`の値は、AUデリミタ(10)の`aud_irap_au_flag(16)`と等しくてもよい。1に等しいフラグ`ph_irap_pic_flag(7)`は、PH(1)と関連付けられたピクチャがIRAPピクチャであることを指定し得る。0に等しいフラグ`ph_irap_pic_flag(6)`は、PH(1)と関連付けられたピクチャがIRAPピクチャではないことを指定し得る。

#### 【 0 2 7 5 】

同じまたは別の実施形態では、図30に示すように、`aud_gdr_au_flag(17)`がAUデリミタ(10)に存在してもよい。1であるフラグ`aud_gdr_au_flag(17)`は、AUデ

10

20

30

40

50

リミタ(10)を含むAUがGDR AUであることを示してもよい。フラグaud\_irap\_au\_flag(17)が0であることは、AUデリミタ(10)を含むAUがGDR AUでないことを示してもよい。

#### 【0276】

また、本実施の形態において、AU delimiter(10)におけるaud\_gdr\_au\_flag(17)が存在する場合、ピクチャヘッダ(1)におけるph\_gdr\_pic\_flag(7)の値は、AU delimiter(10)におけるaud\_gdr\_au\_flag(17)と等しくてもよい。1に等しいフラグph\_gdr\_pic\_flag(7)は、PH(1)と関連付けられたピクチャがGDRピクチャであることを指定し得る。0に等しいフラグph\_gdr\_pic\_flag(7)は、PH(1)と関連付けられたピクチャがGDRピクチャではないことを指定し得る。

10

#### 【0277】

段階的復号リフレッシュ(GDR)は、以下の定義によって指定され得る。

GDR AU : VPSで特定される層毎のPUが存在し、存在する各PU内の符号化ピクチャがGDRピクチャであるAU。

GDR PU : 符号化ピクチャがGDRピクチャであるPU。

GDRピクチャ : 各VCL NALユニットがGDR\_NUTと等しいnal\_unit\_typeを持つピクチャ。

GDRサブピクチャ : 各VCL NALユニットがGDR\_NUTに等しいnal\_unit\_typeを有するサブピクチャ。

GDR\_NUT : GDRピクチャの符号化タイルグループを含むNALユニット型式。

20

#### 【0278】

実施形態によれば、復号順のビットストリーム内の第1のピクチャは、IRAPまたはGDRピクチャとすることができます。IRAPまたはGDRピクチャに関連付けられた後のピクチャもまた、復号順でIRAPまたはGDRピクチャに続き得る。関連付けられたIRAPピクチャの出力順に続き、かつ、関連付けられたIRAPピクチャの復号順に先行するピクチャは、許容されない場合がある。

#### 【0279】

一実施形態では、図31を参照すると、GDRを示す構文要素は、ph\_gdr\_pic\_flag(7)などのPH(1)でシグナリングされる。存在しない場合、ph\_gdr\_pic\_flag(7)の値は0に等しいと推測することができる。sps\_gdr\_enabled\_flagが0に等しいとき、ph\_gdr\_pic\_flag(7)の値は0に等しいと推測することができる。構文要素ph\_recovery\_poc\_cnt(32)は、復号されたピクチャの復元ポイントを出力順に指定することができる。

30

#### 【0280】

現在のピクチャがGDRピクチャである場合、変数recoveryPointPocValは、以下のように導出されてもよい。

```
recoveryPointPocVal = PicOrderCntVal + ph_recovery_poc_cnt
```

#### 【0281】

同じまたは別の実施形態において、図31に示されるように、recoveryPointPocValを導出するために使用されるPicOrderCntValがph\_pic\_order\_cnt\_lsb(32)およびph\_poc\_msb\_cycle\_val(36)の値から導出されるので、ph\_recovery\_poc\_cnt(34)は、ピクチャ順序カウント(POC)構文要素(例えば、ph\_pic\_order\_cnt\_lsb(32)およびph\_poc\_msb\_cycle\_val(36))をシグナリングした後にシグナリングされる。

40

#### 【0282】

現在のピクチャがGDRピクチャであり、recoveryPointPocValに等しいPicOrderCntValを有するCLVS内に復号順で現在のGDRピクチャに続くピクチャpicAがある場合、そのピクチャpicAを復元ポイントピクチャと呼ぶことができる。そうでない場合、CLVSにおけるrecoveryPointPocValよりも大きいPicOrderCntValを有する出力順の最初のピクチャを、復元ポイントピクチャと呼ぶことができる。復元ポイントピクチャは、復号順

50

で現在のGDRピクチャの前になくてもよい。現在のGDRピクチャに関連付けられ、かつrecoveryPointPocVal未満のPicOrderCntValを有するピクチャは、GDRピクチャの復元ピクチャと呼ばれる場合がある。ph\_recovery\_poc\_cnt(34)の値は、0以上、Max PicOrderCntLsb - 1以下であってもよい。

#### 【0283】

同じまたは別の実施形態では、復元ポイントピクチャは、復号順および出力順の両方で現在のGDRピクチャの前になくてもよい。

#### 【0284】

同じ実施形態または別の実施形態では、復元ピクチャは、復号順および出力順の両方で現在のGDRピクチャの前になくてもよい。

10

#### 【0285】

同じ実施形態または別の実施形態において、復元ピクチャは、復号順および出力順の両方において、関連付けられた復元ポイントピクチャに先行し得る。

#### 【0286】

同じまたは別の実施形態において、現在のピクチャがGDRピクチャまたはGDRピクチャの復元ピクチャであり、現在のピクチャが「リフレッシュ領域」（すなわち、存在する場合、前のIRAPピクチャから復号順に復号処理を開始するのと比較して、GDRピクチャから復号処理を開始するときに、復号されたサンプル値の完全一致を有する領域）と「ダーティ領域」（すなわち、存在する場合、前のIRAPピクチャから復号順で復号処理を開始するのと比較して、GDRピクチャから復号処理を開始するときに、復号されたサンプル値の完全一致を有しない可能性がある領域）との間の非CTU整列境界を含むとき、「ダーティ領域」を回避して「リフレッシュ領域」の復号されたサンプル値に影響を及ぼすために、クロマスケーリング(LMCS)を用いたルママッピングのクロマ残差スケーリングを現在のピクチャにおいて無効にする必要があり得る。

20

#### 【0287】

同一または別の実施形態において、層AのrecoveryPointPocValの値は、層Aの参照層のrecoveryPointPocVal以上であってもよい。

#### 【0288】

同じまたは別の実施形態では、layerIdがmに等しい層AのrecoveryPointPocValの値は、layerIdがnに等しい別の層BのrecoveryPointPocVal以上であってもよく、mはnよりも大きい。

30

#### 【0289】

同じまたは別の実施形態では、layerIdがmに等しい層AのrecoveryPointPocValの値は、layerIdがnに等しい別の層BのrecoveryPointPocVal以上であってもよく、mはnよりも大きく、層AおよびBは同じ出力層セットに属する。

#### 【0290】

同じまたは別の実施形態では、層AのrecoveryPointPocValの値は、層Aの参照層のrecoveryPointPocValに等しくてもよい。

#### 【0291】

同じまたは別の実施形態では、layerIdがmに等しい層AのrecoveryPointPocValの値は、layerIdがnに等しい別の層BのrecoveryPointPocValに等しくてもよく、mはnよりも大きい。

40

#### 【0292】

同じまたは別の実施形態では、layerIdがmに等しい層AのrecoveryPointPocValの値は、layerIdがnに等しい別の層BのrecoveryPointPocValに等しくてもよく、mはnよりも大きく、層AおよびBは同じ出力層セットに属する。

#### 【0293】

同じまたは別の実施形態では、pps\_mixed\_nalu\_types\_in\_pic\_flagが1に等しいとき、以下が適用され得る。

(1) ピクチャは、少なくとも2つのサブピクチャを有するものとする。

50

(2) ピクチャのVCL NALユニットは、2つ以上の異なるnal\_unit\_type値を有するものとする。

(3) GDR\_NUTに等しいnal\_unit\_typeを有するピクチャのVCL NALユニットは存在しないものとする。

(4) ピクチャは、GDRピクチャに関連付けられた復元または復元ピクチャではないものとする。

(5) ピクチャの少なくとも1つのサブピクチャのVCL NALユニットが、IDR\_W\_RA\_DL、IDR\_N\_LP、またはCRA\_NUTに等しいnal\_unit\_typeの特定の値を有するとき、ピクチャ内の他のサブピクチャのVCL NALユニットはすべて、TRAIL\_NUTに等しいnal\_unit\_typeを有するものとする。 10

#### 【0294】

TRAIL\_NUTは、VCL非STSAトレイリングピクチャの符号化タイルグループを含むNALユニット型式である。

#### 【0295】

ピクチャの各スライスの復号処理の開始時に、参照ピクチャリスト0(RefPicList[0])および参照ピクチャリスト1(RefPicList[1])の導出のために、参照ピクチャリスト構築のための復号処理が呼び出されてもよい。1つまたは複数の参照ピクチャリストを構築した後、参照ピクチャマーキングのための復号処理が呼び出されてもよく、参照ピクチャは、「参照に未使用」、「短期参照に使用」、または「長期参照に使用」としてマークされてもよい。 20

#### 【0296】

図32を参照すると、実施形態によれば、復号処理(40)は復号器によって実行される。復号処理(40)では、1つまたは複数の参照ピクチャリスト(RPL)を構築することができる(42)。1つまたは複数の参照ピクチャリストが構築されるとき、1つまたは複数のピクチャは、ランダムアクセスまたは意図しないピクチャ損失のために利用できない可能性がある。復号器は、RPL内の参照ピクチャがDPBで利用可能であるか否かを判定し得る(44)。参照ピクチャが利用できないと判定された場合、利用できない参照ピクチャは「参照ピクチャなし」としてマークされ得る。復号器のクラッシュまたは意図しない挙動を回避するために、利用できない参照ピクチャは、ピクセルおよびパラメータの初期設定値を用いて即座に生成され得る(46)。利用できない参照ピクチャを生成した後(および/または、参照ピクチャが利用可能であると判定された後)、復号器は、参照ピクチャリスト内の、生成されたピクチャを含むすべての参照ピクチャの検証を確認することができる(48)。 30

#### 【0297】

同じまたは別の実施形態において、1つまたは複数の参照ピクチャリストは、SPS、PH、および/またはSH内のRPL構文要素を構文解析することによって構築される(42)。構築するステップ(42)の後、CRAピクチャに関連付けられたランダム・アクセス・スキップされた先頭ピクチャ(RASL)は、復号器またはシステムによって廃棄され得るか、または、CRAピクチャでランダムアクセスが発生した場合に正しく復号され得ないので、RPLリスト内の1つまたは複数の参照ピクチャは利用可能ではない場合があり得る。利用できない参照ピクチャは、ピクセルおよびパラメータの初期設定値を用いて生成することができる(46)。 40

#### 【0298】

同じまたは別の実施形態では、現在のピクチャが、1に等しいsps\_idr\_rpl\_present\_flag、または1に等しいpps\_rpl\_info\_in\_ph\_flagを有するIDRピクチャ、1に等しいNoOutputBeforeRecoveryFlagを有するCRAピクチャ、または1に等しいNoOutputBeforeRecoveryFlagを有するGDRピクチャである場合、利用できない参照ピクチャ(46)を生成するための以下の復号処理のうちの少なくとも1つが呼び出され、これはピクチャの第1のスライスに対してのみ呼び出される必要があり得る。

#### 【0299】

### A. 利用できない参照ピクチャを生成するための一般的な復号処理

この処理は、現在のピクチャが、1に等しいsps\_idr\_rpl\_present\_flagまたは1に等しいpps\_rpl\_info\_in\_ph\_flagを有するIDRピクチャ、1に等しいNoOutputBeforeRecoveryFlagを有するCRAピクチャ、または1に等しいNoOutputBeforeRecoveryFlagを有するGDRピクチャである場合、符号化されたピクチャ毎に1回呼び出され得る。この処理が呼び出されると、以下が適用され得る：0以上1以下の範囲のi、および0以上num\_ref\_entries[i] [RplsIdx[i]] - 1以下の範囲のj、すなわち「参照ピクチャなし」に等しい各RefPicList[i][j]について、「1枚の利用できないピクチャの生成」で以下に説明するようにピクチャが生成され、以下が適用される。

(1) 生成されるピクチャのnuh\_layer\_idの値は、現在のピクチャのnuh\_layer\_idと等しく設定される。10

(2) st\_ref\_pic\_flag[i] [RplsIdx[i]] [j] が1に等しく、inter\_layer\_ref\_pic\_flag[i] [RplsIdx[i]] [j] が0に等しい場合、生成されるピクチャのPicOrderCntValの値はRefPicPocList[i][j]に等しく設定され、生成されるピクチャは「短期参照に使用」とマークされる。

(3) そうでない場合、st\_ref\_pic\_flag[i] [RplsIdx[i]] [j] が0に等しく、inter\_layer\_ref\_pic\_flag[i] [RplsIdx[i]] [j] が0に等しい場合、生成されたピクチャに対するPicOrderCntValの値はRefPicLtPocList[i][j]に等しく設定され、生成されたピクチャに対するph\_pic\_order\_cnt\_lsbの値は(RefPicLtPocList[i][j] & (MaxPicOrderCntLsb - 1))に等しいと推測され、生成されたピクチャは「長期参照に使用」とマークされる。20

(4) 生成される参照ピクチャのPictureOutputFlagの値は0に設定される。

(5) 生成された参照ピクチャには、RefPicList[i][j]が設定される。

(6) 生成されたピクチャのTemporalIdの値は、現在のピクチャのTemporalIdと等しく設定される。

(7) 生成されたピクチャのph\_non\_ref\_pic\_flagの値は0に等しく設定される。

(8) 生成されたピクチャのph\_pic\_parameter\_set\_idの値は、現在のピクチャのph\_pic\_parameter\_set\_idと等しく設定される。

#### 【0 3 0 0】

1に等しいフラグph\_non\_ref\_pic\_flagは、PHに関連付けられたピクチャが決して参照ピクチャとして使用されることを指定し得る。0に等しいフラグph\_non\_ref\_pic\_flagは、PHに関連付けられたピクチャが参照ピクチャとして用いられても用いられなくてよいことを指定する。30

#### 【0 3 0 1】

### B. 1枚の利用できないピクチャの生成

この処理が呼び出されると、以下のように利用できないピクチャが生成される。

(1) ピクチャのサンプル配列SL内の各要素の値は、 $1 \ll (\text{BitDepth} - 1)$ に等しく設定される。

(2) sps\_chroma\_format\_idcが0でないとき、そのピクチャに対するサンプル配列Scb, Scrの各要素の値は、 $1 \ll (\text{BitDepth} - 1)$ と等しく設定される。40

(3) 予測モードCuPredMode[0][x][y]は、xが0以上、pps\_pic\_width\_in\_luma\_samples - 1以下であり、yが0以上、pps\_pic\_height\_in\_luma\_samples - 1以下である場合、MODE\_INTRAと等しく設定される。

#### 【0 3 0 2】

同じ実施形態または別の実施形態では、RPLリスト内の利用できない参照ピクチャを生成した後、RPLリスト内のすべてまたはアクティブな参照ピクチャのビットストリーム適合性確認が、例えば復号器によって呼び出され得る。例えば、復号器は、以下の制約がビットストリーム適合性に適用されることを確認することができる。

(1) 0または1に等しい各iについて、num\_ref\_entries[i] [RplsIdx[i]] は、NumRefIdxActive[i] 未満であってはならない。50

(2) RefPicList [ 0 ] またはRefPicList [ 1 ] の各アクティブエントリによって参照されるピクチャは、DPBに存在し、現在のピクチャのTemporalId以下であるものとする。

(3) RefPicList [ 0 ] またはRefPicList [ 1 ] 内の各エントリによって参照されるピクチャは、現在のピクチャではなく、0に等しいph\_non\_ref\_pic\_flagを有するものとする。

(4) ピクチャのスライスのRefPicList [ 0 ] またはRefPicList [ 1 ] 内の短期参照ピクチャ (STRP) エントリと、同じピクチャの同じスライスまたは異なるスライスのRefPicList [ 0 ] またはRefPicList [ 1 ] 内の長期参照ピクチャ (LTRP) エントリとは、同じピクチャを参照してはならない。

(5) 現在のピクチャのPicOrderCntValとエントリによって参照されるピクチャのPicOrderCntValとの間の差が $2^{24}$ 以上であるLTRPエントリがRefPicList [ 0 ] 又はRefPicList [ 1 ] 内に存在してはならない。 10

(6) setOfRefPicsを、現在のピクチャと同じnuh\_layer\_idを有するRefPicList [ 0 ] 内のすべてのエントリ、および現在のピクチャと同じnuh\_layer\_idを有するRefPicList [ 1 ] 内のすべてのエントリによって参照される固有のピクチャのセットとする。setOfRefPicsのピクチャの数は、MaxDpbSize - 1以下でなければならず、ここで、MaxDpbSizeは、であり、setOfRefPicsは、ピクチャのすべてのスライスについて同じでなければならない。

(7) 現在のスライスがSTSA\_NUTに等しいnal\_unit\_typeを有するとき、現在のピクチャのものに等しいTemporalIdおよび現在のピクチャのものに等しいnuh\_layer\_idを有するRefPicList [ 0 ] またはRefPicList [ 1 ] 内にアクティブエントリは存在しないものとする。 20

(8) 現在のピクチャが、復号順に、現在のピクチャと等しいTemporalIdおよび現在のピクチャと等しいnuh\_layer\_idを有する段階的な時間サブ層アクセス (STSA) ピクチャに続くピクチャである場合、復号順にSTSAピクチャに先行し、現在のピクチャと等しいTemporalIdを有し、RefPicList [ 0 ] またはRefPicList [ 1 ] にアクティブエントリとして含まれる現在のピクチャと等しいnuh\_layer\_idを有するピクチャは存在しないものとする。

(9) 特定の値tIdに等しいTemporalId、特定の値layerIdに等しいnuh\_layer\_id、および特定の値subpicIdxに等しいサブピクチャインデックスを有する現在のサブピクチャが、復号順に、tIdに等しいTemporalId、layerIdに等しいnuh\_layer\_id、およびsubpicIdxに等しいサブピクチャインデックスを有するSTSAサブピクチャに続くサブピクチャである場合、RefPicList [ 0 ] またはRefPicList [ 1 ] にアクティブエントリとして含まれる復号順にSTSAサブピクチャを含むピクチャに先行する、tIdに等しいTemporalId、およびlayerIdに等しいnuh\_layer\_idを有するピクチャは存在しないものとする。 30

(10) 特定の値layerIdに等しいnuh\_layer\_idを有する現在のピクチャがIRAPピクチャであるとき、出力順または復号順において、(存在する場合) 復号順においてlayerIdに等しいnuh\_layer\_idを有する先行するIRAPピクチャに先行するRefPicList [ 0 ] またはRefPicList [ 1 ] 内のエントリによって参照されるピクチャは存在しないものとする。

(11) 特定の値layerIdに等しいnuh\_layer\_idおよび、特定の値subpicIdxに等しいサブピクチャインデックスを有する現在のサブピクチャがIRAPサブピクチャであるとき、出力順または復号順において、(存在する場合) 復号順においてlayerIdに等しいnuh\_layer\_idおよびsubpicIdxに等しいサブピクチャインデックスを有するIRAPサブピクチャを含む任意の先行するピクチャに先行するRefPicList [ 0 ] またはRefPicList [ 1 ] 内のエントリによって参照されるピクチャはないものとする。 40

(12) 現在のピクチャが、NoOutputBeforeRecoveryFlagが1であるCRAピクチャに関連付けられたRASLピクチャでない場合、現在のピクチャに関連付けられたCRAピクチャのための利用できない参照ピクチャを生成するための復号処理によって生成されたRefPicList [ 0 ] またはRefPicList [ 1 ] 内のアクティブエントリによって参照されるピクチャは存在しないものとする。 50

(13) 現在のサブピクチャが、NoOutputBeforeRecoveryFlagが1であるCRAピクチャ内のCRAサブピクチャに関連付けられたRASLサブピクチャでない場合、現在のサブピクチャに関連付けられたCRAサブピクチャを含むCRAピクチャ用の利用できない参照ピクチャを生成するための復号処理によって生成されたRefPicList [ 0 ] またはRefPicList [ 1 ] 内のアクティブエントリによって参照されるピクチャはないものとする。

(14) 特定の値layerIdに等しいnuh\_layer\_idを有する現在のピクチャが以下のいずれでもない場合、現在のピクチャに関連付けられたIRAPピクチャまたはGDRピクチャのための利用できない参照ピクチャを生成するための復号処理によって生成されたRefPicList [ 0 ] またはRefPicList [ 1 ] 内のエントリによって参照されるピクチャは存在しないはずである。 10

(a) sps\_idr\_rpl\_present\_flagが1またはpps\_rpl\_info\_in\_ph\_flagが1であるIDRピクチャ。

(b) NoOutputBeforeRecoveryFlagが1であるCRAピクチャ。

(c) NoOutputBeforeRecoveryFlagが1であるCRAピクチャに関連付けられたピクチャであって、同じCRAピクチャに関連付けられた先頭のピクチャよりも復号順が早いピクチャ。

(d) NoOutputBeforeRecoveryFlagが1であるCRAピクチャに関連付けられた先頭ピクチャ

(e) NoOutputBeforeRecoveryFlagが1であるGDRピクチャ。

(f) NoOutputBeforeRecoveryFlagが1であり、nuh\_layer\_idがlayerIdであるGDRピクチャの復元ピクチャ。 20

(15) nuh\_layer\_idが特定の値layerIdに等しく、サブピクチャインデックスが特定の値subpicIdxに等しい、現在のサブピクチャが以下のいずれでもない場合、現在のサブピクチャに関連付けられたIRAPまたはGDRサブピクチャを含むIRAPまたはGDRピクチャ用の利用できない参照ピクチャを生成するための復号処理によって生成されたRefPicList [ 0 ] またはRefPicList [ 1 ] 内のエントリによって参照されるピクチャはないはずである。

(a) sps\_idr\_rpl\_present\_flagが1またはpps\_rpl\_info\_in\_ph\_flagが1であるIDRピクチャ内のIDRサブピクチャ。

(b) NoOutputBeforeRecoveryFlagが1であるCRAピクチャ内のCRAサブピクチャ。

(c) NoOutputBeforeRecoveryFlagが1であるCRAピクチャ内のCRAサブピクチャに関連付けられたサブピクチャであって、同じCRAピクチャに関連付けられた先頭ピクチャよりも復号順が早いサブピクチャ。

(d) NoOutputBeforeRecoveryFlagが1であるCRAピクチャ内のCRAサブピクチャに関連付けられた先頭サブピクチャ。

(e) NoOutputBeforeRecoveryFlagが1であるGDRピクチャ内のGDRサブピクチャ。

(f) NoOutputBeforeRecoveryFlagが1であり、nuh\_layer\_idがlayerIdであるGDRピクチャの復元ピクチャ内のサブピクチャ。 30

(16) 現在のピクチャが、復号順および出力順の両方でnuh\_layer\_idの値が同じIRAPピクチャに続くとき、そのIRAPピクチャの出力順または復号順に先行するRefPicList [ 0 ] またはRefPicList [ 1 ] 内のアクティブエントリによって参照されるピクチャはないものとする。 40

(17) 現在のサブピクチャが、復号順と出力順の両方で同じ値のnuh\_layer\_idおよび同じ値のサブピクチャインデックスを有するIRAPサブピクチャに続くとき、そのIRAPサブピクチャを含むピクチャに出力順または復号順で先行するRefPicList [ 0 ] またはRefPicList [ 1 ] 内のアクティブエントリによって参照されるピクチャはないものとする。

(18) 現在のピクチャが、nuh\_layer\_idの値が同じであるIRAPピクチャと、もしあれば、そのIRAPピクチャに復号順および出力順の両方で関連付けられている先頭ピクチャとを辿るとき、そのIRAPピクチャに出力順または復号順で先行するRefPicList [ 0 ] またはRefPicList [ 1 ] 内のエントリによって参照されるピクチャは存在しないものとする。 50

(19) 現在のサブピクチャが、同じ値のnuh\_layer\_idおよび同じ値のサブピクチャインデックスを有するIRAPサブピクチャと、もしあれば、復号順と出力順の両方でそのIRAPサブピクチャに関連付けられた先頭サブピクチャとに続くとき、出力順または復号順でそのIRAPサブピクチャを含むピクチャに先行するRefPicList [0] またはRefPicList [1] 内のエントリによって参照されるピクチャはないものとする。

(20) 現在のピクチャがランダムアクセス復号可能先頭 (RADL) ピクチャである場合、以下のいずれかであるRefPicList [0] またはRefPicList [1] にアクティブエントリは存在しないものとする。

(a) pps\_mixed\_nalu\_types\_in\_pic\_flag のRASLピクチャは0である。これは、RADLピクチャのRPLのアクティブエントリが、1に等しいpps\_mixed\_nalu\_types\_in\_pic\_flag を有するRASLピクチャを参照し得ることを意味する。しかしながら、RADLピクチャは、RASLサブピクチャを参照するRADLサブピクチャを許可しない次の制約によって課されるように、参照されるRASLピクチャ内のRADLサブピクチャのみを参照するため、参照されるRASLピクチャ内のRADLサブピクチャが正しく復号されることから、関連するCRAピクチャから復号が開始されるとき、そのようなRADLピクチャは依然として正しく復号され得る。

(b) 関連付けられたIRAPピクチャの復号順に先行するピクチャ。

(21) 特定の値layerIdに等しいnuh\_layer\_id および特定の値subpicIdxに等しいサブピクチャインデックスを有する現在のサブピクチャがRADLサブピクチャであるとき、以下のいずれかであるRefPicList [0] またはRefPicList [1] 内にアクティブエントリはないものとする。

(a) nuh\_layer\_id がlayerId に等しいピクチャは、subpicIdx に等しいサブピクチャインデックスを有するRASLサブピクチャを含む。

(b) 関連付けられたIRAPサブピクチャを含むピクチャの復号順に先行するピクチャ。

(22) 各ILRPエントリによって参照されるピクチャには、現在のピクチャのスライスのRefPicList [0] またはRefPicList [1] に存在する場合、以下の制約が適用される。

(a) ピクチャは、現在のピクチャと同じAUにあるものとする。

(b) ピクチャは、DPBに存在するものとする。

(c) ピクチャは、現在のピクチャのnuh\_layer\_id よりも小さいnuh\_layer\_id refPicLayerId を有するものとする。

(d) 以下の制約のいずれかが適用される：ピクチャはIRAPピクチャであるものとする；または、ピクチャはMax (0, vps\_max\_tid\_il\_ref\_pics\_plus1 [ currLayerIdx ] [ refLayerIdx ] - 1) 以下のTemporalIdを有するものとし、currLayerIdx およびrefLayerIdx は、それぞれGeneralLayerIdx [ nuh\_layer\_id ] およびGeneralLayerIdx [ refpicLayerId ] に等しい。

(23) スライスのRefPicList [0] またはRefPicList [1] 内に存在する場合、各ILRPエントリはアクティブエントリであるものとする。

(24) vps\_independent\_layer\_flag [ GeneralLayerIdx [ nuh\_layer\_id ] ] が0に等しく、sps\_num\_subpics\_minus1 が0より大きい場合、以下の2つの条件のいずれか（両方ではない）が真であるものとする。

(a) RefPicList [0] またはRefPicList [1] の各アクティブエントリが参照するピクチャは、現在のピクチャと同一のサブピクチャ・レイアウトを有する（すなわち、そのピクチャおよび現在のピクチャによって参照されるSPSは、0からsps\_num\_subpics\_minus1までの範囲内のjの各値に対して、それぞれ、同じ値のsps\_num\_subpics\_minus1と、同じ値のsps\_subpic\_ctu\_top\_left\_x [ j ]、sps\_subpic\_ctu\_top\_left\_y [ j ]、sps\_subpic\_width\_minus1 [ j ]、およびsps\_subpic\_height\_minus1 [ j ]とを有する）。

(b) RefPicList [0] またはRefPicList [1] の各アクティブエントリが参照するピクチャは、sps\_num\_subpics\_minus1の値が0であるILRPである。

### C. 参照ピクチャマーキングの復号処理

同じ実施形態または別の実施形態では、参照ピクチャがマークされると、以下の復号処理が呼び出され得る。

#### 【0304】

この処理は、スライスヘッダの復号およびスライスの参照ピクチャリスト構築のための復号処理の後であるが、スライスデータの復号の前に、ピクチャ毎に1回呼び出されてもよい。この処理により、DPB内の1つまたは複数の参照ピクチャが「参照に未使用」または「長期参照に使用」としてマークされ得る。

#### 【0305】

DPBにおける復号されたピクチャは、「参照に未使用」、「短期参照に使用」、または「長期参照に使用」としてマークされることが可能であるが、復号処理の動作中の任意の所与の瞬間にこれら3つのうちの1つのみがマークされることが可能である。これらのマークリングのうちの1つをピクチャに割り当てるとは、適用可能な場合、これらのマークリングのうちの別のものを暗黙的に除去することができる。ピクチャが「参照に使用」としてマークされていると言及される場合、これは、「短期参照に使用」または「長期参照に使用」としてマークされているピクチャを総称する（両方ではない）。

#### 【0306】

STRPおよびILRPは、それらのnuh\_layer\_idおよびPicOrderCntVal値によって識別され得る。LTRPは、それらのnuh\_layer\_id値およびそれらのPicOrderCntVal値のLog 2 (MaxLtPicOrderCntLsb) LSBによって識別され得る。

#### 【0307】

現在のピクチャがCLVSSピクチャである場合、現在のピクチャと同じnuh\_layer\_idを有する（もしあれば）DPB内のすべての現在の参照ピクチャが「参照に未使用」としてマークされ得る。

#### 【0308】

そうでなければ、以下が適用され得る。

(1) RefPicList [ 0 ] またはRefPicList [ 1 ] 内の各LTRPエントリについて、ピクチャが現在のピクチャと同じnuh\_layer\_idを持つLTRPである場合、そのピクチャは「長期参照に使用」とマークされる。

(2) RefPicList [ 0 ] やRefPicList [ 1 ] のいずれのエントリも参照していないDPBのカレントピクチャと同じnuh\_layer\_idの参照ピクチャは「参照に未使用」とされる。

(3) RefPicList [ 0 ] またはRefPicList [ 1 ] 内の各ILRPエントリについて、ピクチャは「長期参照に使用」としてマークされる。

#### 【0309】

同一または別の実施形態において、RefPicList [ 0 ] またはRefPicList [ 1 ] 内の各LTRPエントリについて、現在のピクチャと同じnuh\_layer\_idを持つピクチャが「短期参照に使用」とマークされた場合、そのピクチャは「長期参照に使用」とマークされる。

#### 【0310】

同じ実施形態または別の実施形態において、1つまたは複数のRPLリストが構築されるとき、以下が適用され得る。

各RefPicList [ i ] [ j ] について、iは0以上1以下の範囲にあり、jは0以上num\_ref\_entries [ i ] [ RplIdx [ i ] ] - 1以下の範囲にあり、すなわち「参照ピクチャなし」に等しいピクチャが生成される。

#### 【0311】

同じ実施形態または別の実施形態において、1つまたは複数のRPLリストが構築されるとき、以下が適用され得る。

各RefPicList [ i ] [ j ] について、iは0以上1以下の範囲にあり、jは0以上NumRefIdxActive [ i ] - 1以下の範囲にあり、すなわち「参照ピクチャなし」に等しいピクチャが生成される。

#### 【0312】

10

20

30

40

50

参照ピクチャリスト構築のための復号処理では、DPBで欠落している参照ピクチャを「参照ピクチャなし」と等しく設定することができる。「参照ピクチャなし」に等しい利用できないピクチャは、適合性確認目的で利用できない参照ピクチャを生成するための復号処理を介して生成される。問題は、以下のように、同じ層内で利用できない参照ピクチャだけでなく、参照層内で利用できない層間参照ピクチャも「参照ピクチャなし」に等しく設定されることである。

```

if( ! inter_layer_ref_pic_flag [ i ] [ RplIdx [ i ] ] [ j ] ) {
    if( st_ref_pic_flag [ i ] [ RplIdx [ i ] ] [ j ] ) {
        RefPicPocList [ i ] [ j ] = pocBase + DeltaPocValSt [ i ] [ RplIdx [ i ] ] [ j ]
    }
    if( there is a reference picture picA in the DPB with the same nuh_layer_id as the current picture
        and PicOrderCntVal equal to RefPicPocList [ i ] [ j ] )
        RefPicList [ i ] [ j ] = picA
    else
        RefPicList [ i ] [ j ] = "no reference picture"
        pocBase = RefPicPocList [ i ] [ j ]
    } else {
        if( ! delta_poc_msb_cycle_present_flag [ i ] [ k ] ) {
            if( there is a reference picA in the DPB with the same nuh_layer_id as the current picture and
                PicOrderCntVal & ( MaxPicOrderCntLsb - 1 ) equal to PocLsbLt [ i ] [ k ] )
                RefPicList [ i ] [ j ] = picA
            else
                RefPicList [ i ] [ j ] = "no reference picture"
                RefPicLtPocList [ i ] [ j ] = PocLsbLt [ i ] [ k ]
        } else {
            if( there is a reference picA in the DPB with the same nuh_layer_id as the current picture and
                PicOrderCntVal equal to FullPocLt [ i ] [ k ] )
                RefPicList [ i ] [ j ] = picA
            else
                RefPicList [ i ] [ j ] = "no reference picture"
                RefPicLtPocList [ i ] [ j ] = FullPocLt [ i ] [ k ]
        }
        k ++
    }
} else {
    layerIdx = DirectRefLayerIdx [ GeneralLayerIdx [ nuh_layer_id ] ] [ ilrp_idx [ i ] [ RplIdx ] [ j ] ]
    refPicLayerId = vps_layer_id [ layerIdx ]
    if( there is a reference picture picA in the DPB with nuh_layer_id equal to refPicLayerId and
        the same PicOrderCntVal as the current picture )
        RefPicList [ i ] [ j ] = picA
    else
        RefPicList [ i ] [ j ] = "no reference picture"
}
【 0 3 1 3 】
しかしながら、利用できない参照ピクチャを生成するための復号処理では、「参照ピク

```

10

20

30

40

50

「チャなし」に等しい全ての利用できないピクチャが同じ層内の参照ピクチャとして扱われてもよく、その結果、利用できない層間参照ピクチャのnuh\_layer\_idおよびPicOrderCntVal値は、正確な値に設定される。また、利用できない層間参照ピクチャは、「長期参照に使用」として正しくマークされていない。これらの誤った値は、後続のピクチャの復号処理およびビットストリーム適合性確認においてエラーをもたらす。したがって、実施形態によれば、「参照ピクチャなし」に等しい利用できない層間参照ピクチャは、以下のように正しく生成されるべきである。

#### 【0314】

D. 利用できない参照ピクチャを生成するための改善された復号処理

この処理は、現在のピクチャが、1に等しいsps\_idr\_rpl\_present\_flagまたは1に等しいpps\_rpl\_info\_in\_ph\_flagを有するIDRピクチャ、1に等しいNoOutputBeforeRecoveryFlagを有するCRAピクチャ、または1に等しいNoOutputBeforeRecoveryFlagを有するGDRピクチャであるとき、符号化されたピクチャ毎に1回呼び出される。10

#### 【0315】

この処理が呼び出されると、各RefPicList[i][j]について、0以上1以下の範囲のi、および0以上num\_ref\_entries[i][RplsIdx[i]] - 1以下の範囲のj、すなわち「参照ピクチャなし」に等しいiを用いて、本開示で前述したようにピクチャを生成することができ、以下が適用され得る。

(1) inter\_layer\_ref\_pic\_flag[i][RplsIdx[i]][j]が0に等しい場合、生成されるピクチャのnuh\_layer\_idの値は現在のピクチャのnuh\_layer\_idに等しく設定される。20

(2) そうでない場合 (inter\_layer\_ref\_pic\_flag[i][RplsIdx[i]][j]が1に等しい)、生成されたピクチャのnuh\_layer\_idの値はvps\_layer\_id[DirectRefLayerIdx[GeneralLayerIdx[nuh\_layer\_id]][i]rp\_idx[i][RplsIdx[j]]]に等しく設定される。

(3) st\_ref\_pic\_flag[i][RplsIdx[i]][j]が1に等しく、inter\_layer\_ref\_pic\_flag[i][RplsIdx[i]][j]が0に等しい場合、生成されるピクチャのPicOrderCntValの値はRefPicPocList[i][j]に等しく設定され、生成されるピクチャは「短期参照に使用」とマークされる。

(4) st\_ref\_pic\_flag[i][RplsIdx[i]][j]が0に等しく、inter\_layer\_ref\_pic\_flag[i][RplsIdx[i]][j]が0に等しい場合、生成されるピクチャのPicOrderCntValの値はRefPicLtPocList[i][j]に等しく設定され、生成されるピクチャのph\_pic\_order\_cnt\_lsbの値は(RefPicLtPocList[i][j] & (MaxPicOrderCntLsb - 1))に等しいと推測され、生成されるピクチャは「長期参照に使用」とマークされる。30

(5) あるいは、inter\_layer\_ref\_pic\_flag[i][RplsIdx[i]][j]が1に等しい場合、生成されたピクチャのPicOrderCntValの値は現在のピクチャのPicOrderCntValに等しく設定され、生成されたピクチャのph\_pic\_order\_cnt\_lsbの値は現在のピクチャのph\_pic\_order\_cnt\_lsbに等しいと推測され、生成されたピクチャは「長期参照に使用」としてマークされる。

(6) 生成される参照ピクチャのPictureOutputFlagの値は0に設定される。

(7) 生成された参照ピクチャには、RefPicList[i][j]が設定される。

(8) 生成されたピクチャのTemporalIdの値は、現在のピクチャのTemporalIdと等しく設定される。

(9) 生成されたピクチャのph\_non\_ref\_pic\_flagの値は0に等しく設定される。

(10) 生成されたピクチャのph\_pic\_parameter\_set\_idの値は、現在のピクチャのph\_pic\_parameter\_set\_idと等しく設定される。

#### 【0316】

上述した参照ピクチャマーキングのための復号処理では、マーキング処理を明確にするために、以下の文が修正される：RefPicList[0]またはRefPicList[1]内の各LTPR工

10

20

30

40

50

ントリについて、ピクチャが現在のピクチャと同じnuh\_layer\_idを有するSTRPである場合、そのピクチャは「長期参照に使用」としてマークされる。

#### 【0317】

第1の選択肢として、元の文は以下のように修正される：RefPicList[0]またはRefPicList[1]内の各LTPRエントリについて、ピクチャが現在のピクチャと同じnuh\_layer\_idで「短期参照に使用」として現在マークされている場合、ピクチャは「長期参照に使用」としてマークされる。

#### 【0318】

第2の選択肢として、元の文は以下のように修正される：RefPicList[0]またはRefPicList[1]内の各LTPRエントリについて、ピクチャが現在のピクチャと同じnuh\_layer\_idを有するLTPRである場合、ピクチャは「長期参照に使用」としてマークされる。10

#### 【0319】

STRPは「長期参照に使用」と表記されているため、原文の意味がやや分かりにくい。文の元の意図が、RefPicList[0]またはRefPicList[1]内の現在のピクチャのLTPRピクチャを「長期参照に使用」としてマークすることである場合、LTPRが現在「短期参照に使用」としてマークされているとき、選択肢1を使用することができる。そうでなければ、選択肢2を使用することができる。

#### 【0320】

##### E. 参照ピクチャリスト構築のための改善された復号処理

HEVCでは、参照ピクチャリストは、利用できない参照ピクチャをRPSで生成した後に構築される。VVC (current draft) では、参照ピクチャリストは、RPSで利用できない参照ピクチャを生成する前に構築される。以前に指定されたように利用できない参照ピクチャを生成する前にビットストリーム適合性確認が実行される場合、ビットストリームは指定された制約に従うことができない。順序を明確にするために、復号処理は、ビットストリーム適合性について以下の制約を有するように以下のように修正することができる。20

(1) 0または1に等しい各iについて、num\_ref\_entries[i][RpIsIdx[i]]は、NumRefIdxActive[i]未満であってはならない。

(2) RefPicList[0]またはRefPicList[1]の各アクティブエントリによって参照されるピクチャは、DPBに存在し、現在のピクチャのTemporalId以下であるものとする。

(3) RefPicList[0]またはRefPicList[1]内の各エントリによって参照されるピクチャは、現在のピクチャではなく、0に等しいph\_non\_ref\_pic\_flagを有するものとする。30

(4) ピクチャのスライスのRefPicList[0]またはRefPicList[1]内のSTRPエントリ、および同じピクチャの同じスライスまたは異なるスライスのRefPicList[0]またはRefPicList[1]内のLTPRエントリは、同じピクチャを参照しないものとする。

(5) 現在のピクチャのPicOrderCntValとエントリによって参照されるピクチャのPicOrderCntValとの間の差が $2^{24}$ 以上であるLTPRエントリがRefPicList[0]又はRefPicList[1]内に存在してはならない。

(6) setOfRefPicsを、現在のピクチャと同じnuh\_layer\_idを有するRefPicList[0]内のすべてのエントリ、および現在のピクチャと同じnuh\_layer\_idを有するRefPicList[1]内のすべてのエントリによって参照される固有のピクチャのセットとする。setOfRefPicsのピクチャの数は、MaxDpbSize - 1以下でなければならず、MaxDpbSizeおよびsetOfRefPicsは、ピクチャのすべてのスライスについて同じでなければならない。40

(7) 現在のスライスがSTSA\_NUTに等しいnal\_unit\_typeを有するとき、現在のピクチャのものに等しいTemporalIdおよび現在のピクチャのものに等しいnuh\_layer\_idを有するRefPicList[0]またはRefPicList[1]内にアクティブエントリは存在しないものとする。

(8) 現在のピクチャが、復号順に、現在のピクチャと等しいTemporalIdおよび現在のピクチャと等しいnuh\_layer\_idを有するSTSAピクチャに続くピクチャである場合、復号順にSTSAピクチャに先行し、現在のピクチャと等しいTemporalIdおよびRefPicList

10

20

30

40

50

[ 0 ] またはRefPicList [ 1 ] にアクティブエントリとして含まれる現在のピクチャと等しいnuh\_layer\_idを有するピクチャは存在しないものとする。

( 9 ) 特定の値tIdに等しいTemporalId、特定の値layerIdに等しいnuh\_layer\_id、および特定の値subpicIdxに等しいサブピクチャインデックスを有する現在のサブピクチャが、復号順に、tIdに等しいTemporalId、layerIdに等しいnuh\_layer\_id、およびsubpicIdxに等しいサブピクチャインデックスを有するSTSAサブピクチャに続くサブピクチャである場合、RefPicList [ 0 ] またはRefPicList [ 1 ] にアクティブエントリとして含まれる復号順にSTSAサブピクチャを含むピクチャに先行する、tIdに等しいTemporalId、およびlayerIdに等しいnuh\_layer\_idを有するピクチャは存在しないものとする。

( 10 ) 特定の値layerIdに等しいnuh\_layer\_idを有する現在のピクチャがIRAPピクチャであるとき、出力順または復号順において、( 存在する場合 ) 復号順においてlayerIdに等しいnuh\_layer\_idを有する先行するIRAPピクチャに先行するRefPicList [ 0 ] またはRefPicList [ 1 ] 内のエントリによって参照されるピクチャは存在しないものとする。 10

( 11 ) 特定の値layerIdに等しいnuh\_layer\_idおよび、特定の値subpicIdxに等しいサブピクチャインデックスを有する現在のサブピクチャがIRAPサブピクチャであるとき、出力順または復号順において、( 存在する場合 ) 復号順においてlayerIdに等しいnuh\_layer\_idおよびsubpicIdxに等しいサブピクチャインデックスを有するIRAPサブピクチャを含む任意の先行するピクチャに先行するRefPicList [ 0 ] またはRefPicList [ 1 ] 内のエントリによって参照されるピクチャはないものとする。

( 12 ) 現在のピクチャが、NoOutputBeforeRecoveryFlagが1であるCRAピクチャに関連付けられたRASLピクチャでない場合、現在のピクチャに関連付けられたCRAピクチャのための利用できない参照ピクチャを生成するための復号処理によって生成されたRefPicList [ 0 ] またはRefPicList [ 1 ] 内のアクティブエントリによって参照されるピクチャは存在しないものとする。 20

( 13 ) 現在のサブピクチャが、NoOutputBeforeRecoveryFlagが1であるCRAピクチャ内のCRAサブピクチャに関連付けられたRASLサブピクチャでない場合、現在のサブピクチャに関連付けられたCRAサブピクチャを含むCRAピクチャ用の利用できない参照ピクチャを生成するための復号処理によって生成されたRefPicList [ 0 ] またはRefPicList [ 1 ] 内のアクティブエントリによって参照されるピクチャはないものとする。

( 14 ) 特定の値layerIdに等しいnuh\_layer\_idを有する現在のピクチャが以下のいずれでもない場合、現在のピクチャに関連付けられたIRAPピクチャまたはGDRピクチャのための利用できない参照ピクチャを生成するための復号処理によって生成されたRefPicList [ 0 ] またはRefPicList [ 1 ] 内のエントリによって参照されるピクチャは存在しないはずである。 30

( a ) sps\_idr\_rpl\_present\_flagが1またはpps\_rpl\_info\_in\_ph\_flagが1であるIDRピクチャ。

( b ) NoOutputBeforeRecoveryFlagが1であるCRAピクチャ。

( c ) NoOutputBeforeRecoveryFlagが1であるCRAピクチャに関連付けられたピクチャであって、同じCRAピクチャに関連付けられた先頭のピクチャよりも復号順が早いピクチャ。

( d ) NoOutputBeforeRecoveryFlagが1であるCRAピクチャに関連付けられた先頭ピクチャ。 40

( e ) NoOutputBeforeRecoveryFlagが1であるGDRピクチャ。

( f ) NoOutputBeforeRecoveryFlagが1であり、nuh\_layer\_idがlayerIdであるGDRピクチャの復元ピクチャ。

( 15 ) nuh\_layer\_idが特定の値layerIdに等しく、サブピクチャインデックスが特定の値subpicIdxに等しい、現在のサブピクチャが以下のいずれでもない場合、現在のサブピクチャに関連付けられたIRAPまたはGDRサブピクチャを含むIRAPまたはGDRピクチャ用の利用できない参照ピクチャを生成するための復号処理によって生成されたRefPicList [ 0 ] またはRefPicList [ 1 ] 内のエントリによって参照されるピクチャはないはずである 50

。

( a ) `sps_idr_rpl_present_flag`が1または`pps_rpl_info_in_ph_flag`が1であるIDRピクチャ内のIDRサブピクチャ。

( b ) `NoOutputBeforeRecoveryFlag`が1であるCRAピクチャ内のCRAサブピクチャ。

( c ) `NoOutputBeforeRecoveryFlag`が1であるCRAピクチャ内のCRAサブピクチャに関連付けられたサブピクチャであって、同じCRAピクチャに関連付けられた先頭ピクチャよりも復号順が早いサブピクチャ。

( d ) `NoOutputBeforeRecoveryFlag`が1であるCRAピクチャ内のCRAサブピクチャに関連付けられた先頭サブピクチャ。

( e ) `NoOutputBeforeRecoveryFlag`が1であるGDRピクチャ内のGDRサブピクチャ。

( f ) `NoOutputBeforeRecoveryFlag`が1であり、`nuh_layer_id`が`layerId`であるGDRピクチャの復元ピクチャ内のサブピクチャ。

( 16 ) 現在のピクチャが、復号順および出力順の両方で`nuh_layer_id`の値が同じIRAPピクチャに続くとき、そのIRAPピクチャの出力順または復号順に先行するRefPicList [ 0 ] またはRefPicList [ 1 ] 内のアクティブエントリによって参照されるピクチャはないものとする。

( 17 ) 現在のサブピクチャが、復号順と出力順の両方で同じ値の`nuh_layer_id`および同じ値のサブピクチャインデックスを有するIRAPサブピクチャに続くとき、そのIRAPサブピクチャを含むピクチャに出力順または復号順で先行するRefPicList [ 0 ] またはRefPicList [ 1 ] 内のアクティブエントリによって参照されるピクチャはないものとする。

( 18 ) 現在のピクチャが、`nuh_layer_id`の値が同じであるIRAPピクチャと、もしあれば、そのIRAPピクチャに復号順および出力順の両方で関連付けられている先頭ピクチャとを辿るとき、そのIRAPピクチャに出力順または復号順で先行するRefPicList [ 0 ] またはRefPicList [ 1 ] 内のエントリによって参照されるピクチャは存在しないものとする。

( 19 ) 現在のサブピクチャが、同じ値の`nuh_layer_id`および同じ値のサブピクチャインデックスを有するIRAPサブピクチャと、もしあれば、復号順と出力順の両方でそのIRAPサブピクチャに関連付けられた先頭サブピクチャとに続くとき、出力順または復号順でそのIRAPサブピクチャを含むピクチャに先行するRefPicList [ 0 ] またはRefPicList [ 1 ] 内のエントリによって参照されるピクチャはないものとする。

( 20 ) 現在のピクチャがRADLピクチャである場合、以下のいずれかであるRefPicList [ 0 ] またはRefPicList [ 1 ] にアクティブエントリは存在しないものとする。

( a ) `pps_mixed_nalu_types_in_pic_flag`のRASLピクチャは0である。これは、RADLピクチャのRPLのアクティブエントリが、1に等しい`pps_mixed_nalu_types_in_pic_flag`を有するRASLピクチャを参照し得ることを意味する。しかしながら、RADLピクチャは、RASLサブピクチャを参照するRADLサブピクチャを許可しない次の制約によって課されるように、参照されるRASLピクチャ内のRADLサブピクチャのみを参照するため、参照されるRASLピクチャ内のRADLサブピクチャが正しく復号されることから、関連するCRAピクチャから復号が開始されるとき、そのようなRADLピクチャは依然として正しく復号され得る。

( b ) 関連付けられたIRAPピクチャの復号順に先行するピクチャ。

( 21 ) 特定の値`layerId`に等しい`nuh_layer_id`および特定の値`subpicIdx`に等しいサブピクチャインデックスを有する現在のサブピクチャがRADLサブピクチャであるとき、以下のいずれかであるRefPicList [ 0 ] またはRefPicList [ 1 ] 内にアクティブエントリはないものとする。

( a ) `nuh_layer_id`が`layerId`に等しいピクチャは、`subpicIdx`に等しいサブピクチャインデックスを有するRASLサブピクチャを含む。

( b ) 関連付けられたIRAPサブピクチャを含むピクチャの復号順に先行するピクチャ。

( 22 ) 各ILRPエントリによって参照されるピクチャには、現在のピクチャのスライスのRefPicList [ 0 ] またはRefPicList [ 1 ] に存在する場合、以下の制約が適用される。

( a ) ピクチャは、現在のピクチャと同じAUにあるものとする。

10

20

30

40

50

- (b) ピクチャは、DPBに存在するものとする。
- (c) ピクチャは、現在のピクチャのnuh\_layer\_idよりも小さいnuh\_layer\_id refPicLayerIdを有するものとする。
- (d) 以下の制約のいずれかが適用される：ピクチャはIRAPピクチャであるものとする；ピクチャはMax(0, vps\_max\_tid\_il\_ref\_pics\_plus1[currLayerIdx][refLayerIdx] - 1)以下のTemporalIdを有するものとし、currLayerIdxおよびrefLayerIdxは、それぞれGeneralLayerIdx[nuh\_layer\_id]およびGeneralLayerIdx[refpicLayerId]に等しい。

(23) スライスのRefPicList[0]またはRefPicList[1]内に存在する場合、各ILRPエントリはアクティブエントリであるものとする。 10

(24) vps\_independent\_layer\_flag[GeneralLayerIdx[nuh\_layer\_id]]が0に等しく、sps\_num\_subpics\_minus1が0より大きい場合、以下の2つの条件のいずれか（両方ではない）が真であるものとする。

(a) RefPicList[0]またはRefPicList[1]の各アクティブエントリが参照するピクチャは、現在のピクチャと同一のサブピクチャ・レイアウトを有する（すなわち、そのピクチャおよび現在のピクチャによって参照されるSPSは、0からsps\_num\_subpics\_minus1までの範囲内のjの各値に対して、それぞれ、同じ値のsps\_num\_subpics\_minus1と、同じ値のsps\_subpic\_ctu\_top\_left\_x[j]、sps\_subpic\_ctu\_top\_left\_y[j]、sps\_subpic\_width\_minus1[j]、およびsps\_subpic\_height\_minus1[j]とを有する）。

(b) RefPicList[0]またはRefPicList[1]の各アクティブエントリが参照するピクチャは、sps\_num\_subpics\_minus1の値が0であるILRPである。 20

#### 【0321】

実施形態によれば、復号器は、ビットストリーム適合性について上記の制約が満たされているか否かを確認することができる。適合性確認は、上述したように、利用できない参照ピクチャを生成するための復号処理を呼び出した後に行われ得る。

#### 【0322】

本開示の実施形態は、コンピュータコードを記憶する少なくとも1つのプロセッサおよびメモリを備えることができる。コンピュータコードは、少なくとも1つのプロセッサによって実行されると、少なくとも1つのプロセッサに本開示の実施形態の機能を実行させるように構成されてもよい。 30

#### 【0323】

例えば、図33を参照すると、本開示の復号器は、コンピュータコード(80)を格納する少なくとも1つのプロセッサおよびメモリを備えることができる。復号器は、少なくとも1つの符号化されたピクチャおよびパラメータセット（例えば、SPSおよびVPS）、ヘッダ（例えば、ピクチャヘッダおよびスライスヘッダ）、およびAUデリミタを含むビットストリームを受信するように構成され得る。このコンピュータコードは、少なくとも1つのプロセッサに対して、本開示に記載されたような任意の数の復号処理（例えば、参照ピクチャリストの構築、利用できない参照ピクチャの生成、およびビットストリーム適合性の確認）および復号に関連する態様（例えば、ピクチャヘッダ、スライスヘッダ、およびアクセス・ユニット・デリミタにおけるフラグおよび他の構文要素のシグナリング）を実行せ得るよう構成され得る。例えば、コンピュータコード(80)は、複数のシグナリングコード(81)および復号コード(82)を含み得る。 40

#### 【0324】

複数のシグナリングコード(81)は、少なくとも1つのプロセッサに、ピクチャヘッダ、スライスヘッダ、およびアクセス・ユニット・デリミタ内のフラグおよび他の構文要素をシグナリング（および／または推論）せ得るように構成された様々なシグナリングコードを含み得る。

#### 【0325】

復号コード(82)は、少なくとも1つのプロセッサに、1つまたは複数のピクチャを復

号させるように構成され得る。実施形態によれば、復号コード(82)は、構築コード(83)と、生成コード(84)と、確認コード(85)とを含み得る。構築コード(83)は、少なくとも1つのプロセッサに、参照ピクチャリストを構築せしめるように構成され得る。生成コード(84)は、少なくとも1つのプロセッサに、参照ピクチャリスト内の利用できない参照ピクチャを生成せしめるように構成され得る。確認コード(85)は、少なくとも1つのプロセッサに、参照ピクチャリスト内の参照ピクチャについて、以下の制約、すなわち、(a) 参照ピクチャリスト内にあると示されるエントリの数が、参照ピクチャリスト内にあると示されるアクティブエントリの数以上であること、(b) 参照ピクチャリスト内のアクティブエントリによって参照される各ピクチャが、復号ピクチャバッファ(DPB)内に存在し、現在のピクチャの時間識別子値以下の時間識別子値を有すること、および(c) 参照ピクチャリスト内のエントリによって参照される各ピクチャが、現在のピクチャではなく、参照ピクチャである可能性があることを示すピクチャ・ヘッダ・フラグによって示されること、が適用されるビットストリーム適合性を確認せしめるように構成されてもよい。

#### 【0326】

上述した技術は、コンピュータ可読命令を使用し、1つまたは複数のコンピュータ可読媒体に物理的に記憶されたコンピュータソフトウェアとして実装することができる。例えば、図34は、開示された主題の実施形態を実装するのに適したコンピュータシステム(900)を示している。

#### 【0327】

コンピュータソフトウェアは、任意の適切な機械コードまたはコンピュータ言語を使用して符号化でき、アセンブリ、コンパイル、リンク、または同様の機構の対象となり、1つまたは複数のコンピュータ中央処理装置(CPU)、グラフィック処理装置(GPU)などによる直接、または解釈、マイクロコードの実行などを通じて実行できる命令を含むコードを作成する。

#### 【0328】

命令は、例えば、パーソナルコンピュータ、タブレットコンピュータ、サーバ、スマートフォン、ゲームデバイス、モノのインターネット装置などを含む様々な種類のコンピュータまたはその構成要素上で実行することができる。

#### 【0329】

コンピュータシステム(900)について図34に示される構成要素は、本質的に例示であり、本開示の実施形態を実装するコンピュータソフトウェアの使用または機能の範囲についていかなる制限を示唆することを意図しない。また、構成要素の構成は、コンピュータシステム(900)の例示的な実施形態に示されている構成要素のいずれか1つまたは組み合わせに関する依存性または要件を有するものとして解釈されるべきではない。

#### 【0330】

コンピュータシステム(900)は、特定のヒューマンインターフェース入力デバイスを含み得る。そのようなヒューマンインターフェース入力デバイスは、例えば、触覚入力(キーストローク、スワイプ、データグローブの動作など)、音声入力(声、拍手など)、視覚入力(ジェスチャなど)、嗅覚入力(図示せず)など、1人または複数のユーザによる入力に応答し得る。ヒューマン・インターフェース・デバイスを使用して、音声(スピーチ、音楽、環境音など)、画像(走査した画像、静止画像カメラから得られるピクチャ画像など)、ビデオ(2次元ビデオ、立体ビデオを含む3次元ビデオなど)など、人間による意識的な入力に必ずしも直接関係しない特定の媒体をキャプチャすることもできる。

#### 【0331】

入力ヒューマン・インターフェース・デバイスには、キーボード901、マウス902、トラックパッド903、タッチスクリーン910、データグローブ、ジョイスティック905、マイク906、スキャナ907、およびカメラ908のうち1つまたは複数(それぞれ図示のものの1つのみ)が含まれ得る。

#### 【0332】

10

20

30

40

50

コンピュータシステム（900）はまた、特定のヒューマンインターフェース出力デバイスを含み得る。そのようなヒューマンインターフェース出力デバイスは、例えば、触覚出力、音、光、および匂い／味によって1人または複数の人間のユーザの感覚を刺激することができる。そのようなヒューマンインターフェース出力デバイスは、触覚出力デバイス（例えば、タッチスクリーン（910）、データグローブ、またはジョイスティック（905）による触覚フィードバックであるが、入力デバイスとして機能しない触覚フィードバックデバイスも存在し得る）を含むことができる。例えば、そのようなデバイスは、音声出力デバイス（例えば、スピーカ（909）、ヘッドホン（図示せず））、視覚出力装置（例えば、CRTスクリーン、LCDスクリーン、プラズマスクリーン、OLEDスクリーンを含むスクリーン（910）であって、それぞれがタッチスクリーン入力機能を有するか又は有さず、それが触覚フィードバック機能を有するか又は有さず、その一部は、ステレオ出力などの手段を介して二次元視覚出力又は三次元超出手力を出力することができてもよい、スクリーン、仮想現実メガネ（図示せず）、ホログラフィックディスプレイ、および煙タンク（図示せず）、およびプリンタ（図示せず）であってもよい。

#### 【0333】

コンピュータシステム（900）には、人間がアクセスできる記憶デバイスと、CD / DVDを含むCD / DVD ROM / RW（920）などの光学メディア（921）、サムドライブ（922）、リムーバブル・ハード・ドライブまたはソリッド・ステート・ドライブ（923）、テープおよびフロッピーディスク（図示せず）などのレガシー磁気媒体、セキュリティドングル（図示せず）などの専用のROM / ASIC / PLDベースのデバイスなどの関連媒体も含めることができる。

#### 【0334】

当業者はまた、ここで開示される主題に関連して使用される「コンピュータ可読媒体」という用語は、送信媒体、搬送波、または他の一時的な信号を包含しないことを理解するべきである。

#### 【0335】

コンピュータシステム（900）は、1つまたは複数の通信ネットワークへのインターフェースも含み得る。ネットワークは、例えば、無線、有線、光であり得る。ネットワークはさらに、ローカル、広域、メトロポリタン、車両および産業、リアルタイム、遅延耐性などであり得る。ネットワークの例としては、イーサネット、無線LAN、GSM、3G、4G、5G、LTEなどを含むセルラーネットワークなどのローカル・エリア・ネットワーク、ケーブルテレビ、衛星テレビ、地上波放送テレビを含むTV有線または無線広域デジタルネットワーク、CANBusなどが含まれる車両用、産業用など、がある。特定のネットワークでは、一般に、特定の汎用データポートまたは周辺バス（949）（例えば、コンピュータシステム（900）のUSBポートなど）に接続された外部ネットワーク・インターフェース・アダプタが必要であり、他のものは一般に、以下に説明するようにシステムバスに接続することにより、コンピュータシステム900のコアに統合される（例えば、PCコンピュータシステムへのイーサネットインターフェースまたはスマートフォン・コンピュータ・システムへのセルラー・ネットワーク・インターフェース）。これらのネットワークのいずれかを使用して、コンピュータシステム（900）は他のエンティティと通信できる。このような通信は、単方向、受信のみ（例えば、放送TV）、単方向送信のみ（例えば、CANbusから特定のCANbusデバイス）、または双方向、例えば、ローカル・エリア・デジタル・ネットワークまたはワイド・エリア・デジタル・ネットワークを使用する他のコンピュータシステムへの通信であり得る。そのような通信は、クラウドコンピューティング環境（955）への通信を含むことができる。特定のプロトコルおよびプロトコルスタックは、上述したように、それらのネットワークおよびネットワークインターフェースのそれぞれで使用することができる。

#### 【0336】

前述のヒューマン・インターフェース・デバイス、ヒューマンアクセス可能な記憶デバイス、およびネットワークインターフェース（954）は、コンピュータシステム（900）

のコア(940)に接続することができる。

#### 【0337】

コア(940)には、1つまたは複数の中央処理装置(CPU)(941)、グラフィックス処理装置(GPU)(942)、フィールド・プログラマブル・ゲート・エリア(FPGA)(943)、特定のタスクのハードウェアアクセラレータ(944)などの形式の特殊なプログラマブル処理装置を含めることができる。これらのデバイスは、読み取り専用メモリ(ROM)(945)、ランダム・アクセス・メモリ(946)、ユーザがアクセスできない内部ハードドライブ、SSDなどの内部大容量記憶装置(947)とともに、システムバス(948)を介して接続され得る。いくつかのコンピュータシステムでは、システムバス(948)に1つまたは複数の物理プラグの形でアクセスして、追加のCPU、GPUなどによる拡張を可能にすることができる。周辺デバイスは、コアのシステムバス(948)に直接、または周辺バス(949)を介して接続できる。周辺バスのアーキテクチャには、PCI、USBなどが含まれる。グラフィックスアダプタ950は、コア940に含まれてもよい。

#### 【0338】

CPU(941)、GPU(942)、FPGA(943)、およびアクセラレータ(944)は、組み合わせて前述のコンピュータコードを構成できる特定の命令を実行できる。そのコンピュータコードは、ROM(945)またはRAM(946)に記憶することができる。移行データはまた、RAM(946)に記憶することができ、一方、永続データは、例えば内部大容量記憶装置(947)に記憶することができる。1つまたは複数のCPU(941)、GPU(942)、大容量記憶装置(947)、ROM(945)、RAM(946)などと密接に関連付けることができるキャッシュメモリを使用することにより、任意のメモリデバイスに対する高速記憶および読み出しが可能になる。

#### 【0339】

コンピュータ可読媒体は、様々なコンピュータ実装動作を実行するためのコンピュータコードを有することができる。媒体およびコンピュータコードは、本開示の目的のために特別に設計および構築されたものであってもよく、またはコンピュータソフトウェア技術の当業者に周知で利用可能な種類のものであってもよい。

#### 【0340】

限定ではなく例として、アーキテクチャ(900)、特にコア(940)を有するコンピュータシステムは、1つまたは複数の有形のコンピュータ可読媒体に組み込まれたソフトウェアを実行するプロセッサ(CPU、GPU、FPGA、アクセラレータなどを含む)の結果として機能を提供できる。このようなコンピュータ可読媒体は、上で紹介したユーザがアクセス可能な大容量記憶装置、およびコア内部大容量記憶装置(947)やROM(945)などの非一時的な性質を持つコア(940)の特定の記憶装置に関連付けられた媒体であり得る。本開示の様々な実施形態を実装するソフトウェアは、そのようなデバイスに記憶され、コア(940)によって実行され得る。コンピュータ可読媒体は、特定のニーズに従って、1つまたは複数のメモリデバイスまたはチップを含み得る。ソフトウェアは、コア(940)、特にその中のプロセッサ(CPU、GPU、FPGAなどを含む)に、RAM(946)に格納されているデータ構造を定義すること、およびソフトウェアで定義された処理に従ってそのようなデータ構造を変更することを含む、ここで説明する特定の処理または特定の処理の特定の部分を実行させることができる。加えて、または代替として、コンピュータシステムは、ハードワイヤードまたは他の方法で回路(例えば、アクセラレータ(944))に具現化された論理の結果として機能を提供することができ、ソフトウェアの代わりに、またはソフトウェアと共に動作して、本明細書に記載の特定の処理または特定の処理の特定の部分を実行することができる。ソフトウェアへの参照はロジックを含むことができ、その逆も適宜可能である。コンピュータ可読媒体への言及は、必要に応じて、実行のためのソフトウェアを記憶する回路(集積回路(IC)など)、実行のための論理を具現化する回路、またはその両方を包含することができる。本開示は、ハードウェアとソフトウェアとの任意の適切な組み合わせを包含する。

#### 【0341】

10

20

30

40

50

本開示はいくつかの例示的な実施形態を説明してきたが、本開示の範囲内にある変更、置換、および様々な代替均等物が存在する。したがって、当業者は、本明細書に明示的に示されていないまたは記載されていないが、本開示の原理を具体化し、したがってその趣旨および範囲内にある多数のシステムおよび方法を考案することができることが理解されよう。

【符号の説明】

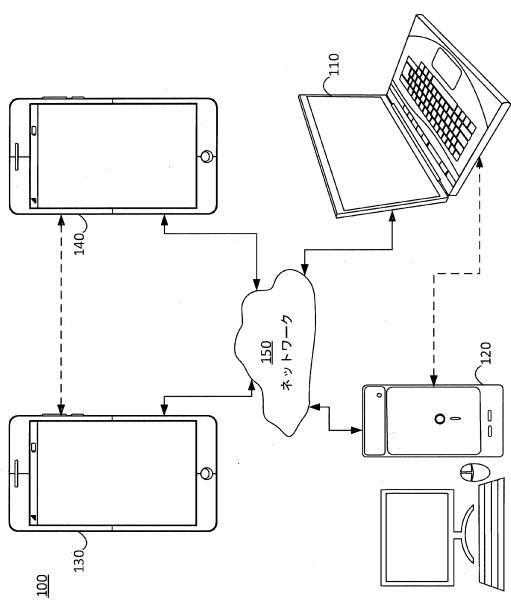
【0 3 4 2】

80	コンピュータコード	
81	シグナリングコード	
82	復号コード	10
83	構築コード	
84	生成コード	
85	確認コード	
100	通信システム	
110	端末	
120	端末	
130	端末	
140	端末	
150	ネットワーク	
200	ストリーミングシステム	20
201	ビデオソース	
202	非圧縮ビデオ・サンプル・ストリーム	
203	符号化器	
204	ビデオ・ビットストリーム	
205	ストリーミングサーバ	
206	ストリーミングクライアント	
209	ビデオ・ビットストリーム	
210	ビデオ復号器	
211	出力ビデオ・サンプル・ストリーム	
212	ディスプレイ	30
310	受信機	
312	チャネル	
315	バッファメモリ	
320	解析器	
321	シンボル	
351	スケーラ / 逆変換	
352	イントラ画像予測	
353	動作補償予測ユニット	
356	ループフィルタ	
357	参照画像メモリ	40
358	現在のピクチャ	
430	ソース符号化器	
432	符号化エンジン	
433	復号器	
434	参照画像メモリ	
435	予測器	
440	送信機	
445	エントロピー符号化器	
450	コントローラ	
460	チャネル	50

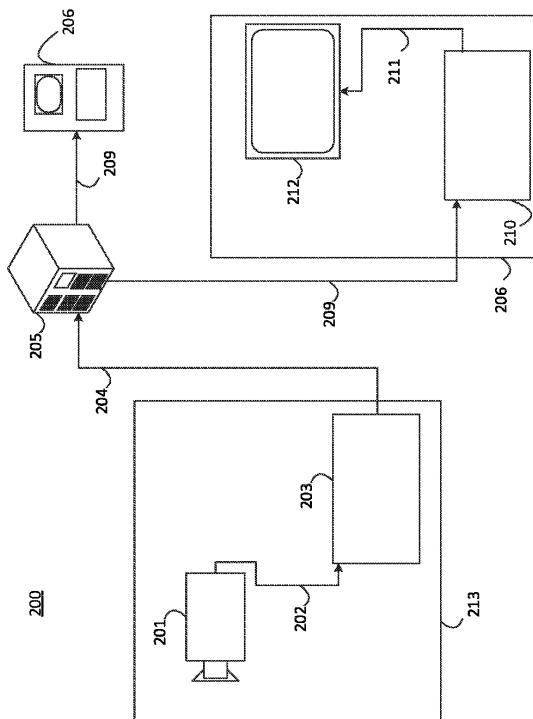
501	ピクチャヘッダ	
502	ARC情報	
503	ARC(ワーピング座標)	
504	ピクチャ・パラメータ・セット	
505	ARC参照情報	
506	表	
507	シーケンス・パラメータ・セット	
508	ヘッダ	
509	ARC情報	
511	適応パラメータセット	10
512	ARC情報	
513	ARC参照情報	
514	タイル・グループ・ヘッダ	
515	ARC情報	
600	タイル・グループ・ヘッダ	
516	パラメータセット	
602	構文要素dec_pic_size_idx	
603	適応型解像度	
610	シーケンス・パラメータ・セット	
611	構文要素	20
613	出力解像度	
615	参照ピクチャ寸法	
617	表エントリ	
681	Iスライス	
682	Bスライス	
683	Bスライス	
684	Bスライス	
685	Bスライス	
686	Bスライス	
702	ベース層	30
712	ベース層	
704	拡張CSPS層	
714	拡張CSPS層	
742	球面360ピクチャ	
745	サブピクチャ	
752	サブピクチャ	
746	サブピクチャの拡張層	
762	左上サブ領域	
763	右上サブ領域	
764	左下サブ領域	40
765	右下サブ領域	
941	CPU	
942	GPU	
943	フィールド・プログラマブル・ゲート・エリア(FPGA)	
944	ハードウェアアクセラレータ	
948	システムバス	
950	グラフィックアダプタ	
954	ネットワークインターフェース	

【図面】

【図 1】



【図 2】



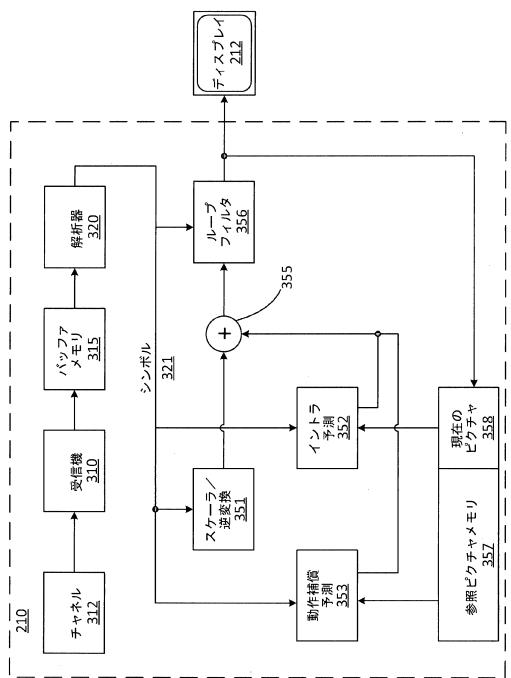
10

20

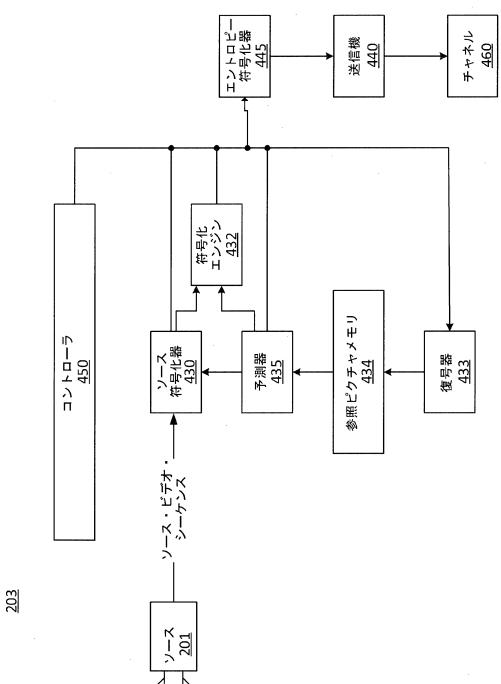
30

40

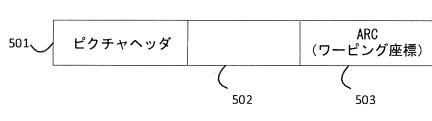
【図 3】



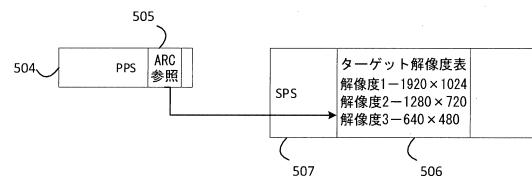
【図 4】



【図 5 A】

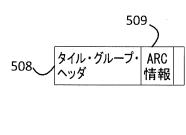


【図 5 B】



10

【図 6 A】

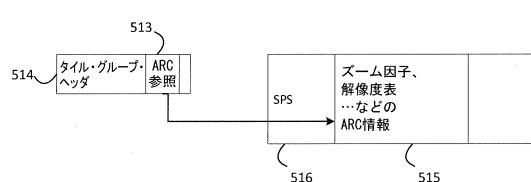


【図 6 B】

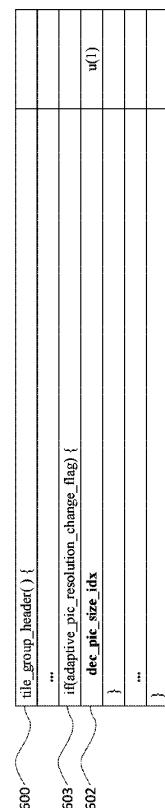


20

【図 6 C】



【図 7 A】



30

40

50

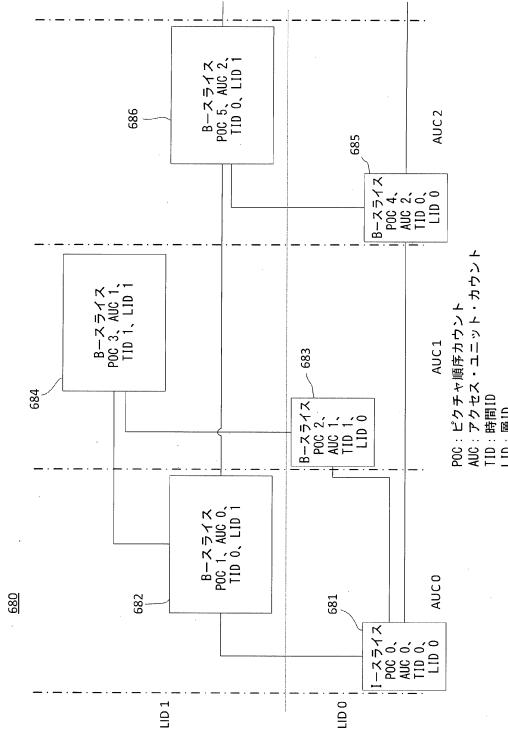
【図7B】

```

610 seq_parameter_set_resph() {
611     ...
612     adaptive.pic_resolution_change_flag
613     if(adaptive.pic_resolution_change_flag) {
614         output.pic_width_in_luma_samples
615         output.pic_height_in_luma_samples
616         reference.pic_size_present_flag
617         if(reference.pic_size_present_flag)
618             {
619                 reference.pic_width_in_luma_samples
620                 reference.pic_height_in_luma_samples
621             }
622         num_dec_pic_size_in_luma_samples_minus1
623         for (i = 0; i <= num_dec_pic_size_in_luma_samples_minus1; i++) {
624             dec.pic_width_in_luma_samples[i]
625             dec.pic_height_in_luma_samples[i]
626         }
627     }
628     ...
}

```

【 义 8 】



【図 9 A】

```

630   video parameter_set_idsp0 {
    vps.video_parameter_set_id
    vps.max_layers_minus1
    for(i=0; i < vps.max_layers_minus1; i++) {
      vps.included_layer_id[i]
      vps.reserved_zero_bit
    }
  }

632   vps_constraint_info_present_flag
   vps_constraint_pcoc_cycle_per_an
   if(vps_constraint_pcoc_cycle_per_an)
     vps_pcoc_cycle_an
   ...
}

634

```

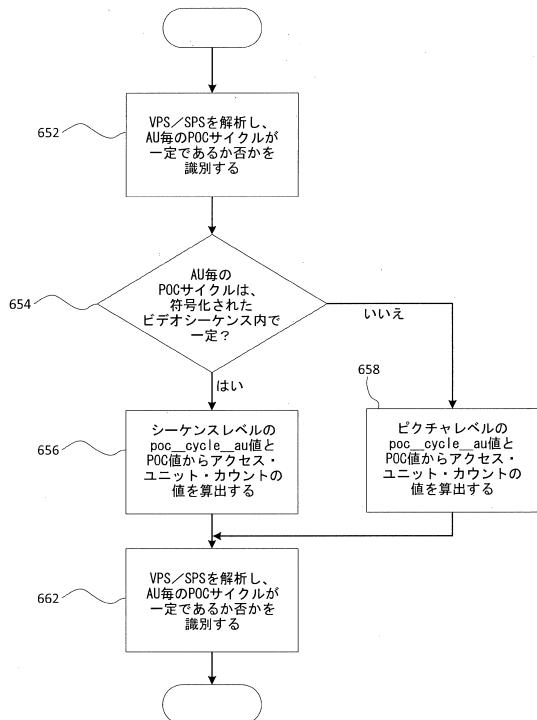
【図9B】

```

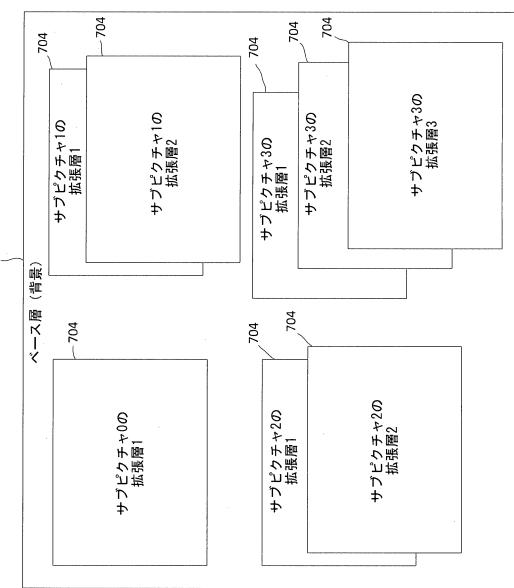
640 slice_header() {
    slice.pic_parameter_set_id
    if(rect_slice_flag || NumBricksInPic>1)
        slice_address
        if(!rect_slice_flag && !single_brick_per_slice_flag)
            num_bricks_in_slice_minus1
    slice_type
    if(NalUnitType == GRA_NUT)
        recovery_pic_cnt
        slice.pic_order_cnt_lsb
        if(lvps_constant_pic_cycle_per_au)
            slice.pic_cycle_nu
}

```

【図10】



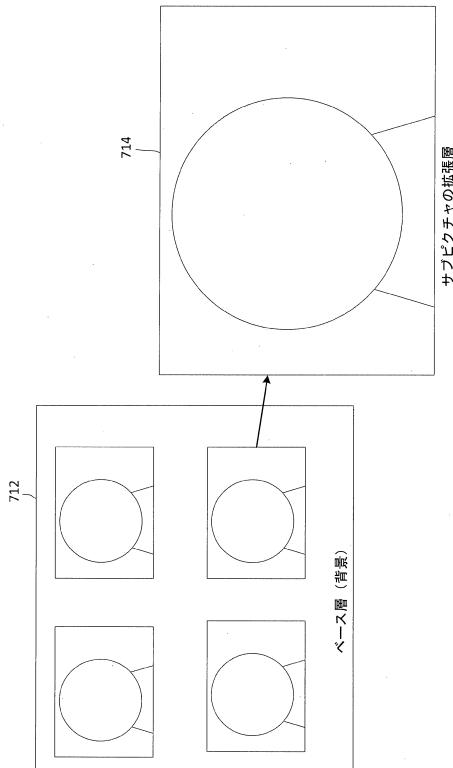
【図11】



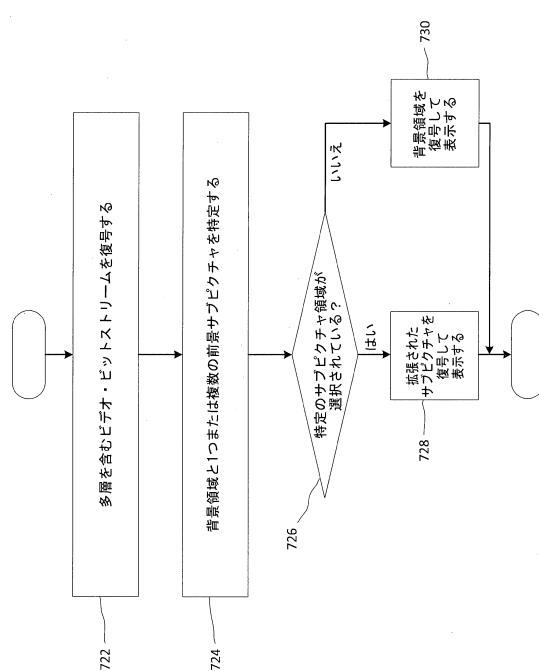
10

20

【図12】



【図13】

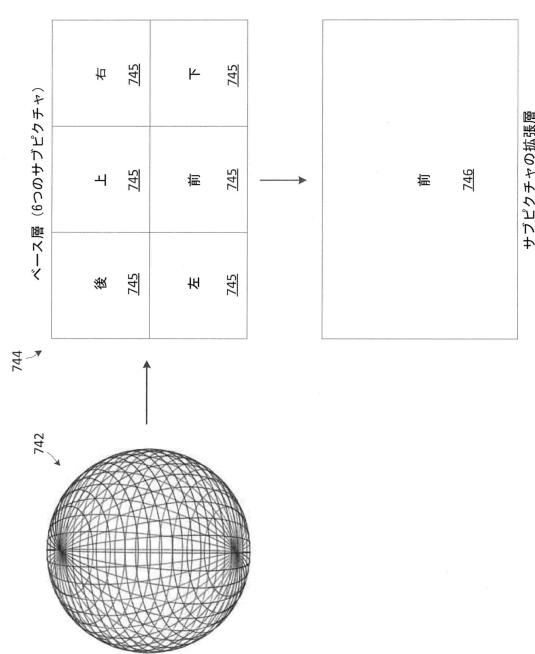


30

40

50

【図 1 4】



【図 1 5 A】

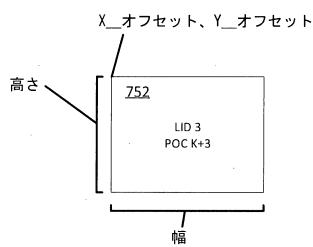
752	LID 0 POC K	LID 1 POC K+1
752	LID 2 POC K+2	LID 3 POC K+3

FIG. 15A

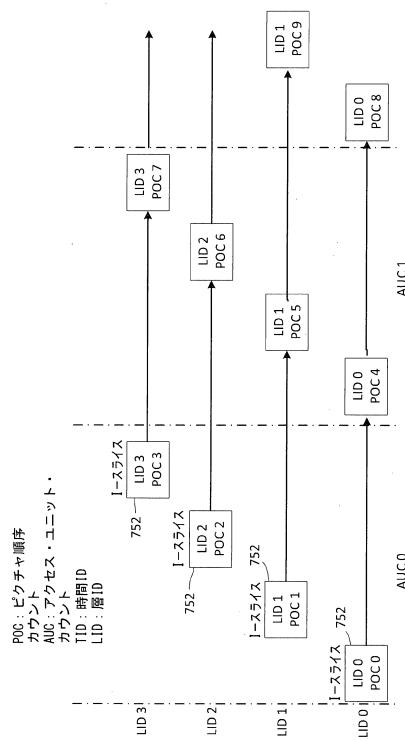
10

20

【図 1 5 B】



【図 1 6】



30

40

50

【図 1 7】

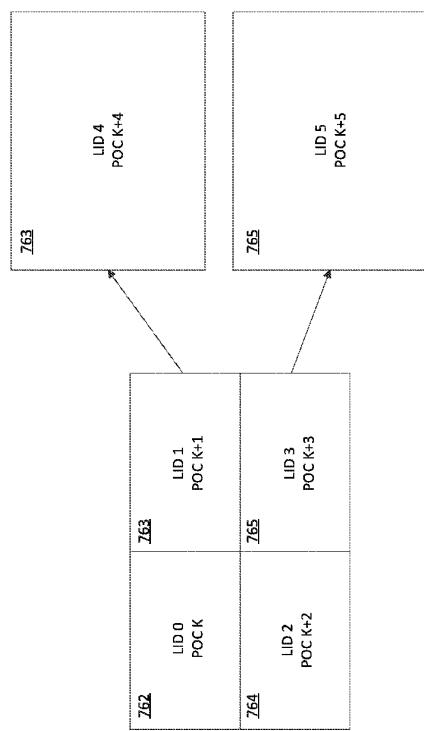
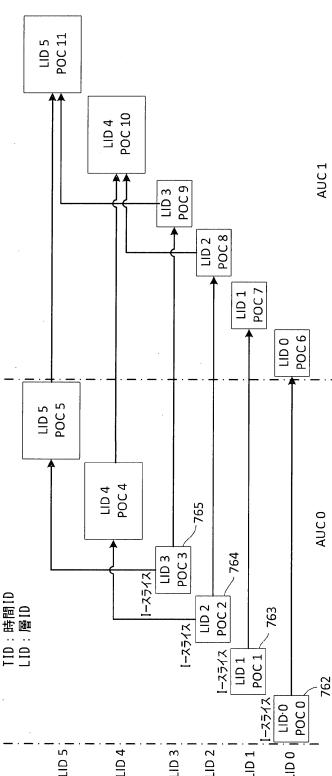


FIG. 17

【図 1 8】



10

【図 1 9 A】

```

video_parameter_set_rbsp() {
    vps_video_parameter_set_id
    vps_max_layers_minus1
    for(i=0; i <= vps_max_layers_minus1; i++) {
        vps_included_layer_id[i]
        vps_reserved_zero_bit
    }
    ...
    vps_sub_picture_dividing_flag
    if(vps_sub_picture_dividing_flag) {
        vps_full_pic_width_in_luma_samples
        vps_full_pic_height_in_luma_samples
    }
    ...
}

```

【図 1 9 B】

```

seq_parameter_set_rbsp() {
    sps_decoding_parameter_set_id
    sps_video_parameter_set_id
    sps_max_sub_layers_minus1
    ...
    pic_width_in_luma_samples
    pic_height_in_luma_samples
    if(vps_sub_picture_dividing_flag) {
        pic_offset_x
        pic_offset_y
    }
    ...
}

```

20

30

40

50

【図 2 0】

```

sub_region_partitioning_info(payloadSize) {
    記述子
    ue(v)
    ue(v)

    num_sub_region
    num_layers
    for (i = 0; i <= num_layers; i++) {
        layer_id[i]
        for (j = 1; j < num_layers; j++) {
            for(j = 0; j < i;j++)
                direct_dependency_flag[i][j]
        }
        for (i = 0; i < num_sub_region; i++) {
            num_layers_for_region[i]
            for(j = 0; j < num_layers_for_region[i];j++)
                sub_region_layer_id[i][j]
        }
        sub_region_offset_x[i]
        sub_region_offset_y[i]
        sub_region_width[i]
        sub_region_height[i]
    }
    ...
}

```

【図 2 1】

```

video_parameter_set_rbsp() {
    記述子
    ue(v)
    ue(v)

    vs_max_layers_minus1
    if(vs_max_layers_minus1 > 0) {
        num_output_layer_sets
        num_profile_tier_level
    }
    for(i = 0; i < num_profile_tier_level; i++) {
        profile_tier_level_idx[i]
        if(cps_output_layers_mode[i] == 2) {
            output_layer_flag[i]
            if(vps_ntl_signal_theta[i])
                profile_tier_level_idx[i]
        }
    }
    ...
}

vs_max_layers_minus1
if(vs_max_layers_minus1 > 0) {
    num_output_layer_sets
    num_profile_tier_level
}
max_subpics_minus1
for(i = 0; i < max_subpics_minus1; i++) {
    sub_pic_id[i]
}
for(i = 0; i < num_profile_tier_level; i++) {
    profile_tier_level_vs_max_sub_layers_minus1
    for(i = 0; i < num_output_layer_sets; i++) {
        vs_output_layers_mode[i]
        if(pic_signal_flag[i])
            for(i = 0; i < NumLayersInIdxs[i]; i++) {
                num_output_subpic_layerfill[i]
                if(cps_ntl_signal_flag[i])
                    sub_pic_id[layerfill[i].k]
                    if(pic_output_layers_mode[i] == 2)
                        output_layer_flag[i]
                    if(cps_ntl_signal_flag[i])
                        profile_tier_level_idx[i]
                }
            }
        ...
}

```

【図 2 2】

```

video_parameter_set_rbsp() {
    記述子
    ue(v)
    ue(v)

    vs_max_layers_minus1
    if(vs_max_layers_minus1 > 0) {
        num_output_layer_sets
        num_profile_tier_level
    }
    max_subpics_minus1
    for(i = 0; i < max_subpics_minus1; i++) {
        sub_pic_id[i]
    }
    for(i = 0; i < num_profile_tier_level; i++) {
        profile_tier_level_vs_max_sub_layers_minus1
        for(i = 0; i < num_output_layer_sets; i++) {
            vs_output_layers_mode[i]
            if(pic_signal_flag[i])
                for(i = 0; i < NumLayersInIdxs[i]; i++) {
                    num_output_subpic_layerfill[i]
                    if(cps_ntl_signal_flag[i])
                        sub_pic_id[layerfill[i].k]
                        if(pic_output_layers_mode[i] == 2)
                            output_layer_flag[i]
                        if(cps_ntl_signal_flag[i])
                            profile_tier_level_idx[i]
                    }
                }
            ...
        }
    }
}

```

【図 2 3】

```

video_parameter_set_rbsp() {
    記述子
    ue(v)
    ue(v)

    vs_max_layers_minus1
    if(vs_max_layers_minus1 > 0) {
        num_output_layer_sets
        num_profile_tier_level
        for(i = 0; i < num_profile_tier_level; i++) {
            profile_tier_level_vs_max_sub_layers_minus1
            for(i = 0; i < num_output_layer_sets; i++) {
                vs_output_layers_mode[i]
                if(pic_signal_flag[i])
                    for(i = 0; i < NumLayersInIdxs[i]; i++) {
                        num_output_subpic_layerfill[i]
                        if(cps_ntl_signal_flag[i])
                            sub_pic_id[layerfill[i].k]
                            if(pic_output_layers_mode[i] == 2)
                                output_layer_flag[i]
                            if(cps_ntl_signal_flag[i])
                                profile_tier_level_idx[i]
                            ...
                        }
                    }
                ...
            }
        }
    }
}

```

【図 2 4】

video_parameter_set_rbsp( ) {		記述子
842	vps_video_parameter_set_id	u(4)
802	vps_max_layers_minus1	u(6)
846	vps_max_sublayers_minus1	u(3)
	if( vps_max_layers_minus1 > 0 && vps_max_sublayers_minus1 > 0 )	
848	vps_all_layers_same_num_sublayers_flag	u(1)
	if( vps_max_layers_minus1 > 0 )	
850	vps_all_independent_layers_flag	u(1)
	for( i=0; i <= vps_max_layers_minus1; i++ ) {	
852	vps_layer_id[i]	u(6)
	if( i > 0 && !vps_all_independent_layers_flag ) {	
854	vps_independent_layer_flag[i]	u(1)
	if( !vps_independent_layer_flag[i] ) {	
	for( j=0; j < i; j++ )	
	vps_direct_ref_layer_flag[i][j]	u(1)
856	max_tid_ref_present_flag[i]	u(1)
858	if( max_tid_ref_present_flag[i] )	
860	max_tid_il_ref_pics_plus1[i]	u(3)
	}	
	if( vps_max_layers_minus1 > 0 )	
	if( vps_all_independent_layers_flag )	
862	each_layer_is_an_ols_flag	u(1)
	if( each_layer_is_an_ols_flag )	
864	ols_mode_idc	u(2)
	if( ols_mode_idc == 2 ) {	
866	num_output_layer_sets_minus1	u(8)
	for( i=1; i <= num_output_layer_sets_minus1; i++ )	
	for( j=0; j <= vps_max_layers_minus1; j++ )	
868	ols_output_layer_flag[i][j]	u(1)
	}	
	}	
	}	
	...	
	}	

【図 2 5】

video_parameter_set_rbsp( ) {		記述子
842	vps_video_parameter_set_id	u(4)
802	vps_max_layers_minus1	u(6)
846	vps_max_sublayers_minus1	u(3)
	if( vps_max_layers_minus1 > 0 && vps_max_sublayers_minus1 > 0 )	
848	vps_all_layers_same_num_sublayers_flag	u(1)
	if( vps_max_layers_minus1 > 0 )	
850	vps_all_independent_layers_flag	u(1)
	for( i=0; i <= vps_max_layers_minus1; i++ ) {	
852	vps_layer_id[i]	u(6)
	if( i > 0 && !vps_all_independent_layers_flag ) {	
854	vps_independent_layer_flag[i]	u(1)
	if( !vps_independent_layer_flag[i] ) {	
	for( j=0; j < i; j++ )	
	vps_direct_ref_layer_flag[i][j]	u(1)
856	max_tid_ref_present_flag[i]	u(1)
858	if( max_tid_ref_present_flag[i] )	
860	max_tid_il_ref_pics_plus1[i]	u(3)
	}	
	}	
	if( vps_max_layers_minus1 > 0 ) {	
	if( vps_all_independent_layers_flag )	
862	each_layer_is_an_ols_flag	u(1)
	if( each_layer_is_an_ols_flag ) {	
864	ols_mode_idc	u(2)
	if( ols_mode_idc == 2 && vps_max_layers_minus1 > 1 ) {	
866	num_output_layer_sets_minus1	u(8)
	for( i=1; i <= num_output_layer_sets_minus1; i++ )	
	for( j=0; j <= vps_max_layers_minus1; j++ )	
868	ols_output_layer_flag[i][j]	u(1)
	}	
	}	
	}	
	...	
	}	

10

20

【図 2 6】

picture_header_structure() {		記述子
1	ph_gdr_or_irap_pic_flag	u(1)
2	if( ph_gdr_or_irap_pic_flag )	
	ph_gdr_pic_flag	u(1)
	ph_inter_slice_allowed_flag	u(1)
	if( ph_inter_slice_allowed_flag )	
	ph_intra_slice_allowed_flag	u(1)
	ph_non_ref_pic_flag	u(1)
	ph_pic_parameter_set_id	ue(v)
	ph_pic_order_cnt_lsb	u(v)
	if( ph_gdr_or_irap_pic_flag )	
	ph_no_output_of_prior_pics_flag	u(1)
	if( ph_gdr_pic_flag )	
3	ph_recovery_pic_cmt	ue(v)
	...	
	}	

【図 2 7】

access_unit_delimiter_rbsp() {		記述子
10	aud_irap_or_gdr_au_flag	u(1)
12	and_pic_type	u(3)
	rbsp_trailing_bits()	

30

40

50

【図 2 8】

1	picture header structure( ) {	記述子
6	ph_trap(pic_flag)	u(1)
	if(ph_trap(pic_flag))	
7	ph_gdr(pic_flag)	u(1)
	...	
	}	

【図 2 9】

20	slice header( ) {	記述子
	sh_picture_header_in_slice_header_flag	u(1)
	if(sh_picture_header_in_slice_header_flag)	
	picture_header_structure( )	
	...	
	if(nal_unit_type == IDR_W_RADL    nal_unit_type == IDR_N_LP	
	nal_unit_type == CRA_NUT)	
23	sh_no_output_of_prior_pics_flag	u(1)
	}	

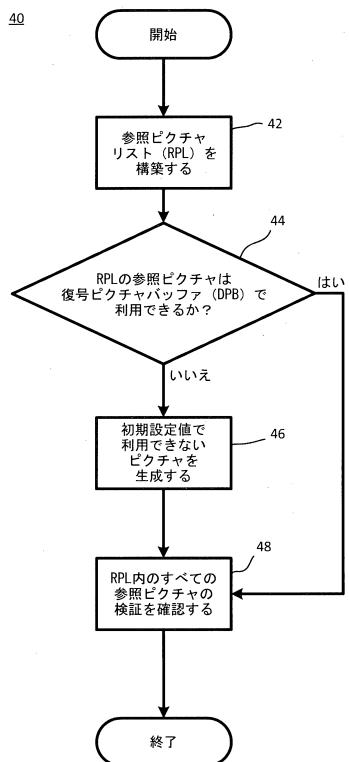
【図 3 0】

10	access_unit_delimiter_rbsp( ) {	記述子
16	aud_irap_au_flag	u(1)
17	aud_gdr_au_flag	u(1)
	aud_pic_type	u(3)
	rbsp_trailing_bits( )	
	}	

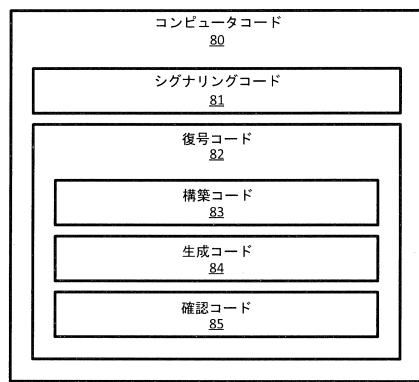
【図 3 1】

1	picture_header_structure( ) {	記述子
2	ph_gdr_or_irap(pic_flag)	u(1)
	ph_non_ref(pic_flag)	u(1)
	if(ph_gdr_or_irap(pic_flag))	
7	ph_gdr(pic_flag)	u(1)
	...	
32	ph_pic_order_cnt_lsb	u(v)
	if(ph_gdr_or_irap(pic_flag))	
	ph_no_output_of_prior_pics_flag	u(1)
	for(i = 0; i < NumExtraBitFields; i++)	
	ph_extra_bit[i]	u(1)
	if(sps_pic_msb_cycle_flag) {	
	ph_pic_msb_cycle_present_flag	u(1)
	if(pic_msb_cycle_present_flag)	
36	ph_pic_msb_cycle_val	u(v)
	}	
	if(ph_gdr(pic_flag))	
34	ph_recovery_pic_cnt	u(v)
	...	
	}	

【図3-2】



【図3-3】



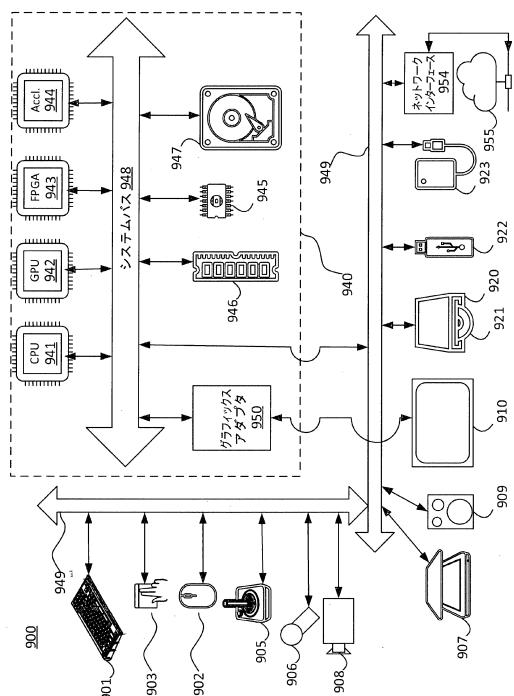
10

20

30

40

【図3-4】



50

---

フロントページの続き

(33) 優先権主張国・地域又は機関

米国(US)

(31) 優先権主張番号 63/036,335

(32) 優先日 令和2年6月8日(2020.6.8)

(33) 優先権主張国・地域又は機関

米国(US)

(31) 優先権主張番号 63/037,903

(32) 優先日 令和2年6月11日(2020.6.11)

(33) 優先権主張国・地域又は機関

米国(US)

(31) 優先権主張番号 17/320,764

(32) 優先日 令和3年5月14日(2021.5.14)

(33) 優先権主張国・地域又は機関

米国(US)

アメリカ合衆国・カリフォルニア・94306・パロ・アルト・パーク・ブルバード・2747  
・テンセント・アメリカ・エルエルシー内

(72) 発明者 シャン・リュウ

アメリカ合衆国・カリフォルニア・94306・パロ・アルト・パーク・ブルバード・2747  
・テンセント・アメリカ・エルエルシー内

審査官 田部井 和彦

(56) 参考文献 國際公開第2020/037272 (WO, A1)

Benjamin Bross et al. , Versatile Video Coding (Draft 9) [online] , JVET-R2001-vA , ITU-T  
インターネット <URL:[https://jvet-experts.org/doc\\_end\\_user/documents/18\\_Teleconference/wg11/JVET-R2001-vA.zip](https://jvet-experts.org/doc_end_user/documents/18_Teleconference/wg11/JVET-R2001-vA.zip)> , 2020年05月15日 , PP.1, 48, 52

Lulin Chen et al. , AHG9: On flag ph\_gdr\_or\_irap\_pic\_flag [online] , JVET-S0070-v1 , ITU-T  
インターネット <URL:[https://jvet-experts.org/doc\\_end\\_user/documents/19\\_Teleconference/wg11/JVET-S0070-v1.zip](https://jvet-experts.org/doc_end_user/documents/19_Teleconference/wg11/JVET-S0070-v1.zip)> , 2020年05月23日 , pp.1-4

Benjamin Bross Jianle Chen Shan Liu Ye-Kui Wang , Versatile Video Coding (Draft 9) [online] , JVET-S JVET-R2001-vA , ITU-T インターネット <URL:[https://jvet-experts.org/doc\\_end\\_user/documents/18\\_Teleconference/wg11/JVET-R2001-vA.zip](https://jvet-experts.org/doc_end_user/documents/18_Teleconference/wg11/JVET-R2001-vA.zip)> , 2020年05月15日 , PP.1, 48, 52

Lulin Chen Chih-Wei Hsu Shih-Ta Hsiang Olena Chubach Yu-Wen Huang Shaw-Min Lei  
No. 1, Dusing 1 SP st /SP Rd., Hsinchu City 30078, Taiwan , AHG9: On flag ph\_gdr\_  
or\_irap\_pic\_flag [online] , JVET-S JVET-S0070-v1 , ITU-T インターネット <URL:[https://jvet-experts.org/doc\\_end\\_user/documents/19\\_Teleconference/wg11/JVET-S0070-v1.zip](https://jvet-experts.org/doc_end_user/documents/19_Teleconference/wg11/JVET-S0070-v1.zip)> , 2020年05月23日 , pp.1-4

(58) 調査した分野 (Int.Cl. , D B名)

H 04 N 19 / 70

H 04 N 19 / 30