

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7675718号
(P7675718)

(45)発行日 令和7年5月13日(2025.5.13)

(24)登録日 令和7年5月1日(2025.5.1)

(51)国際特許分類

F I

H 0 4 N 19/70 (2014.01)

H 0 4 N 19/70

H 0 4 N 19/80 (2014.01)

H 0 4 N 19/80

請求項の数 18 (全51頁)

(21)出願番号	特願2022-535057(P2022-535057)	(73)特許権者	511050697
(86)(22)出願日	令和2年12月18日(2020.12.18)		アリババ グループ ホウルディング リ
(65)公表番号	特表2023-509838(P2023-509838		ミテッド
	A)		英国領ケイマン諸島 グランド ケイマン
(43)公表日	令和5年3月10日(2023.3.10)		ジョージ タウン ピーオーボックス 8
(86)国際出願番号	PCT/US2020/066009		4 7 ワン キャピタル プレイス フォー
(87)国際公開番号	WO2021/133672		ス フロア
(87)国際公開日	令和3年7月1日(2021.7.1)	(74)代理人	100079108
審査請求日	令和5年11月29日(2023.11.29)		弁理士 稲葉 良幸
(31)優先権主張番号	62/954,014	(74)代理人	100109346
(32)優先日	令和1年12月27日(2019.12.27)		弁理士 大貫 敏史
(33)優先権主張国・地域又は機関	米国(US)	(74)代理人	100117189
			弁理士 江口 昭彦
		(74)代理人	100134120
			弁理士 内藤 和彦

最終頁に続く

(54)【発明の名称】 サブピクチャ分割情報をシグナリングするための方法及び機器

(57)【特許請求の範囲】

【請求項 1】

復号器において実施される映像処理方法であって、
ビットストリーム内でシグナリングされるサブピクチャ情報存在フラグに従って、前記ビットストリームがサブピクチャ情報を含むかどうかを判定すること、
前記ビットストリームが前記サブピクチャ情報を含むことに応答して、
ピクチャ内のサブピクチャの数、
ターゲットサブピクチャの幅、高さ、位置及び識別子 (I D) マッピング、
subpic_treated_as_pic_flag、及び
loop_filter_across_subpic_enabled_flag
の少なくとも 1 つを前記ビットストリーム内で復号すること、
前記ターゲットサブピクチャが前記ピクチャ内の最後のサブピクチャである場合、前記ターゲットサブピクチャの前記幅及び前記高さの少なくとも 1 つを復号することをスキップすること
を含む映像処理方法。

【請求項 2】

前記ターゲットサブピクチャの前記幅、前記高さ及び前記位置の少なくとも 1 つを復号することは、前記ピクチャ内の前記サブピクチャの前記数に基づく、請求項 1 に記載の方法。

【請求項 3】

前記ピクチャ内に少なくとも 2 つのサブピクチャがある場合、前記subpic_treated_as_pic_flag、前記loop_filter_across_subpic_enabled_flag、並びに前記ターゲットサブピクチャの前記幅、前記高さ及び前記位置の少なくとも 1 つを復号すること
をさらに含む、

前記ピクチャ内に 1 つのみのサブピクチャがある場合、前記ターゲットサブピクチャの前記幅、前記高さ及び前記位置の前記少なくとも 1 つを復号することは、スキップされ、

前記subpic_treated_as_pic_flagは、符号化レイヤ映像シーケンス (C L V S) 内の各符号化ピクチャのサブピクチャが、インループフィルタリング演算を除外する復号化プロセス内のピクチャとして扱われるかどうかを示し、及び

前記loop_filter_across_subpic_enabled_flagは、前記 C L V S 内の各符号化ピクチャのサブピクチャの境界にわたり、サブピクチャの境界にわたるインループフィルタリング演算が可能であるかどうかを示す、請求項 2 に記載の方法。

【請求項 4】

前記ターゲットサブピクチャの前記幅がシグナリングされない場合、前記ターゲットサブピクチャの前記幅の値を前記ピクチャの幅として決定すること、及び

前記ターゲットサブピクチャの前記高さがシグナリングされない場合、前記ターゲットサブピクチャの前記高さの値を前記ピクチャの高さとして決定すること
をさらに含む、請求項 3 に記載の方法。

【請求項 5】

前記ターゲットサブピクチャの前記幅がシグナリングされない場合、符号化ツリーブロック (C T B) サイズ単位の前記ターゲットサブピクチャの前記幅の値を前記 C T B サイズ単位の前記ピクチャの幅として決定すること、及び

前記ターゲットサブピクチャの前記高さがシグナリングされない場合、前記 C T B サイズ単位の前記ターゲットサブピクチャの前記高さの値を前記 C T B サイズ単位の前記ピクチャの高さとして決定すること

をさらに含む、請求項 4 に記載の方法。

【請求項 6】

前記subpic_treated_as_pic_flagが前記ビットストリーム内でシグナリングされない場合、前記subpic_treated_as_pic_flagが 1 の値を有すると判定すること、及び

前記loop_filter_across_subpic_enabled_flagが前記ビットストリーム内でシグナリングされない場合、前記loop_filter_across_subpic_enabled_flagが 0 の値を有すると判定すること

をさらに含む、請求項 3 に記載の方法。

【請求項 7】

前記ターゲットサブピクチャの前記幅がシグナリングされない場合、符号化ツリーブロック (C T B) サイズ単位の前記ターゲットサブピクチャの前記幅の値を、前記 C T B サイズ単位の前記ピクチャの幅から前記 C T B サイズ単位の前記ターゲットサブピクチャの左上符号化ツリーユニット (C T U) の水平位置を引いたものとして決定するか、又は前記ターゲットサブピクチャの前記幅の前記値を、前記ピクチャの幅から前記ターゲットサブピクチャの左上符号化ツリーユニット (C T U) の水平位置を引いたものとして決定すること、及び

前記ターゲットサブピクチャの前記高さがシグナリングされない場合、前記 C T B サイズ単位の前記ターゲットサブピクチャの前記高さの値を、前記 C T B サイズ単位の前記ピクチャの高さから前記 C T B サイズ単位の前記ターゲットサブピクチャの前記左上 C T U の垂直位置を引いたものとして決定するか、又は前記ターゲットサブピクチャの前記高さの値を、前記ピクチャの高さから前記ターゲットサブピクチャの前記左上 C T U の垂直位置を引いたものとして決定すること

をさらに含む、請求項 1 に記載の方法。

【請求項 8】

前記ターゲットサブピクチャの前記 I D マッピングを復号することは、

10

20

30

40

50

前記ビットストリーム内で第 1 のフラグを復号すること、及び

前記第 1 のフラグが 1 に等しいことに応答して、第 1 のデータ単位又は第 2 のデータ単位内で前記ターゲットサブピクチャの前記 ID マッピングを復号すること
をさらに含み、

0 に等しい前記第 1 のフラグは、前記ターゲットサブピクチャの前記 ID マッピングが前記ビットストリーム内でシグナリングされないことを示す、請求項 1 に記載の方法。

【請求項 9】

前記第 1 のフラグが 1 に等しく、及び前記ターゲットサブピクチャの前記 ID マッピングが前記第 1 のデータ単位内でシグナリングされないことに応答して、前記ターゲットサブピクチャの前記 ID マッピングを前記第 2 のデータ単位内で復号すること、又は

前記第 1 のフラグが 0 に等しいか、又は前記ターゲットサブピクチャの前記 ID マッピングが前記第 1 のデータ単位内でシグナリングされることに応答して、前記ターゲットサブピクチャの前記 ID マッピングを前記第 2 のデータ単位内で復号することをスキップすること

をさらに含む、請求項 8 に記載の方法。

【請求項 10】

前記第 1 のデータ単位及び前記第 2 のデータ単位のそれぞれは、シーケンスパラメータセット (S P S) 、ピクチャパラメータセット (P P S) 又はピクチャヘッダ (P H) の 1 つである、請求項 9 に記載の方法。

【請求項 11】

符号器において実装される映像処理機器であって、
命令を記憶するための少なくとも 1 つのメモリと、
少なくとも 1 つのプロセッサとを含み、前記少なくとも 1 つのプロセッサは、
ビットストリーム内でシグナリングされるサブピクチャ情報存在フラグに従って、前記ビットストリームがサブピクチャ情報を含むかどうかを判定すること、

前記ビットストリームが前記サブピクチャ情報を含むことに応答して、

ピクチャ内のサブピクチャの数、

ターゲットサブピクチャの幅、高さ、位置及び識別子 (I D) マッピング、

subpic_treated_as_pic_flag、及び

loop_filter_across_subpic_enabled_flag

の少なくとも 1 つを前記ビットストリーム内でシグナリングすること、

前記ターゲットサブピクチャが前記ピクチャ内の最後のサブピクチャである場合、前記ターゲットサブピクチャの前記幅及び前記高さの少なくとも 1 つのシグナリングをスキップすること

を前記機器に行わせるために前記命令を実行するように構成される、映像処理機器。

【請求項 12】

前記ターゲットサブピクチャの前記幅、前記高さ及び前記位置の少なくとも 1 つを前記シグナリングすることは、前記ピクチャ内の前記サブピクチャの前記数に基づく、請求項 11 に記載の機器。

【請求項 13】

前記少なくとも 1 つのプロセッサは、

前記ピクチャ内に少なくとも 2 つのサブピクチャがある場合、前記subpic_treated_as_pic_flag、前記loop_filter_across_subpic_enabled_flag、並びに前記ターゲットサブピクチャの前記幅、前記高さ及び前記位置の少なくとも 1 つをシグナリングすること
を前記機器に行わせるために前記命令を実行するように構成され、

前記ピクチャ内に 1 つのみのサブピクチャがある場合、前記ターゲットサブピクチャの前記幅、前記高さ及び前記位置の前記少なくとも 1 つの前記シグナリングは、スキップされ、

前記subpic_treated_as_pic_flagは、符号化レイヤ映像シーケンス (C L V S) 内の各符号化ピクチャのサブピクチャが、インループフィルタリング演算を除外する復号化プ

10

20

30

40

50

ロセス内のピクチャとして扱われるかどうかを示し、及び

前記loop_filter_across_subpic_enabled_flagは、前記C L V S内の各符号化ピクチャのサブピクチャの境界にわたり、サブピクチャの境界にわたるインループフィルタリング演算が可能であるかどうかを示す、請求項12に記載の機器。

【請求項14】

前記少なくとも1つのプロセッサは、

前記ターゲットサブピクチャの前記幅がシグナリングされない場合、前記ターゲットサブピクチャの前記幅の値を前記ピクチャの幅として決定すること、及び

前記ターゲットサブピクチャの前記高さがシグナリングされない場合、前記ターゲットサブピクチャの前記高さを前記ピクチャの高さとして決定すること

10

を前記機器に行わせるために前記命令を実行するように構成される、請求項13に記載の機器。

【請求項15】

前記少なくとも1つのプロセッサは、

前記ターゲットサブピクチャの前記幅がシグナリングされない場合、符号化ツリーブロック(C T B)サイズ単位の前記ターゲットサブピクチャの前記幅の値を前記C T Bサイズ単位の前記ピクチャの幅として決定すること、及び

前記ターゲットサブピクチャの前記高さがシグナリングされない場合、前記C T Bサイズ単位の前記ターゲットサブピクチャの前記高さを前記C T Bサイズ単位の前記ピクチャの高さとして決定すること

20

を前記機器に行わせるために前記命令を実行するように構成される、請求項14に記載の機器。

【請求項16】

前記少なくとも1つのプロセッサは、

前記subpic_treated_as_pic_flagが前記ビットストリーム内でシグナリングされない場合、前記subpic_treated_as_pic_flagが1の値を有すると判定すること、及び

前記loop_filter_across_subpic_enabled_flagが前記ビットストリーム内でシグナリングされない場合、前記loop_filter_across_subpic_enabled_flagが0の値を有すると判定すること

を前記機器に行わせるために前記命令を実行するように構成される、請求項13に記載の機器。

30

【請求項17】

前記少なくとも1つのプロセッサは、

前記ターゲットサブピクチャの前記幅がシグナリングされない場合、符号化ツリーブロック(C T B)サイズ単位の前記ターゲットサブピクチャの前記幅の値を、前記C T Bサイズ単位の前記ピクチャの幅から前記C T Bサイズ単位の前記ターゲットサブピクチャの左上符号化ツリーユニット(C T U)の水平位置を引いたものとして決定するか、又は前記ターゲットサブピクチャの前記幅の前記値を、前記ピクチャの幅から前記ターゲットサブピクチャの左上符号化ツリーユニット(C T U)の水平位置を引いたものとして決定すること、及び

40

前記ターゲットサブピクチャの前記高さがシグナリングされない場合、前記C T Bサイズ単位の前記ターゲットサブピクチャの前記高さを、前記C T Bサイズ単位の前記ピクチャの高さから前記C T Bサイズ単位の前記ターゲットサブピクチャの前記左上C T Uの垂直位置を引いたものとして決定するか、又は前記ターゲットサブピクチャの前記高さを、前記ピクチャの高さから前記ターゲットサブピクチャの前記左上C T Uの垂直位置を引いたものとして決定すること

を前記機器に行わせるために前記命令を実行するように構成される、請求項11に記載の機器。

【請求項18】

映像のビットストリームを記憶するための方法であって、前記方法は、

50

映像シーケンスを受信することと、

前記映像シーケンスの1つ以上のピクチャを符号化することと、

前記符号化することに基づくビットストリームを生成することと、

前記ビットストリームを非一時的コンピュータ可読記憶媒体に記憶することと、

を含み、前記符号化することは、

前記ビットストリーム内でシグナリングされるサブピクチャ情報存在フラグに従って、
前記ビットストリームがサブピクチャ情報を含むかどうかを判定すること、

前記ビットストリームが前記サブピクチャ情報を含むことに応答して、

ピクチャ内のサブピクチャの数、

ターゲットサブピクチャの幅、高さ、位置、識別子 (I D) マッピング、

subpic_treated_as_pic_flag、及び

loop_filter_across_subpic_enabled_flag

の少なくとも1つを前記ビットストリーム内でシグナリングすること、

前記ターゲットサブピクチャが前記ピクチャ内の最後のサブピクチャである場合、前記
ターゲットサブピクチャの前記幅及び前記高さの少なくとも1つのシグナリングをスキッ
プすること

を含む、方法。

【発明の詳細な説明】

【技術分野】

【 0 0 0 1 】

関連出願の相互参照

[0001] 本開示は、参照によりその全体が本明細書に援用される、2019年12月2
7日に出願された米国仮特許出願第62/954,014号に対する優先権を主張する。

【 0 0 0 2 】

技術分野

[0002] 本開示は、概して、映像処理に関し、より詳細には、サブピクチャ分割情報を
シグナリングための方法及び機器に関する。

【背景技術】

【 0 0 0 3 】

背景

[0003] 映像は、視覚情報を取り込んだ静的ピクチャ（又は「フレーム」）のセットで
ある。記憶メモリ及び伝送帯域幅を低減するために、映像を記憶又は伝送前に圧縮し、表
示前に復元することができる。圧縮プロセスは、通常、符号化と称され、復元プロセスは
、通常、復号化と称される。最も一般的には、予測、変換、量子化、エントロピー符号化
及びインループフィルタリングに基づく、標準化映像符号化技術を用いる様々な映像符号
化フォーマットが存在する。特定の映像符号化フォーマットを指定する、高効率ビデオコ
ーディング (H E V C / H . 2 6 5) 規格、多用途ビデオコーディング (V V C / H . 2
6 6) 標準 A V S 規格などの映像符号化規格が標準化機関によって開発されている。進化
した映像符号化技術が映像規格に次々と採用されるに従って、新たな映像符号化規格の符
号化効率が一層高くなる。

【発明の概要】

【課題を解決するための手段】

【 0 0 0 4 】

[0004] いくつかの実施形態では、サブピクチャ分割情報をシグナリングするための例
示的方法は、ビットストリーム内でシグナリングされるサブピクチャ情報存在フラグに従
って、ビットストリームがサブピクチャ情報を含むかどうかを判定すること、及びビット
ストリームがサブピクチャ情報を含むことに応答して、ピクチャ内のサブピクチャの数、
ターゲットサブピクチャの幅、高さ、位置及び識別子 (I D) マッピング、subpic_treat
ed_as_pic_flag、及びloop_filter_across_subpic_enabled_flagの少なくとも1つを
ビットストリーム内でシグナリングすることを含む。

10

20

30

40

50

【 0 0 0 5 】

【0005】いくつかの実施形態では、例示的な映像処理機器は、命令を記憶するための少なくとも1つのメモリと、少なくとも1つのプロセッサとを含む。少なくとも1つのプロセッサは、ビットストリーム内でシグナリングされるサブピクチャ情報存在フラグに従って、ビットストリームがサブピクチャ情報を含むかどうかを判定すること、及びビットストリームがサブピクチャ情報を含むことに応答して、ピクチャ内のサブピクチャの数、ターゲットサブピクチャの幅、高さ、位置及び識別子 (I D) マッピング、subpic_treated_as_pic_flag、及びloop_filter_across_subpic_enabled_flagの少なくとも1つをビットストリーム内でシグナリングすることを機器に行わせるために命令を実行するように構成される。

10

【 0 0 0 6 】

【0006】いくつかの実施形態では、例示的な非一時的コンピュータ可読記憶媒体は、命令のセットを記憶する。命令のセットは、ビットストリーム内でシグナリングされるサブピクチャ情報存在フラグに従って、ビットストリームがサブピクチャ情報を含むかどうかを判定すること、及びビットストリームがサブピクチャ情報を含むことに応答して、ピクチャ内のサブピクチャの数、ターゲットサブピクチャの幅、高さ、位置及び識別子 (I D) マッピング、subpic_treated_as_pic_flag、及びloop_filter_across_subpic_enabled_flagの少なくとも1つをビットストリーム内でシグナリングすることを映像処理機器に行わせるために1つ以上の処理装置によって実行可能である。

20

【 0 0 0 7 】

図面の簡単な説明

【0007】本開示の実施形態及び様々な態様が以下の詳細な説明及び添付の図において例示される。図に示される様々な特徴は、原寸に比例して描かれていない。

【図面の簡単な説明】

【 0 0 0 8 】

【図 1】【0008】本開示のいくつかの実施形態に係る、例示的な映像シーケンスの構造を示す概略図である。

【図 2 A】【0009】本開示の実施形態に従う、ハイブリッド映像符号化システムの例示的な符号化プロセスの概略図を示す。

【図 2 B】【0010】本開示の実施形態に従う、ハイブリッド映像符号化システムの別の例示的な符号化プロセスの概略図を示す。

30

【図 3 A】【0011】本開示の実施形態に従う、ハイブリッド映像符号化システムの例示的な復号化プロセスの概略図を示す。

【図 3 B】【0012】本開示の実施形態に従う、ハイブリッド映像符号化システムの別の例示的な復号化プロセスの概略図を示す。

【図 4】【0013】本開示のいくつかの実施形態に係る、映像を符号化又は復号化するための例示的な機器のブロック図を示す。

【図 5】【0014】本開示のいくつかの実施形態に係る、符号化ツリーユニット (C T U) に分割されるピクチャの一例を示す概略図である。

【図 6】【0015】本開示のいくつかの実施形態に係る、タイル及びラスタスキャンスライスに分割されるピクチャの一例を示す概略図である。

40

【図 7】【0016】本開示のいくつかの実施形態に係る、タイル及び矩形スライスに分割されるピクチャの一例を示す概略図である。

【図 8】【0017】本開示のいくつかの実施形態に係る、タイル及び矩形スライスに分割されるピクチャの別の例を示す概略図である。

【図 9】【0018】本開示のいくつかの実施形態に係る、サブピクチャに分割されるピクチャの一例を示す概略図である。

【図 1 0】【0019】本開示のいくつかの実施形態に係る、サブピクチャ分割の例示的なシーケンスパラメータセット (S P S) 構文を示す例示的な表 1 を示す。

【図 1 1】【0020】本開示のいくつかの実施形態に係る、サブピクチャ識別子の例示的な S

50

P S 構文を示す例示的な表 2 を示す。

【図 1 2】[0021]本開示のいくつかの実施形態に係る、サブピクチャ識別子の例示的なピクチャパラメータセット (P P S) 構文を示す例示的な表 3 を示す。

【図 1 3】[0022]本開示のいくつかの実施形態に係る、サブピクチャ識別子の例示的なピクチャヘッダ (P H) 構文を示す例示的な表 4 を示す。

【図 1 4】[0023]本開示のいくつかの実施形態に係る、例示的なビットストリーム適合制約を示す概略図である。

【図 1 5】[0024]本開示のいくつかの実施形態に係る、サブピクチャ識別子の別の例示的な P H 構文を示す例示的な表 5 を示す。

【図 1 6】[0025]本開示のいくつかの実施形態に係る、サブピクチャ識別子の別の例示的な P H 構文を示す例示的な表 6 を示す。

10

【図 1 7 A】[0026]本開示のいくつかの実施形態に係る、例示的な S P S 構文を示す例示的な表 7 A を示す。

【図 1 7 B】[0027]本開示のいくつかの実施形態に係る、別の例示的な S P S 構文を示す例示的な表 7 B を示す。

【図 1 8】[0028]本開示のいくつかの実施形態に係る、別の例示的な S P S 構文を示す例示的な表 8 を示す。

【図 1 9】[0029]本開示のいくつかの実施形態に係る、別の例示的な S P S 構文を示す例示的な表 9 を示す。

【図 2 0】[0030]本開示のいくつかの実施形態に係る、別の例示的な S P S 構文を示す例示的な表 1 0 を示す。

20

【図 2 1】[0031]本開示のいくつかの実施形態に係る、例示的な映像処理方法のフローチャートを示す。

【図 2 2】[0032]本開示のいくつかの実施形態に係る、別の例示的な映像処理方法のフローチャートを示す。

【図 2 3】[0033]本開示のいくつかの実施形態に係る、別の例示的な映像処理方法のフローチャートを示す。

【図 2 4】[0034]本開示のいくつかの実施形態に係る、別の例示的な映像処理方法のフローチャートを示す。

【図 2 5】[0035]本開示のいくつかの実施形態に係る、別の例示的な映像処理方法のフローチャートを示す。

30

【発明を実施するための形態】

【 0 0 0 9 】

詳細な説明

[0036] ここで、添付の図面に例が示された例示的な実施形態を詳細に参照する。以下の説明は、添付の図面を参照し、図面において、異なる図面における同じ符号は、別途示されない限り、同じ又は同様の要素を表す。例示的な実施形態の以下の説明において示される実装形態は、本発明に従う全ての実装形態を表すものではない。むしろ、それらは、添付の請求項において列挙されるとおりの本発明に関連する態様に従う機器及び方法の単なる例にすぎない。本開示の特定の態様が以下においてより詳細に説明される。参照により組み込まれる用語及び / 又は定義と矛盾する場合、本明細書において提供される用語及び定義が優先する。

40

【 0 0 1 0 】

[0037] I T U - T ビデオコーディングエキスパートグループ (I T U - T V C E G) 及び I S O / I E C ムービングピクチャエキスパートグループ (I S O / I E C M P E G) のジョイントビデオエキスパートチーム (J V E T) は、現在、多用途ビデオコーディング (V V C / H . 2 6 6) 規格を開発している。V V C 規格は、その前身、高効率ビデオコーディング (H E V C / H . 2 6 5) 規格の圧縮効率を 2 倍にすることを目指している。換言すれば、V V C の目標は、半分の帯域幅を用いて H E V C / H . 2 6 5 と同じ主観的品質を達成することである。

50

【 0 0 1 1 】

[0038] 半分の帯域幅を用いて H E V C / H . 2 6 5 と同じ主観的品質を達成するために、J V E T は、共同探索モデル (J E M) 参照ソフトウェアを用いて H E V C を超える技術を開発している。符号化技術が J E M に組み込まれたため、J E M は H E V C よりも実質的に高い符号化性能を達成した。

【 0 0 1 2 】

[0039] V V C 規格は最近開発されたものであり、より優れた圧縮性能をもたらすより多くの符号化技術を組み込み続けている。V V C は、H E V C、H . 2 6 4 / A V C、M P E G 2、H . 2 6 3 等などの現代的な映像圧縮規格において用いられてきた同じハイブリッド映像符号化システムに基づく。

10

【 0 0 1 3 】

[0040] 映像は、視覚情報を記憶するために時系列で配列された静的ピクチャ (又は「フレーム」) のセットである。映像取り込みデバイス (例えば、カメラ) を、それらのピクチャを時系列で取り込んで記憶するために用いることができ、映像再生デバイス (例えば、テレビ、コンピュータ、スマートフォン、タブレットコンピュータ、ビデオプレーヤ又は表示機能を有する任意のエンドユーザ端末) を、このようなピクチャを時系列で表示するために用いることができる。また、用途によっては、映像取り込みデバイスが、取り込まれた映像を、監督、会議開催又は生放送などのために、映像再生デバイス (例えば、モニタを有するコンピュータ) へリアルタイムに伝送することができる。

【 0 0 1 4 】

20

[0041] このような用途によって必要とされる記憶空間及び伝送帯域幅を低減するために、映像を記憶及び伝送前に圧縮し、表示前に復元することができる。圧縮及び復元は、プロセッサ (例えば、汎用コンピュータのプロセッサ) によって実行されるソフトウェア又は特殊ハードウェアによって実施され得る。圧縮のためのモジュールは一般的に「符号器」と称され、復元のためのモジュールは一般的に「復号器」と称される。符号器及び復号器はまとめて「コーデック」と称され得る。符号器及び復号器は、種々の好適なハードウェア、ソフトウェア又はこれらの組み合わせの任意のものとして実施することができる。例えば、符号器及び復号器のハードウェア実装形態は、1 つ以上のマイクロプロセッサ、デジタル信号プロセッサ (D S P)、特定用途向け集積回路 (A S I C)、フィールドプログラマブルゲートアレイ (F P G A)、個別論理又はこれらの任意の組み合わせなどの回路機構を含むことができる。符号器及び復号器のソフトウェア実装形態は、プログラムコード、コンピュータ実行可能命令、ファームウェア又はコンピュータ可読媒体内に固定された任意の好適なコンピュータ実施アルゴリズム若しくはプロセスを含むことができる。映像圧縮及び復元は、M P E G - 1、M P E G - 2、M P E G - 4、H . 2 6 x シリーズ又は同様のものなど、様々なアルゴリズム又は規格によって実施され得る。用途によっては、コーデックは映像を第 1 の符号化規格から復元し、復元された映像を、第 2 の符号化規格を用いて再圧縮することができる。この場合、コーデックは、「トランスコーダ」と称され得る。

30

【 0 0 1 5 】

[0042] 映像符号化プロセスは、ピクチャを再構成するために用いることができる有用な情報を識別して維持し、再構成のために重要でない情報を無視することができる。無視された、重要でない情報を完全に再構成することができない場合、このような符号化プロセスは、「非可逆」と称され得る。さもなければ、それは「可逆」と称され得る。大抵の符号化プロセスは非可逆であり、これは、必要とされる記憶空間及び伝送帯域幅を低減するためのトレードオフである。

40

【 0 0 1 6 】

[0043] 符号化中のピクチャ (「現在のピクチャ」と称される) の有用な情報は、参照ピクチャ (例えば、以前に符号化され、再構成されたピクチャ) に対する変化を含む。このような変化は、ピクセルの位置の変化、明るさの変化、又は色の変化を含むことができ、これらの中でも、位置の変化が最も重要である。オブジェクトを表現するピクセルのグ

50

ループの位置の変化は、参照ピクチャと現在のピクチャとの間のオブジェクトの動きを反映することができる。

【 0 0 1 7 】

[0044] 別のピクチャを参照することなく符号化されたピクチャ（すなわち、それがそれ自身の参照ピクチャである）は「Iピクチャ」と称される。以前のピクチャを参照ピクチャとして用いて符号化されたピクチャは「Pピクチャ」と称される。以前のピクチャ及び将来のピクチャの両方を参照ピクチャとして用いて符号化されたピクチャ（すなわち、参照は「双方向性」である）は「Bピクチャ」と称される。

【 0 0 1 8 】

[0045] 図1は、本開示のいくつかの実施形態に従う、例示的な映像シーケンス100の構造を示す。映像シーケンス100は、ライブ映像又は取り込まれ、アーカイブされた映像であり得る。映像100は、現実の映像、コンピュータ生成映像（例えば、コンピュータゲーム映像）、又はこれらの組み合わせ（例えば、拡張現実感効果を伴う現実の映像）であり得る。映像シーケンス100は、映像取り込みデバイス（例えば、カメラ）、以前に取り込まれた映像を包含する映像アーカイブ（例えば、記憶デバイス内に記憶された映像ファイル）、又は映像コンテンツプロバイダからの映像を受信するための映像供給インターフェース（例えば、映像放送トランシーバ）から入力され得る。

【 0 0 1 9 】

[0046] 図1に示されるように、映像シーケンス100は、ピクチャ102、104、106及び108を含む、タイムラインに沿って時間的に配列された一連のピクチャを含むことができる。ピクチャ102～106は連続しており、ピクチャ106及び108間にさらなるピクチャが存在する。図1において、ピクチャ102はIピクチャであり、その参照ピクチャはピクチャ102自身である。ピクチャ104はPピクチャであり、その参照ピクチャは、矢印によって指示されるように、ピクチャ102である。ピクチャ106はBピクチャであり、その参照ピクチャは、矢印によって指示されるように、ピクチャ104及び108である。いくつかの実施形態では、ピクチャ（例えば、ピクチャ104）の参照ピクチャは、そのピクチャの直前又は直後になくてもよい。例えば、ピクチャ104の参照ピクチャは、ピクチャ102の前のピクチャであり得る。ピクチャ102～106の参照ピクチャは単なる例にすぎず、本開示は参照ピクチャの実施形態を、図1に示される例として限定しないことに留意されたい。

【 0 0 2 0 】

[0047] 典型的に、映像コーデックは、このようなタスクの計算の複雑性のため、ピクチャ全体を一度に符号化又は復号化しない。むしろ、それらはピクチャを基本セグメントに分割し、ピクチャをセグメントごとに符号化又は復号化することができる。このような基本セグメントは本開示において基本処理ユニット（「BPU」）と称される。例えば、図1における構造110は、映像シーケンス100のピクチャ（例えば、ピクチャ102～108の任意のもの）の例示的な構造を示す。構造110では、ピクチャが4×4基本処理ユニットに分割され、それらの境界は破線として示されている。いくつかの実施形態では、基本処理ユニットは、いくつかの映像符号化規格（例えば、MPEGファミリー、H.261、H.263若しくはH.264/AVC）では「マクロブロック」と、又はいくつかの他の映像符号化規格（例えば、H.265/HEVC若しくはH.266/VVC）では「符号化ツリーユニット」（「CTU」）と称され得る。基本処理ユニットは、128×128、64×64、32×32、16×16、4×8、16×32など、ピクチャにおける可変サイズ、又はピクセルの任意の形状及びサイズを有することができる。基本処理ユニットのサイズ及び形状は、ピクチャのために、符号化効率と、基本処理ユニットにおいて維持されるべき詳細さのレベルとのバランスに基づいて選択することができる。

【 0 0 2 1 】

[0048] 基本処理ユニットは、コンピュータメモリ内に（例えば、映像フレームバッファ内に）記憶された異なる種類の映像データのグループを含むことができる、論理ユニッ

10

20

30

40

50

トであり得る。例えば、カラーピクチャの基本処理ユニットは、無色の輝度情報を表現するルマ成分（Ｙ）、色情報を表現する１つ以上のクロマ成分（例えば、Ｃｂ及びＣｒ）、並びに関連構文要素を含むことができ、ここで、ルマ及びクロマ成分は基本処理ユニットの同じサイズを有することができる。ルマ及びクロマ成分は、いくつかの映像符号化規格（例えば、Ｈ．２６５／ＨＥＶＣ又はＨ．２６６／ＶＶＣ）では、「符号化ツリーブロック」（「ＣＴＢ」）と称され得る。基本処理ユニットに対して遂行される任意の演算は、そのルマ及びクロマ成分の各々に対して繰り返し遂行され得る。

【００２２】

【００４９】映像符号化は複数の演算段階を有し、図２Ａ～図２Ｂ及び図３Ａ～図３Ｂにその例が示されている。段階ごとに、基本処理ユニットのサイズは依然として処理のために大きすぎるものになり得、そのため、本開示において「基本処理サブユニット」と称されるセグメントにさらに分割され得る。いくつかの実施形態では、基本処理サブユニットは、いくつかの映像符号化規格（例えば、ＭＰＥＧファミリー、Ｈ．２６１、Ｈ．２６３若しくはＨ．２６４／ＡＶＣ）では「ブロック」と、又はいくつかの他の映像符号化規格（例えば、Ｈ．２６５／ＨＥＶＣ若しくはＨ．２６６／ＶＶＣ）では「符号化ユニット」（「ＣＵ」）と称され得る。基本処理サブユニットは、基本処理ユニットと同じであるか又はそれよりも小さいサイズを有することができる。基本処理ユニットと同様に、基本処理サブユニットも、コンピュータメモリ内に（例えば、映像フレームバッファ内に）記憶された異なる種類の映像データ（例えば、Ｙ、Ｃｂ、Ｃｒ及び関連構文要素）のグループを含むことができる、論理ユニットである。基本処理サブユニットに対して遂行される任意の動作は、そのルマ及びクロマ成分の各々に対して繰り返し遂行され得る。このような分割は処理の必要に応じてさらなるレベルまで遂行され得ることに留意されたい。また、異なる段階は、異なる方式を用いて基本処理ユニットを分割することができることに留意されたい。

【００２３】

【００５０】例えば、モード決定段階（図２Ｂにその一例が示されている）において、符号器は、どのような予測モード（例えば、イントラピクチャ予測又はインターピクチャ予測）を基本処理ユニットのために用いるかを決定することができるが、基本処理ユニットは、このような決定を行うには大きすぎるものになり得る。符号器は、基本処理ユニットを複数の基本処理サブユニット（例えば、Ｈ．２６５／ＨＥＶＣ又はＨ．２６６／ＶＶＣの場合のように、ＣＵ）に分割し、個々の基本処理サブユニットごとに予測の種類を決めることができる。

【００２４】

【００５１】別の例として、予測段階（図２Ａ～図２Ｂにその例が示されている）において、符号器は、基本処理サブユニット（例えば、ＣＵ）のレベルで予測演算を遂行することができる。しかし、場合により、基本処理サブユニットは、依然として、処理するには大きすぎるものになり得る。符号器は、基本処理サブユニットをより小さいセグメント（例えば、Ｈ．２６５／ＨＥＶＣ又はＨ．２６６／ＶＶＣでは「予測ブロック」又は「ＰＢ」と称される）にさらに分割することができ、そのレベルで予測演算が遂行され得る。

【００２５】

【００５２】別の例として、変換段階（図２Ａ～図２Ｂにその例が示されている）において、符号器は、残差基本処理サブユニット（例えば、ＣＵ）のための変換演算を遂行することができる。しかし、場合により、基本処理サブユニットは、依然として、処理するには大きすぎるものになり得る。符号器は、基本処理サブユニットをより小さいセグメント（例えば、Ｈ．２６５／ＨＥＶＣ又はＨ．２６６／ＶＶＣでは「変換ブロック」又は「ＴＢ」と称される）にさらに分割することができ、そのレベルで変換演算が遂行され得る。同じ基本処理サブユニットの分割方式は、予測段階及び変換段階において異なり得ることに留意されたい。例えば、Ｈ．２６５／ＨＥＶＣ又はＨ．２６６／ＶＶＣでは、同じＣＵの予測ブロック及び変換ブロックが異なるサイズ及び数を有し得る。

【００２６】

10

20

30

40

50

[0053] 図1の構造110では、基本処理ユニット112は、 3×3 基本処理サブユニットにさらに分割され、それらの境界は、点線として示されている。同じピクチャの異なる基本処理ユニットは、異なる方式で基本処理サブユニットに分割され得る。

【0027】

[0054] 実装形態によっては、並列処理及び誤り耐性の能力を映像符号化及び復号化にもたすために、ピクチャを処理のための領域に分割することができ、これにより、符号化又は復号化プロセスは、ピクチャの領域に関して、ピクチャのいかなる他の領域からの情報にも依存しなくてすむ。換言すれば、ピクチャの各領域は独立して処理され得る。そうすることにより、コーデックはピクチャの異なる領域を並行して処理することができ、そのため、符号化効率を増大させる。また、領域のデータが処理中に破損したか、又はネットワーク伝送中に失われたときには、コーデックは、破損した又は失われたデータを頼ることなく、同じピクチャの他の領域を正しく符号化又は復号化することもでき、そのため、誤り耐性の能力をもたす。いくつかの映像符号化規格では、ピクチャを異なる種類の領域に分割することができる。例えば、H.265/HEVC及びH.266/VVCは2種類の領域：「スライス」及び「タイル」を提供する。また、映像シーケンス100の異なるピクチャは、ピクチャを領域に分割するための異なる区分方式を有することができることに留意されたい。

【0028】

[0055] 例えば、図1では、構造110は、3つの領域114、116、及び118に分割され、それらの境界は構造110の内部の実線として示されている。領域114は4つの基本処理ユニットを含む。領域116及び118の各々は6つの基本処理ユニットを含む。図1における構造110の基本処理ユニット、基本処理サブユニット、及び領域は単なる例にすぎず、本開示は、その実施形態を限定しないことに留意されたい。

【0029】

[0056] 図2Aは、本開示の実施形態に従う、例示的な符号化プロセス200Aの概略図を示す。例えば、符号化プロセス200Aは符号器によって遂行され得る。図2Aに示されるように、符号器は、プロセス200Aに従って映像シーケンス202を映像ビットストリーム228に符号化することができる。図1における映像シーケンス100と同様に、映像シーケンス202は、時間的順序で配列されたピクチャ（「原ピクチャ」と称される）のセットを含むことができる。図1における構造110と同様に、映像シーケンス202の各原ピクチャは、符号器によって処理のために基本処理ユニット、基本処理サブユニット、又は領域に分割され得る。いくつかの実施形態では、符号器は、映像シーケンス202の原ピクチャごとに基本処理ユニットのレベルでプロセス200Aを遂行することができる。例えば、符号器はプロセス200Aを反復的な仕方ですべて遂行することができ、その場合、符号器は基本処理ユニットをプロセス200Aの1回の反復において符号化することができる。いくつかの実施形態では、符号器は、プロセス200Aを映像シーケンス202の各原ピクチャの領域（例えば、領域114～118）のために並行して遂行することができる。

【0030】

[0057] 図2Aにおいて、符号器は、映像シーケンス202の原ピクチャの基本処理ユニット（「原BPU」と称される）を予測段階204に供給し、予測データ206及び予測BPU208を生成することができる。符号器は、予測BPU208を原BPUから減算し、残差BPU210を生成することができる。符号器は、残差BPU210を変換段階212及び量子化段階214に供給し、量子化変換係数216を生成することができる。符号器は、予測データ206及び量子化変換係数216を2値符号化段階226に供給し、映像ビットストリーム228を生成することができる。構成要素202、204、206、208、210、212、214、216、226、及び228は「順方向経路」と称され得る。プロセス200Aの間、量子化段階214の後に、符号器は、量子化変換係数216を逆量子化段階218及び逆変換段階220に供給し、再構成残差BPU222を生成することができる。符号器は、再構成残差BPU222を予測BPU208に加

算し、プロセス 200A の次の反復のために予測段階 204 において用いられる、予測基準 224 を生成することができる。プロセス 200A の構成要素 218、220、222、及び 224 は「再構成経路」と称され得る。再構成経路は、符号器及び復号器の両方が同じ参照データを予測のために用いることを確実にするために用いられ得る。

【0031】

[0058] 符号器は、原ピクチャの各原 BPU を（順方向経路内で）符号化し、原ピクチャの次の原 BPU を符号化するための予測基準 224 を（再構成経路内で）生成するために、プロセス 200A を反復的に遂行することができる。原ピクチャの全ての原 BPU を符号化した後、符号器は、映像シーケンス 202 内の次のピクチャを符号化するために進むことができる。

10

【0032】

[0059] プロセス 200A を参照すると、符号器は、映像取り込みデバイス（例えば、カメラ）によって生成された映像シーケンス 202 を受信することができる。本明細書において用いられる用語「受信する」は、受信すること、入力すること、獲得すること、取得すること、得ること、読み込むこと、アクセスすること、又はデータを入力するための任意の仕方による任意の行為を指すことができる。

【0033】

[0060] 予測段階 204 において、現在の反復において、符号器は原 BPU 及び予測基準 224 を受信し、予測演算を遂行し、予測データ 206 及び予測 BPU 208 を生成することができる。予測基準 224 は、プロセス 200A の以前の反復の再構成経路から生成され得る。予測段階 204 の目的は、予測データ 206 を抽出することにより、情報冗長性を低減することであり、予測データ 206 は、予測データ 206 及び予測基準 224 から原 BPU を予測 BPU 208 として再構成するために用いることができる。

20

【0034】

[0061] 理想的には、予測 BPU 208 は、原 BPU と同一であり得る。しかし、非理想的な予測及び再構成演算のため、予測 BPU 208 は、概して、原 BPU とは若干異なる。このような差を記録するために、予測 BPU 208 を生成した後、符号器は、それを原 BPU から減算し、残差 BPU 210 を生成することができる。例えば、符号器は、予測 BPU 208 のピクセルの値（例えば、グレースケール値又は RGB 値）を原 BPU の対応するピクセルの値から減算することができる。残差 BPU 210 の各ピクセルは、原 BPU 及び予測 BPU 208 の対応するピクセル間のこのような減算の結果としての残差値を有することができる。原 BPU と比べて、予測データ 206 及び残差 BPU 210 はより少数のビットを有することができるが、それらは、著しい品質劣化を伴うことなく原 BPU を再構成するために用いられ得る。そのため、原 BPU は、圧縮される。

30

【0035】

[0062] 残差 BPU 210 をさらに圧縮するために、変換段階 212 において、符号器は、それを 2 次元「基底パターン」のセットに分解することにより、残差 BPU 210 の空間的冗長性を低減することができ、各基底パターンは「変換係数」に関連付けられている。基底パターンは同じサイズ（例えば、残差 BPU 210 のサイズ）を有することができる。各基底パターンは残差 BPU 210 の変化周波数（例えば、輝度変化の周波数）成分を表現することができる。基底パターンはいずれも、いかなる他の基底パターンのいかなる結合（例えば、線形結合）からも再現することができない。換言すれば、分解は残差 BPU 210 の変化を周波数領域に分解することができる。このような分解は関数の離散フーリエ変換と類似しており、この場合、基底パターンは離散フーリエ変換の基底関数（例えば、三角関数）と類似しており、変換係数は、基底関数に関連付けられた係数と類似している。

40

【0036】

[0063] 異なる変換アルゴリズムは、異なる基底パターンを用いることができる。例えば、離散余弦変換、離散正弦変換又は同様のものなど、様々な変換アルゴリズムを変換段階 212 において用いることができる。変換段階 212 における変換は逆演算可能である

50

。すなわち、符号器は、変換の逆演算（「逆変換」と称される）によって残差 B P U 2 1 0 を回復することができる。例えば、残差 B P U 2 1 0 のピクセルを回復するために、逆変換は、基底パターンの対応するピクセルの値にそれぞれの関連係数を乗算し、積を加算していき、加重和を生成することができる。映像符号化規格のために、符号器及び復号器は両方とも同じ変換アルゴリズム（従って同じ基底パターン）を用いることができる。そのため、符号器は変換係数のみを記録することができ、復号器は、基底パターンを符号器から受信することなく、変換係数から残差 B P U 2 1 0 を再構成することができる。残差 B P U 2 1 0 と比べて、変換係数はより少数のビットを有することができるが、それらは、著しい品質劣化を伴うことなく残差 B P U 2 1 0 を再構成するために用いられ得る。そのため、残差 B P U 2 1 0 は、さらに圧縮される。

10

【 0 0 3 7 】

[0064] 符号器は、量子化段階 2 1 4 において変換係数をさらに圧縮することができる。変換プロセスにおいて、異なる基底パターンは異なる変化周波数（例えば、輝度変化周波数）を表現することができる。人間の眼は、概して、低周波数変化を認識することがより得意であるため、符号器は、復号化において著しい品質劣化を生じさせることなく高周波数変化の情報を無視することができる。例えば、量子化段階 2 1 4 において、符号器は、各変換係数を整数値（「量子化パラメータ」と称される）で除算し、商をその最近傍の整数に丸めることにより、量子化変換係数 2 1 6 を生成することができる。このような演算後、高周波数基底パターンの一部の変換係数は、0 に変換され得、低周波数基底パターンの変換係数はより小さい整数に変換され得る。符号器は 0 値の量子化変換係数 2 1 6 を無視することができ、これによって変換係数は、さらに圧縮される。量子化プロセスも逆演算可能であり、この場合、量子化変換係数 2 1 6 は量子化の逆演算（「逆量子化」と称される）において変換係数に再構成され得る。

20

【 0 0 3 8 】

[0065] 符号器はこのような除算の剰余を丸め演算において無視するため、量子化段階 2 1 4 は非可逆になり得る。典型的に、量子化段階 2 1 4 はプロセス 2 0 0 A において最大の情報損失に寄与し得る。情報損失が大きいほど、量子化変換係数 2 1 6 に必要なビットは少なくなる。異なる情報損失レベルを得るために、符号器は、量子化パラメータ又は量子化プロセスの任意の他のパラメータの異なる値を用いることができる。

【 0 0 3 9 】

30

[0066] 2 値符号化段階 2 2 6 において、符号器は、例えば、エントロピー符号化、可変長符号化、算術符号化、ハフマン符号化、コンテキスト適応 2 値算術符号化、又は任意の他の可逆若しくは非可逆圧縮アルゴリズムなどの 2 値符号化技法を用いて、予測データ 2 0 6 及び量子化変換係数 2 1 6 を符号化することができる。いくつかの実施形態では、予測データ 2 0 6 及び量子化変換係数 2 1 6 のほかに、符号器は、例えば、予測段階 2 0 4 において用いられる予測モード、予測演算のパラメータ、変換段階 2 1 2 における変換の種類、量子化プロセスのパラメータ（例えば、量子化パラメータ）、符号器制御パラメータ（例えば、ビットレート制御パラメータ）、又は同様のものなど、他の情報を 2 値符号化段階 2 2 6 において符号化することができる。符号器は、2 値符号化段階 2 2 6 の出力データを用いて映像ビットストリーム 2 2 8 を生成することができる。いくつかの実施形態では、映像ビットストリーム 2 2 8 をネットワーク伝送のためにさらにパケット化することができる。

40

【 0 0 4 0 】

[0067] プロセス 2 0 0 A の再構成経路を参照すると、逆量子化段階 2 1 8 において、符号器は、量子化変換係数 2 1 6 に対して逆量子化を遂行し、再構成変換係数を生成することができる。逆変換段階 2 2 0 において、符号器は、再構成変換係数に基づいて再構成残差 B P U 2 2 2 を生成することができる。符号器は、再構成残差 B P U 2 2 2 を予測 B P U 2 0 8 に加算し、プロセス 2 0 0 A の次の反復において用いられることになる予測基準 2 2 4 を生成することができる。

【 0 0 4 1 】

50

【0068】 プロセス200Aの他の変形を、映像シーケンス202を符号化するために用いることもできることに留意されたい。いくつかの実施形態では、プロセス200Aの段階は符号器により、異なる順序で遂行され得る。いくつかの実施形態では、プロセス200Aの1つ以上の段階を単一の段階に組み合わせることができる。いくつかの実施形態では、プロセス200Aの単一の段階を複数の段階に分割することができる。例えば、変換段階212及び量子化段階214を単一の段階に組み合わせることができる。いくつかの実施形態では、プロセス200Aは追加の段階を含むことができる。いくつかの実施形態では、プロセス200Aは図2Aにおける1つ以上の段階を省略することができる。

【0042】

【0069】 図2Bは、本開示の実施形態に従う、別の例示的な符号化プロセス200Bの概略図を示す。プロセス200Bはプロセス200Aから変更され得る。例えば、プロセス200Bは、ハイブリッド映像符号化規格（例えば、H.26xシリーズ）に準拠した符号器によって用いられ得る。プロセス200Aと比べて、プロセス200Bの順方向経路はモード決定段階230を追加的に含み、予測段階204を空間的予測段階2042及び時間的予測段階2044に分割する。プロセス200Bの再構成経路はループフィルタ段階232及びバッファ234を追加的に含む。

【0043】

【0070】 概して、予測技法は2つの種類：空間的予測及び時間的予測に分類することができる。空間的予測（例えば、イントラピクチャ予測又は「イントラ予測」）は、現在のBPUを予測するために、同じピクチャ内の1つ以上のすでに符号化された隣接BPUからのピクセルを用いることができる。すなわち、空間的予測における予測基準224は隣接BPUを含むことができる。空間的予測はピクチャの固有の空間的冗長性を低減することができる。時間的予測（例えば、インターピクチャ予測又は「インター予測」）は、現在のBPUを予測するために、1つ以上のすでに符号化されたピクチャからの領域を用いることができる。すなわち、時間的予測における予測基準224は符号化ピクチャを含むことができる。時間的予測はピクチャの固有の時間的冗長性を低減することができる。

【0044】

【0071】 プロセス200Bを参照すると、順方向経路内において、符号器は、予測演算を空間的予測段階2042及び時間的予測段階2044において遂行する。例えば、空間的予測段階2042において、符号器はイントラ予測を遂行することができる。符号化中のピクチャの原BPUのために、予測基準224は、（順方向経路内で）符号化され、（再構成経路内で）再構成された1つ以上の隣接BPUを同じピクチャ内に含むことができる。符号器は、隣接BPUを外挿することによって予測BPU208を生成することができる。外挿技法は、例えば、線形外挿若しくは補間、多項式外挿若しくは補間、又は同様のものを含むことができる。いくつかの実施形態では、符号器は、予測BPU208のピクセルごとに、対応するピクセルの値を外挿することによるなどして、外挿をピクセルレベルで遂行することができる。外挿のために用いられる隣接BPUは、（例えば、原BPUの上の）鉛直方向、（例えば、原BPUの左の）水平方向、（例えば、原BPUの左下、右下、左上若しくは右上の）対角方向、又は用いられる映像符号化規格において定義される任意の方向など、様々な方向から原BPUに対して位置することができる。イントラ予測のために、予測データ206は、例えば、用いられる隣接BPUの場所（例えば、座標）、用いられる隣接BPUのサイズ、外挿のパラメータ、原BPUに対する用いられる隣接BPUの方向、又は同様のものを含むことができる。

【0045】

【0072】 別の例として、時間的予測段階2044において、符号器はインター予測を遂行することができる。現在のピクチャの原BPUのために、予測基準224は、（順方向経路内で）符号化され、（再構成経路内で）再構成された1つ以上のピクチャ（「参照ピクチャ」と称される）を含むことができる。いくつかの実施形態では、参照ピクチャはBPUごとに符号化され、再構成され得る。例えば、符号器は、再構成残差BPU222を予測BPU208に加算し、再構成BPUを生成することができる。同じピクチャの全て

10

20

30

40

50

の再構成 B P U が生成されたとき、符号器は再構成ピクチャを参照ピクチャとして生成することができる。符号器は、参照ピクチャの範囲（「探索窓」と称される）内のマッチング領域を探索するために「動き推定」の演算を遂行することができる。参照ピクチャ内の探索窓の場所は、現在のピクチャの原 B P U の場所に基づいて決定することができる。例えば、探索窓は、参照ピクチャ内の、現在のピクチャ内の原 B P U と同じ座標を有する場所に中心を有することができる、所定の距離にわたって外へ拡張され得る。符号器が（例えば、画素再帰アルゴリズム、ブロックマッチングアルゴリズム、又は同様のものを用いることによって）探索窓内の原 B P U と同様の領域を識別したとき、符号器はこのような領域をマッチング領域と決定することができる。マッチング領域は、原 B P U とは異なる（例えば、原 B P U よりも小さい、それに等しい、それよりも大きい、又は異なる形状の）寸法を有することができる。参照ピクチャ及び現在のピクチャが（例えば、図 1 に示されるように）タイムライン内で時間的に分離されているため、時間が経過するにつれてマッチング領域は原 B P U の場所へ「移動する」と見なすことができる。符号器はこのような動きの方向及び距離を「動きベクトル」として記録することができる。複数の参照ピクチャが（例えば、図 1 におけるピクチャ 106 として）用いられるときには、符号器は、参照ピクチャごとにマッチング領域を探索し、その関連動きベクトルを決定することができる。いくつかの実施形態では、符号器は、それぞれのマッチング参照ピクチャのマッチング領域のピクセル値に重みを付与することができる。

10

【0046】

[0073] 動き推定は、例えば、並進、回転、ズーミング又は同様のものなど、様々な種類の動きを識別するために用いることができる。インター予測のために、予測データ 206 は、例えば、マッチング領域の場所（例えば、座標）、マッチング領域に関連付けられた動きベクトル、参照ピクチャの数、参照ピクチャに関連付けられた重み、又は同様のものを含むことができる。

20

【0047】

[0074] 予測 B P U 208 を生成するために、符号器は「動き補償」の演算を遂行することができる。動き補償は、予測データ 206（例えば、動きベクトル）及び予測基準 224 に基づいて予測 B P U 208 を再構成するために用いることができる。例えば、符号器は、動きベクトルに従って参照ピクチャのマッチング領域を移動させることができ、その場合、符号器は現在のピクチャの原 B P U を予測することができる。（例えば、図 1 におけるピクチャ 106 のように）複数の参照ピクチャが用いられるときには、符号器は、それぞれの動きベクトルに従って参照ピクチャのマッチング領域を移動させ、マッチング領域のピクセル値を平均することができる。いくつかの実施形態では、符号器がそれぞれのマッチング参照ピクチャのマッチング領域のピクセル値に重みを付与した場合、符号器は、ピクセル値の加重和を、移動されたマッチング領域に加算することができる。

30

【0048】

[0075] いくつかの実施形態では、インター予測は、一方向性又は双方向性であり得る。一方向性インター予測は、現在のピクチャに対して同じ時間方向の 1 つ以上の参照ピクチャを用いることができる。例えば、図 1 におけるピクチャ 104 は、参照ピクチャ（すなわちピクチャ 102）がピクチャ 104 に先行する、一方向インター予測ピクチャである。双方向インター予測は、現在のピクチャに対して両方の時間方向にある 1 つ以上の参照ピクチャを用いることができる。例えば、図 1 におけるピクチャ 106 は、参照ピクチャ（すなわち、ピクチャ 104 及び 108）がピクチャ 104 に対して両方の時間方向にある、双方向インター予測ピクチャである。

40

【0049】

[0076] プロセス 200B の順方向経路をなおも参照すると、空間的予測 2042 及び時間的予測段階 2044 の後に、モード決定段階 230 において、符号器は、予測モード（例えば、イントラ予測又はインター予測の一方）をプロセス 200B の現在の反復のために選択することができる。例えば、符号器は、レート - 歪み最適化技法を遂行することができる。本技法では、符号器は、候補予測モードのビットレート及びかかる候補予測モ

50

ード下での再構成参照ピクチャの歪みに依存するコスト関数の値を最小化するための予測モードを選択することができる。選択された予測モードに応じて、符号器は、対応する予測 B P U 2 0 8 及び予測データ 2 0 6 を生成することができる。

【 0 0 5 0 】

[0077] プロセス 2 0 0 B の再構成経路内において、イントラ予測モードが順方向経路内で選択された場合、予測基準 2 2 4 (例えば、現在のピクチャにおいて符号化され、再構成された現在の B P U) を生成した後、符号器は、予測基準 2 2 4 を後の使用のために (例えば、現在のピクチャの次の B P U の外挿のために) 空間的予測段階 2 0 4 2 に直接供給することができる。インター予測モードが順方向経路内で選択された場合、予測基準 2 2 4 (例えば、全ての B P U が符号化され、再構成された現在のピクチャ) を生成した後、符号器は、予測基準 2 2 4 をループフィルタ段階 2 3 2 に供給することができ、そこで、符号器は、ループフィルタを予測基準 2 2 4 に適用し、インター予測によって導入された歪み (例えば、ブロッキングアーチファクト) を低減又は解消することができる。符号器は、例えば、デブロッキング、サンプル適応オフセット、適応ループフィルタ、又は同様のものなど、様々なループフィルタ技法をループフィルタ段階 2 3 2 において適用することができる。ループフィルタリングされた参照ピクチャは、後の使用のために (例えば、映像シーケンス 2 0 2 の将来のピクチャのためのインター予測基準ピクチャとして用いられるために) バッファ 2 3 4 (又は「復号化ピクチャバッファ」) 内に記憶され得る。符号器は、1つ以上の参照ピクチャを、時間的予測段階 2 0 4 4 において用いられるためにバッファ 2 3 4 内に記憶することができる。いくつかの実施形態では、符号器は、ループフィルタのパラメータ (例えば、ループフィルタ強度) を、量子化変換係数 2 1 6、予測データ 2 0 6 及び他の情報と共に、2 値符号化段階 2 2 6 において符号化することができる。

【 0 0 5 1 】

[0078] 図 3 A は、本開示の実施形態に従う、例示的な復号化プロセス 3 0 0 A の概略図を示す。プロセス 3 0 0 A は、図 2 A における圧縮プロセス 2 0 0 A に対応する復元プロセスであり得る。いくつかの実施形態では、プロセス 3 0 0 A はプロセス 2 0 0 A の再構成経路と似たものであり得る。復号器は、プロセス 3 0 0 A に従って映像ビットストリーム 2 2 8 を映像ストリーム 3 0 4 に復号化することができる。映像ストリーム 3 0 4 は映像シーケンス 2 0 2 とよく似たものであり得る。しかし、圧縮及び復元プロセス (例えば、図 2 A ~ 図 2 B における量子化段階 2 1 4) における情報損失のため、概して、映像ストリーム 3 0 4 は映像シーケンス 2 0 2 と同一ではない。図 2 A ~ 図 2 B におけるプロセス 2 0 0 A 及び 2 0 0 B と同様に、復号器は、映像ビットストリーム 2 2 8 内に符号化されたピクチャごとに基本処理ユニット (B P U) のレベルでプロセス 3 0 0 A を遂行することができる。例えば、復号器はプロセス 3 0 0 A を反復的な仕方で行うことができ、その場合、復号器は基本処理ユニットをプロセス 3 0 0 A の 1 回の反復において復号化することができる。いくつかの実施形態では、復号器は、プロセス 3 0 0 A を、映像ビットストリーム 2 2 8 内に符号化された各ピクチャの領域 (例えば、領域 1 1 4 ~ 1 1 8) のために並行して遂行することができる。

【 0 0 5 2 】

[0079] 図 3 A において、復号器は、符号化ピクチャの基本処理ユニット (「符号化 B P U 」と称される) に関連付けられた映像ビットストリーム 2 2 8 の部分を 2 値復号化段階 3 0 2 に供給することができる。2 値復号化段階 3 0 2 において、復号器は、当該部分を予測データ 2 0 6 及び量子化変換係数 2 1 6 に復号化することができる。復号器は、量子化変換係数 2 1 6 を逆量子化段階 2 1 8 及び逆変換段階 2 2 0 に供給し、再構成残差 B P U 2 2 2 を生成することができる。復号器は、予測データ 2 0 6 を予測段階 2 0 4 に供給し、予測 B P U 2 0 8 を生成することができる。復号器は、再構成残差 B P U 2 2 2 を予測 B P U 2 0 8 に加算し、予測基準 2 2 4 を生成することができる。いくつかの実施形態では、予測基準 2 2 4 をバッファ (例えば、コンピュータメモリ内の復号化ピクチャバッファ) 内に記憶することができる。復号器は、予測演算をプロセス 3 0 0 A の次の反復

において遂行するために、予測基準 2 2 4 を予測段階 2 0 4 に供給することができる。

【 0 0 5 3 】

[0080] 復号器は、符号化ピクチャの各符号化 B P U を復号化し、符号化ピクチャの次の符号化 B P U を符号化するための予測基準 2 2 4 を生成するために、プロセス 3 0 0 A を反復的に遂行することができる。符号化ピクチャの全ての符号化 B P U を復号化した後、復号器は、ピクチャを表示のために映像ストリーム 3 0 4 に出力し、映像ビットストリーム 2 2 8 内の次の符号化ピクチャを復号化するために進むことができる。

【 0 0 5 4 】

[0081] 2 値復号化段階 3 0 2 において、復号器は、符号器によって用いられた 2 値符号化技法（例えば、エントロピー符号化、可変長符号化、算術符号化、ハフマン符号化、コンテキスト適応 2 値算術符号化又は任意の他の可逆圧縮アルゴリズム）の逆演算を遂行することができる。いくつかの実施形態では、予測データ 2 0 6 及び量子化変換係数 2 1 6 のほかに、復号器は、例えば、予測モード、予測演算のパラメータ、変換の種類、量子化プロセスのパラメータ（例えば、量子化パラメータ）、符号器制御パラメータ（例えば、ビットレート制御パラメータ）、又は同様のものなど、他の情報を 2 値復号化段階 3 0 2 において復号化することができる。いくつかの実施形態では、映像ビットストリーム 2 2 8 がネットワークを通じてパケットの形で伝送される場合、復号器は、映像ビットストリーム 2 2 8 を、それを 2 値復号化段階 3 0 2 に供給する前にデパケット化することができる。

【 0 0 5 5 】

[0082] 図 3 B は、本開示の実施形態に従う、別の例示的な復号化プロセス 3 0 0 B の概略図を示す。プロセス 3 0 0 B はプロセス 3 0 0 A から変更され得る。例えば、プロセス 3 0 0 B は、ハイブリッド映像符号化規格（例えば、H . 2 6 x シリーズ）に準拠した復号器によって用いられ得る。プロセス 3 0 0 A と比べて、プロセス 3 0 0 B は、予測段階 2 0 4 を空間的予測段階 2 0 4 2 及び時間的予測段階 2 0 4 4 に追加的に分割し、ループフィルタ段階 2 3 2 及びバッファ 2 3 4 を追加的に含む。

【 0 0 5 6 】

[0083] プロセス 3 0 0 B において、復号化中の符号化ピクチャ（「現在のピクチャ」と称される）の符号化基本処理ユニット（「現在の B P U」と称される）のために、復号器によって 2 値復号化段階 3 0 2 から復号化された予測データ 2 0 6 は、いかなる予測モードが符号器によって現在の B P U を符号化するために用いられたかに依存して、様々な種類のデータを含むことができる。例えば、イントラ予測が符号器により、現在の B P U を符号化するために用いられた場合、予測データ 2 0 6 は、イントラ予測、イントラ予測演算のパラメータ、又は同様のものを指示する予測モードインジケータ（例えば、フラグ値）を含むことができる。イントラ予測演算のパラメータは、例えば、参照として用いられる 1 つ以上の隣接 B P U の場所（例えば、座標）、隣接 B P U のサイズ、外挿のパラメータ、原 B P U に対する隣接 B P U の方向、又は同様のものを含むことができる。別の例として、インター予測が符号器により、現在の B P U を符号化するために用いられた場合、予測データ 2 0 6 は、インター予測、インター予測演算のパラメータ、又は同様のものを指示する予測モードインジケータ（例えば、フラグ値）を含むことができる。インター予測演算のパラメータは、例えば、現在の B P U に関連付けられた参照ピクチャの数、参照ピクチャにそれぞれ関連付けられた重み、それぞれの参照ピクチャ内の 1 つ以上のマッ

チング領域の場所（例えば、座標）、マッチング領域にそれぞれ関連付けられた 1 つ以上の動きベクトル、又は同様のものを含むことができる。

【 0 0 5 7 】

[0084] 予測モードインジケータに基づいて、復号器は、空間的予測段階 2 0 4 2 において空間的予測（例えば、イントラ予測）を遂行するべきか、又は時間的予測段階 2 0 4 4 において時間的予測（例えば、インター予測）を遂行するべきかを決定することができる。このような空間的予測又は時間的予測を遂行することの詳細は図 2 B において説明されており、以下、繰り返されない。このような空間的予測又は時間的予測を遂行した後に

、復号器は、予測 B P U 2 0 8 を生成することができる。復号器は、図 3 A において説明されたように、予測 B P U 2 0 8 及び再構成残差 B P U 2 2 2 を加算し、予測基準 2 2 4 を生成することができる。

【 0 0 5 8 】

[0085] プロセス 3 0 0 B において、復号器は、予測演算をプロセス 3 0 0 B の次の反復において遂行するために、予測基準 2 2 4 を空間的予測段階 2 0 4 2 又は時間的予測段階 2 0 4 4 に供給することができる。例えば、現在の B P U が空間的予測段階 2 0 4 2 においてイントラ予測を用いて復号化される場合、予測基準 2 2 4 (例えば、復号化された現在の B P U) を生成した後に、復号器は、予測基準 2 2 4 を後の使用のために (例えば、現在のピクチャの次の B P U の外挿のために) 空間的予測段階 2 0 4 2 に直接供給することができる。現在の B P U が時間的予測段階 2 0 4 4 においてインター予測を用いて復号化される場合、予測基準 2 2 4 (例えば、全ての B P U が復号化された参照ピクチャ) を生成した後に、符号器は、予測基準 2 2 4 をループフィルタ段階 2 3 2 に供給し、歪み (例えば、ブロックングアーチファクト) を低減又は解消することができる。復号器は、図 2 B において説明されたとおりの仕方でループフィルタを予測基準 2 2 4 に適用することができる。ループフィルタリングされた参照ピクチャは、後の使用のために (例えば、映像ビットストリーム 2 2 8 の将来の符号化ピクチャのためのインター予測基準ピクチャとして用いられるために) バッファ 2 3 4 (例えば、コンピュータメモリ内の復号化ピクチャバッファ) 内に記憶され得る。復号器は、1 つ以上の参照ピクチャを、時間的予測段階 2 0 4 4 において用いられるためにバッファ 2 3 4 内に記憶することができる。いくつかの実施形態では、予測データ 2 0 6 の予測モードインジケータが、インター予測が現在の B P U を符号化するために用いられたことを指示するときには、予測データはループフィルタのパラメータ (例えば、ループフィルタ強度) をさらに含むことができる。

【 0 0 5 9 】

[0086] 図 4 は、本開示の実施形態に従う、映像を符号化又は復号化するための例示的な機器 4 0 0 のブロック図である。図 4 に示されるように、機器 4 0 0 はプロセッサ 4 0 2 を含むことができる。プロセッサ 4 0 2 が、本明細書において説明される命令を実行したとき、機器 4 0 0 は映像符号化又は復号化のための特殊機械になることができる。プロセッサ 4 0 2 は、情報を操作又は処理する能力を有する任意の種類の回路機構であり得る。例えば、プロセッサ 4 0 2 は、中央処理装置 (又は「C P U」)、グラフィック処理装置 (又は「G P U」)、ニューラル処理装置 (「N P U」)、マイクロコントローラユニット (「M C U」)、光プロセッサ、プログラマブル論理コントローラ、マイクロコントローラ、マイクロプロセッサ、デジタル信号プロセッサ、知的財産 (I P) コア、プログラマブル論理アレイ (P L A)、プログラマブルアレイ論理 (P A L)、ジェネリックアレイ論理 (G A L)、複合プログラマブル論理装置 (C P L D)、フィールドプログラマブルゲートアレイ (F P G A)、システムオンチップ (S o C)、特定用途向け集積回路 (A S I C) 又は同様のものの任意の数の任意の組み合わせを含むことができる。いくつかの実施形態では、プロセッサ 4 0 2 は、単一の論理構成要素としてグループ化されたプロセッサのセットでもあり得る。例えば、図 4 に示されるように、プロセッサ 4 0 2 は、プロセッサ 4 0 2 a、プロセッサ 4 0 2 b 及びプロセッサ 4 0 2 n を含む、複数のプロセッサを含むことができる。

【 0 0 6 0 】

[0087] 機器 4 0 0 は、データ (例えば、命令のセット、コンピュータコード、中間データ又は同様のもの) を記憶するように構成されたメモリ 4 0 4 も含むことができる。例えば、図 4 に示されるように、記憶されるデータは、プログラム命令 (例えば、プロセス 2 0 0 A、2 0 0 B、3 0 0 A 又は 3 0 0 B における段階を実施するためのプログラム命令)、並びに処理のためのデータ (例えば、映像シーケンス 2 0 2、映像ビットストリーム 2 2 8 又は映像ストリーム 3 0 4) を含むことができる。プロセッサ 4 0 2 は、(例えば、バス 4 1 0 を介して) プログラム命令及び処理のためのデータにアクセスし、プログラム命令を実行し、処理のためのデータに対する演算又は操作を遂行することができる。

メモリ 404 は、高速ランダムアクセス記憶デバイス又は不揮発性記憶デバイスを含むことができる。いくつかの実施形態では、メモリ 404 は、ランダムアクセスメモリ (RAM)、リードオンリーメモリ (ROM)、光ディスク、磁気ディスク、ハードドライブ、ソリッドステートドライブ、フラッシュドライブ、セキュリティデジタル (SD) カード、メモリスティック、コンパクトフラッシュ (登録商標) (CF) カード、又は同様のものの任意の数の任意の組み合わせを含むことができる。メモリ 404 は、単一の論理構成要素としてグループ化されたメモリのグループ (図 4 には示されていない) でもあり得る。

【0061】

[0088] バス 410 は、内部バス (例えば、CPU - メモリバス)、外部バス (例えば、ユニバーサルシリアルバスポート、ペリフェラルコンポーネントインターコネクトエクスプレスポート)、又は同様のものなど、機器 400 の内部の構成要素間でデータを転送する通信デバイスであり得る。

【0062】

[0089] 曖昧さを生じさせることなく説明を容易にするために、プロセッサ 402 及び他のデータ処理回路は本開示においてまとめて「データ処理回路」と称される。データ処理回路は、完全にハードウェアとして、又はソフトウェア、ハードウェア若しくはファームウェアの組み合わせとして実施され得る。加えて、データ処理回路は、単一の独立モジュールであり得るか、又は機器 400 の任意の他の構成要素に完全に若しくは部分的に組み合わされ得る。

【0063】

[0090] 機器 400 は、ネットワーク (例えば、インターネット、イントラネット、ローカルエリアネットワーク、移動通信ネットワーク、又は同様のもの) との有線又は無線通信を提供するためのネットワークインターフェース 406 をさらに含むことができる。いくつかの実施形態では、ネットワークインターフェース 406 は、ネットワークインターフェースコントローラ (NIC)、無線周波数 (RF) モジュール、トランスポンダ、トランシーバ、モデム、ルータ、ゲートウェイ、有線ネットワークアダプタ、無線ネットワークアダプタ、Bluetooth (登録商標) アダプタ、赤外線アダプタ、近距離無線通信 (「NFC」) アダプタ、セルラーネットワークチップ、又は同様のものの任意の数の任意の組み合わせを含むことができる。

【0064】

[0091] いくつかの実施形態では、任意選択的に、機器 400 は、1 つ以上の周辺デバイスへの接続を提供するための周辺インターフェース 408 をさらに含むことができる。図 4 に示されるように、周辺デバイスは、限定するものではないが、カーソル制御デバイス (例えば、マウス、タッチパッド若しくはタッチスクリーン)、キーボード、ディスプレイ (例えば、陰極線管ディスプレイ、液晶ディスプレイ若しくは発光ダイオードディスプレイ)、映像入力デバイス (例えば、カメラ若しくは映像アーカイブに結合された入力インターフェース)、又は同様のものを含むことができる。

【0065】

[0092] 映像コーデック (例えば、プロセス 200A、200B、300A 又は 300B を遂行するコーデック) は、機器 400 内の任意のソフトウェア又はハードウェアモジュールの任意の組み合わせとして実施され得ることに留意されたい。例えば、プロセス 200A、200B、300A、又は 300B の一部又は全ての段階は、メモリ 404 内にロードされ得るプログラム命令など、機器 400 の 1 つ以上のソフトウェアモジュールとして実施され得る。別の例として、プロセス 200A、200B、300A、又は 300B の一部又は全ての段階は、特殊データ処理回路 (例えば、FPGA、ASIC、NPU、又は同様のもの) など、機器 400 の 1 つ以上のハードウェアモジュールとして実施され得る。

【0066】

[0093] 量子化及び逆量子化機能ブロック (例えば、図 2A 又は図 2B の量子化 214 及び逆量子化 218、図 3A 又は図 3B の逆量子化 218) では、量子化パラメータ (Q

10

20

30

40

50

をもたらしている（各スライスはサブピクチャである）。

【 0 0 7 5 】

【00102】 開示するいくつかの実施形態によれば、サブピクチャ分割情報がシーケンスパラメータセット（SPS）内でシグナリングされる。図10は、本開示のいくつかの実施形態に係る、サブピクチャ分割の例示的なSPS構文を示す例示的な表1を示す。

【 0 0 7 6 】

【00103】 表1では、構文要素sps_num_subpics_minus1プラス1は1つのピクチャ内のサブピクチャの数を指定し、構文要素subpic_ctu_top_left_x[i]及びsubpic_ctu_top_left_y[i]はCtbSizeY単位のi番目のサブピクチャの左上CTUの位置を指定し、構文要素subpic_width_minus1[i]プラス1及び構文要素subpic_height_minus1[i]プラス1はCtbSizeY単位のi番目のサブピクチャの幅及び高さをそれぞれ指定する。これらの構文要素のセマンティクスは以下のとおりである。

【 0 0 7 7 】

【00104】 1に等しいsubpics_present_flagはサブピクチャパラメータがSPSRBSP構文内にあることを指定し、0に等しいsubpics_present_flagはサブピクチャパラメータがSPSRBSP構文内にないことを指定する。

【 0 0 7 8 】

【00105】 sps_num_subpics_minus1プラス1はサブピクチャの数を指定する。構文要素sps_num_subpics_minus1は、0～254の範囲内にある。ない場合、構文要素sps_num_subpics_minus1の値は0に等しいと推論される。

【 0 0 7 9 】

【00106】 subpic_ctu_top_left_x[i]はCtbSizeY単位のi番目のサブピクチャの左上CTUの水平位置を指定する。この構文要素の長さは $\text{Ceil}(\text{Log}_2(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素subpic_ctu_top_left_x[i]の値は0に等しいと推論される。

【 0 0 8 0 】

【00107】 subpic_ctu_top_left_y[i]はCtbSizeY単位のi番目のサブピクチャの左上CTUの垂直位置を指定する。この構文要素の長さは $\text{Ceil}(\text{Log}_2(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素subpic_ctu_top_left_y[i]の値は0に等しいと推論される。

【 0 0 8 1 】

【00108】 subpic_width_minus1[i]プラス1はCtbSizeY単位のi番目のサブピクチャの幅を指定する。この構文要素の長さは $\text{Ceil}(\text{Log}_2(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素subpic_width_minus1[i]の値は $\text{Ceil}(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}) - 1$ に等しいと推論される。

【 0 0 8 2 】

【00109】 subpic_height_minus1[i]プラス1はCtbSizeY単位のi番目のサブピクチャの高さを指定する。この構文要素の長さは $\text{Ceil}(\text{Log}_2(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素subpic_height_minus1[i]の値は $\text{Ceil}(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}) - 1$ に等しいと推論される。

【 0 0 8 3 】

【00110】 1に等しいsubpic_treated_as_pic_flag[i]は、符号化レイヤ映像シーケンス（CLVS）内の各符号化ピクチャのi番目のサブピクチャが、インループフィルタリング演算を除外する復号化プロセス内のピクチャとして扱われることを指定する。0に等しい構文要素subpic_treated_as_pic_flag[i]は、CLVS内の各符号化ピクチャのi番目のサブピクチャが、インループフィルタリング演算を除外する復号化プロセス内のピクチャとして扱われないことを指定する。ない場合、構文要素subpic_treated_as_pic_flag[i]の値は0に等しいと推論される。

【 0 0 8 4 】

10

20

30

40

50

[00111] 1 に等しいloop_filter_across_subpic_enabled_flag[i]は、C L V S 内の各符号化ピクチャ内の i 番目のサブピクチャの境界にわたってインループフィルタリング演算を行うことができることを指定する。0 に等しい構文要素loop_filter_across_subpic_enabled_flag[i]は、C L V S 内の各符号化ピクチャ内の i 番目のサブピクチャの境界にわたってインループフィルタリング演算が行われないことを指定する。ない場合、構文要素loop_filter_across_subpic_enabled_pic_flag[i]の値は1 に等しいと推論される。

【0085】

[00112] 本開示の開示する実施形態に従い、各サブピクチャに識別子を割り当てることができる。サブピクチャの識別子情報は、シーケンスパラメータセット (S P S) 、ピクチャパラメータセット (P P S) 、又はピクチャヘッダ (P H) 内でシグナリングされ得る。図 1 1 は、本開示のいくつかの実施形態に係る、サブピクチャ識別子の例示的な S P S 構文を示す例示的な表 2 を示す。図 1 2 は、本開示のいくつかの実施形態に係る、サブピクチャ識別子の例示的な P P S 構文を示す例示的な表 3 を示す。図 1 3 は、本開示のいくつかの実施形態に係る、サブピクチャ識別子の例示的な P H 構文を示す例示的な表 4 を示す。

10

【0086】

[00113] 表 2 ~ 表 4 に示すように、構文要素sps_subpic_id_present_flagはサブピクチャ I D マッピングが S P S 内にあるかどうかを示し、構文要素sps_subpic_id_signaling_present_flag、pps_subpic_id_signaling_present_flag及びph_subpic_id_signaling_present_flagはサブピクチャ I D マッピングが S P S 、 P P S 又は P H のそれぞれの中でシグナリングされるかどうかを示し、構文要素sps_subpic_id_len_minus1プラス1、構文要素pps_subpic_id_len_minus1プラス1 及び構文要素ph_subpic_id_len_minus1プラス1 は構文要素sps_subpic_id[i]、pps_subpic_id[i]及びph_subpic_id[i]をそれぞれ提示するために使用されるビット数を指定し、それらは S P S 、 P P S 及び P H のそれぞれの中でシグナリングされるサブピクチャ I D である。

20

【0087】

[00114] 上記の構文要素のセマンティクス及び関係するビットストリーム適合要件を下記のとおり記載する。

【0088】

[00115] 1 に等しいsps_subpic_id_present_flagはサブピクチャ I D マッピングが S P S 内にあることを指定し、0 に等しい構文要素sps_subpic_id_present_flagはサブピクチャ I D マッピングが S P S 内にないことを指定する。

30

【0089】

[00116] 1 に等しいsps_subpic_id_signaling_present_flagはサブピクチャ I D マッピングが S P S 内でシグナリングされることを指定し、0 に等しい構文要素sps_subpic_id_signaling_present_flagはサブピクチャ I D マッピングが S P S 内でシグナリングされないことを指定する。ない場合、構文要素sps_subpic_id_signaling_present_flagの値は0 に等しいと推論される。

【0090】

[00117] sps_subpic_id_len_minus1プラス1 は構文要素sps_subpic_id[i]を表すために使用されるビット数を指定する。構文要素sps_subpic_id_len_minus1の値は、0 ~ 1 5 の範囲内にあり得る。

40

【0091】

[00118] sps_subpic_id[i]は i 番目のサブピクチャのそのサブピクチャ I D を指定する。構文要素sps_subpic_id[i]の長さはsps_subpic_id_len_minus1 + 1 ビットである。ない場合及び構文要素sps_subpic_id_present_flagが0 に等しい場合、構文要素sps_subpic_id[i]の値は、0 ~ sps_num_subpics_minus1の範囲内の各 i について i に等しいと推論される。

【0092】

[00119] 1 に等しいpps_subpic_id_signaling_present_flagは、サブピクチャ I D

50

マッピングが P P S 内でシグナリングされることを指定する。0 に等しい構文要素 pps_subpic_id_signaling_present_flag はサブピクチャ I D マッピングが P P S 内でシグナリングされないことを指定する。構文要素 sps_subpic_id_present_flag が 0 であるか、又は構文要素 sps_subpic_id_signaling_present_flag が 1 に等しい場合、構文要素 pps_subpic_id_signaling_present_flag は 0 に等しくあり得る。

【 0 0 9 3 】

[00120] pps_num_subpics_minus1 プラス 1 は、P P S を参照する符号化ピクチャ内のサブピクチャの数を指定する。構文要素 pps_num_subpic_minus1 の値が構文要素 sps_num_subpics_minus1 に等しいことがビットストリーム適合の要件であり得る。

【 0 0 9 4 】

[00121] pps_subpic_id_len_minus1 プラス 1 は構文要素 pps_subpic_id[i] を表すために使用されるビット数を指定する。構文要素 pps_subpic_id_len_minus1 の値は、0 ~ 1 5 の範囲内にある。構文要素 pps_subpic_id_len_minus1 の値が C L V S 内の符号化ピクチャによって参照される全ての P P S について同じであることがビットストリーム適合の要件であり得る。

【 0 0 9 5 】

[00122] pps_subpic_id[i] は i 番目のサブピクチャのサブピクチャ I D を指定する。構文要素 pps_subpic_id[i] の長さは pps_subpic_id_len_minus1 + 1 ビットである。

【 0 0 9 6 】

[00123] 1 に等しい ph_subpic_id_signaling_present_flag はサブピクチャ I D マッピングが P H 内でシグナリングされることを指定し、0 に等しい構文要素 ph_subpic_id_signaling_present_flag はサブピクチャ I D マッピングが P H 内でシグナリングされないことを指定する。

【 0 0 9 7 】

[00124] ph_subpic_id_len_minus1 プラス 1 は構文要素 ph_subpic_id[i] を表すために使用されるビット数を指定する。構文要素 ph_subpic_id_len_minus1 の値は、0 ~ 1 5 の範囲内にあり得る。構文要素 ph_subpic_id_len_minus1 の値が C L V S 内の符号化ピクチャによって参照される全ての P H について同じであることがビットストリーム適合の要件であり得る。

【 0 0 9 8 】

[00125] ph_subpic_id[i] は i 番目のサブピクチャのそのサブピクチャ I D を指定する。構文要素 ph_subpic_id[i] の長さは ph_subpic_id_len_minus1 + 1 ビットである。

【 0 0 9 9 】

[00126] サブピクチャ I D に関係するこれらの構文要素を構文解析した後、サブピクチャ I D リスト SubpicIdList が以下の構文 (1) を使用して導出される：

```
for(i=0; i ≤ sps_num_subpics_minus1; i++)
    SubpicIdList[i]=sps_subpic_id_present_flag? 構文 ( 1 )
        (sps_subpic_id_signaling_present_flag?sps_subpic_id[i]:
        (ph_subpic_id_signaling_present_flag?ph_subpic_id[i]:pps_subpic_id[i]))
);i
```

【 0 1 0 0 】

[00127] しかし、サブピクチャ分割の上記のシグナリングにはいくつかの問題がある。まず、セマンティクスによれば、サブピクチャ I D が S P S 内にあるかどうかを両方の構文要素 sps_subpic_id_present_flag 及び sps_subpic_id_signaling_present_flag が指定する。表 2 によれば、これらの 2 つの構文要素の両方が真である場合にのみ、サブピクチャ i d 情報が S P S 内でシグナリングされる。従ってシグナリングに冗長性がある。第 2 に、たとえ sps_subpics_id_present_flag が真でも、pps_subpic_id_signaling_present_flag 又は ph_subpic_id_signaling_present_flag が真である場合、サブピクチャ I D が P P S 又は P H 内でやはりシグナリングされ得る。第 3 に、構文 (1) によれば、sps_subpic_id_present_flag が真でない場合、サブピクチャインデックスに等し

10

20

30

40

50

い既定IDが各サブピクチャに指定される。sps_subpic_id_present_flagが真である場合、SubpicIdList[i]がsps_subpic_id[i]、ph_subpic_id[i]又はpps_subpic_id[i]として導出される。しかし、sps_subpic_id_present_flagが真である場合にはサブピクチャIDがSPS、PPS又はPH内にはない可能性があり、そのため、その場合には構文要素pps_subpic_idの未定義値がSubpicIdListに指定される。構文(1)では、構文要素sps_subpic_id_present_flagが真である場合、構文要素sps_subpic_id_signaling_present_flag及びph_subpic_id_signaling_present_flagがどちらも偽であり、構文要素pps_subpic_id_signaling_present_flagの値に関係なく構文要素pps_subpic_id[i]がSubpicIdList[i]に指定される。構文要素ps_subpic_id_signaling_present_flagが偽である場合、構文要素pps_subpic_id[i]は未定義である。

10

【0101】

[00128] さらに、図10の表1に示すように、構文要素subpics_present_flagが真である場合、サブピクチャの数が最初にシグナリングされ、その後で各サブピクチャの左上位置、幅及び高さ並びに2つの制御フラグsubpic_treated_as_pic_flag及びloop_filter_across_subpic_enabled_flagがシグナリングされる。サブピクチャが1つのみの場合(構文要素sps_num_subpics_minus1が0に等しい場合)でも、左上位置、幅及び高さ並びにこれらの2つの制御フラグがシグナリングされる。しかし、ピクチャ内に1つのみのサブピクチャがある場合、サブピクチャがピクチャに等しく、そのためシグナリングされる情報がピクチャ自体から導出され得るため、これらの事項を指示する必要はない。

【0102】

20

[00129] さらに、サブピクチャはピクチャを分割することによって得られ、ピクチャは全てのサブピクチャを併合することによって形成される。最後のサブピクチャの位置及びサイズは、全ピクチャのサイズ並びに前の全てのサブピクチャの位置及びサイズから導出することができる。従って最後のサブピクチャの位置、幅及び高さの情報をシグナリングする必要はない。

【0103】

[00130] さらに図11の表2に示すように、構文要素sps_subpic_id_present_flagは構文要素subpics_present_flagの値に関係なく常にシグナリングされる。従って上記のシグナリングの方法では、たとえサブピクチャがなくてもサブピクチャ識別子が依然としてシグナリングされる可能性があり、そのことは無意味である。

30

【0104】

[00131] 本開示は、上記の問題を解決するためのシグナリングの方法を提供する。いくつかの例示的实施形態を以下で詳細に記載する。

【0105】

[00132] 本開示のいくつかの実施形態では、構文要素sps_subpic_id_present_flagが真だが、構文要素sps_subpic_id_signaling_present_flag及びpps_subpic_id_signaling_present_flagの両方が偽である場合にはピクチャヘッダ内でのサブピクチャIDのシグナリングを強いることができる。これにより、サブピクチャIDが定義されない事例を回避することができる。

【0106】

40

[00133] 例えばビットストリーム適合制約は以下の2つの方法で課すことができる。第1の方法では、ビットストリーム適合制約のためのセマンティクスは次のとおりである(イタリック体で強調): 1に等しいph_subpic_id_signaling_present_flagはサブピクチャIDマッピングがPH内でシグナリングされることを指定する。0に等しい構文要素ph_subpic_id_signaling_present_flagはサブピクチャIDマッピングがPH内でシグナリングされないことを指定する。構文要素sps_subpic_id_present_flagが1に等しく、構文要素sps_subpic_id_signaling_present_flagが0に等しく、構文要素pps_subpic_id_signaling_present_flagが0に等しい場合、構文要素ph_subpic_id_signaling_present_flagの値が1であることはビットストリーム適合要件であり得る。

【0107】

50

[00134] 第2の方法では、ビットストリーム適合制約のためのセマンティクスは次のとおりである（イタリック体で強調）：1に等しい`ph_subpic_id_signaling_present_flag`はサブピクチャIDマッピングがPH内でシグナリングされることを指定する。0に等しい構文要素`ph_subpic_id_signaling_present_flag`はサブピクチャIDマッピングがPH内でシグナリングされないことを指定する。構文要素`sps_subpic_id_present_flag`が1に等しく、構文要素`sps_subpic_id_signaling_present_flag`が0に等しく、CLVS内の符号化ピクチャによって参照されるPPSの全ての中の構文要素`pps_subpic_id_signaling_present_flag`が0に等しい場合、構文要素`ph_subpic_id_signaling_present_flag`の値が1に等しい、CLVSの参照内で符号化ピクチャによって参照される全てのPHの少なくとも1つのPHがあることはビットストリーム適合要件であり得る。

10

【0108】

[00135] 図14は、本開示のいくつかの実施形態に係る、この第2の方法の例示的なビットストリーム適合制約を示す概略図である。

【0109】

[00136] 構文要素`sps_subpic_id_present_flag`のセマンティクスは現在のVVCドラフト内で明確に定義されておらず、以下のように変更することができる（イタリック体で強調）。

【0110】

[00137] 1に等しい`sps_subpic_id_present_flag`はサブピクチャIDマッピングがSPS、PPS又はPH内にあることを指定する。0に等しい構文要素`sps_subpic_id_present_flag`はサブピクチャIDマッピングがSPS、PPS及びPH内にないことを指定する。

20

【0111】

[00138] このことは、構文要素`sps_subpic_id_present_flag`が真だが、構文要素`sps_subpic_id_signaling_present_flag`及び`pps_subpic_id_signaling_present_flag`の両方が偽である場合に構文要素`ph_subpic_id`がシグナリングされることを保証し得る。この構文を図11～図13の表2～表4に示す。

【0112】

[00139] 上記の実施形態では、サブピクチャID存在フラグが真（構文要素`sps_subpic_id_present_flag` = 1）である場合、使用されるサブピクチャIDがビットストリーム内で（SPS、PPS又はPH内で）シグナリングされ、推論規則は必要とされない。サブピクチャID存在フラグが真（構文要素`sps_subpic_id_present_flag` = 1）である場合、サブピクチャIDが存在することを構文要素`sps_subpic_id_present_flag`が示すため、SPS、PPS又はPHの1つにおいてサブピクチャIDのシグナリングを強いることは、ビットストリーム内でサブピクチャIDをシグナリングすることなしに推論規則を使用してサブピクチャIDを導出するよりもよい可能性がある。

30

【0113】

[00140] 別の例として図15は、本開示のいくつかの実施形態に係る、サブピクチャ識別子の別の例示的なPH構文を示す例示的な表5を示す。表5は、表4に示すPH構文の修正（ボックス1501内に示しイタリック体で強調表示する）を示す。表5を参照し、構文要素`sps_subpic_id_present_flag`が真だが、構文要素`sps_subpic_id_signaling_present_flag`及び`pps_subpic_id_signaling_present_flag`の両方が偽である場合、構文要素`ph_subpic_id_signaling_present_flag`が真であると推論することによって構文要素`ph_subpic_id`のシグナリングが強いられる。

40

【0114】

[00141] 構文要素`ph_subpic_id_signaling_present_flag`は以下の2つの代替的セマンティクス（イタリック体で強調）を有し得る。

【0115】

[00142] 第1のセマンティクスは以下を含む（イタリック体で強調）：

1に等しい`ph_subpic_id_signaling_present_flag`はサブピクチャIDマッピングが

50

P H内でシグナリングされることを指定する。0に等しい構文要素ph_subpic_id_signaling_present_flagはサブピクチャIDマッピングがP H内でシグナリングされないことを指定する。ない場合、構文要素ph_subpic_id_signaling_present_flagの値は1であると推論される。

【0116】

[00143] 第2のセマンティクスは以下を含む（イタリック体で強調）：

1に等しいph_subpic_id_signaling_present_flagはサブピクチャIDマッピングがP H内でシグナリングされることを指定し、0に等しい構文要素ph_subpic_id_signaling_present_flagはサブピクチャIDマッピングがP H内でシグナリングされないことを指定する。ないとき、構文要素sps_subpic_id_present_flagが1に等しく、構文要素sps_subpic_id_signaling_present_flagが0に等しい場合、構文要素ph_subpic_id_signaling_present_flagの値は1であると推論される。

10

【0117】

[00144] 構文要素sps_subpic_id_present_flagのセマンティクスは以下のように変更することができる（イタリック体で強調）。

【0118】

[00145] 1に等しいsps_subpic_id_present_flagはサブピクチャIDマッピングがSPS、PPS又はP H内にあることを指定する。0に等しい構文要素sps_subpic_id_present_flagはサブピクチャIDマッピングがSPS、PPS及びP H内にないことを指定する。

20

【0119】

[00146] 構文要素sps_subpic_id_present_flagが真である場合、サブピクチャIDの推論規則は、いくつかの実施形態では必要ない。従ってサブピクチャID存在フラグが真（構文要素sps_subpic_id_present_flag = 1）だがサブピクチャIDがSPS又はPPS内でシグナリングされない（構文要素sps_subpic_id_signaling_present_flag = 0及び構文要素pps_subpic_id_signaling_present_flag = 0である）場合、構文要素ph_subpic_id_signaling_present_flagのシグナリングがスキップされる。すると1ビット節約することができる。

【0120】

[00147] 別の例として、構文要素sps_subpic_id_present_flagが真だが、構文要素sps_subpic_id_signaling_present_flag及びpps_subpic_id_signaling_present_flagの両方が偽である（即ちサブピクチャIDがSPS又はPPS内でシグナリングされない）場合、構文要素ph_subpic_idのシグナリングが強いられ、この事例ではサブピクチャIDがP H内でシグナリングされることを意味する。構文要素pps_subpic_idがシグナリングされる（構文要素sps_subpic_id_present_flagが真であり、構文要素sps_subpic_id_signaling_present_flagが偽であり、構文要素pps_subpic_id_signaling_present_flagが真である）場合、構文要素ph_subpic_idをシグナリングすることはできない。図16は本開示のいくつかの実施形態に係る、サブピクチャ識別子の別の例示的なP H構文を示す例示的な表6を示す（ボックス1601内に強調を示しイタリック体で強調表示する）。

30

40

【0121】

[00148] リストSubpicIdList[i]が以下のように構文（2）に従って導出される。

```
for(i=0; i =sps_num_subpics_minus1; i++)
    SubpicIdList[i]=sps_subpic_id_present_flag? 構文（2）
        (sps_subpic_id_signaling_present_flag?sps_subpic_id[i]:
        (pps_subpic_id_signaling_present_flag?pps_subpic_id[i]:ph_subpic_id[i]
    ));i
```

【0122】

[00149] 構文要素sps_subpic_id_present_flagが真である場合、サブピクチャIDの推論規則は、いくつかの実施形態では必要ない。構文要素ph_subpic_id_signaling_p

50

resent_flagを除去することができる。従って構文要素sps_subpic_id_present_flagが1に等しく、構文要素sps_subpic_id_signaling_present_flagが0に等しく、構文要素pps_subpic_id_signaling_present_flagが1に等しい場合には1ビット節約される。

【0123】

[00150] サブピクチャIDがPPS内で既にシグナリングされている場合、いくつかの実施形態はサブピクチャIDをPH内で再びシグナリングすることによってPPS内のサブピクチャIDを符号器がオーバーライドする選択肢を与え得る。それは符号器にとってより柔軟な方法である。

【0124】

[00151] 本開示のいくつかの実施形態では、サブピクチャIDリストSubpicIdListを導出できることを保証するために推論規則が与えられる。

10

【0125】

[00152] 一例として、サブピクチャIDがSPS、PPS又はPH内でシグナリングされない場合、推論規則によって推論される構文要素pps_subpic_idの既定値を使用してサブピクチャIDリストSubpicIdListを導出するように構文要素pps_subpic_idに推論規則が与えられる。

【0126】

[00153] 構文要素pps_subpic_idのセマンティクスは次のとおりである（イタリック体で強調）：pps_subpic_id[i]はi番目のサブピクチャのサブピクチャIDを指定する。構文要素pps_subpic_id[i]の長さはpps_subpic_id_len_minus1 + 1ビットである。ない場合、構文要素pps_subpic_id[i]の値は、0 ~ pps_num_subpics_minus1の範囲内の各iについてiであると推論される。

20

【0127】

[00154] 別の例として、推論規則はSubpicIdListの導出プロセス内で与えられる。構文要素sps_subpic_id_present_flagが真であり、構文要素sps_subpic_id_signaling_present_flag、pps_subpic_id_signaling_present_flag及びph_subpic_id_signaling_present_flagが全て偽である場合、既定値がSubpicIdList[i]に指定される。

【0128】

[00155] SubpicIdListの導出は以下のように構文(3)に従う（イタリック体で強調）：

30

```
for(i=0;i =sps_num_subpics_minus1;i++)
  SubpicIdList[i]=sps_subpic_id_present_flag? 構文(3)
    (sps_subpic_id_signaling_present_flag?sps_subpic_id[i]:
    (ph_subpic_id_signaling_present_flag?ph_subpic_id[i]:
    (pps_subpic_id_signaling_present_flag?pps_subpic_id[i]:i)))
```

【0129】

[00156] いくつかの実施形態では、いずれかのpps_subpic_idに推論規則を課すことにより、たとえビットストリーム内でサブピクチャIDを全くシグナリングしなくてもSubpicIdListを導出できることを保証することができる。従ってサブピクチャIDのシグナリングに充てられるビットを節約することができる。

40

【0130】

[00157] 本開示のいくつかの実施形態では、PHよりもPPS内でシグナリングされるサブピクチャIDの方に高い優先度を与えるために、サブピクチャIDリストSubpicIdListの導出規則を変更することができる。従って、構文要素ph_subpic_id_signaling_present_flagの前に構文要素pps_subpic_id_signaling_present_flagがチェックされる。

【0131】

[00158] 一例として、SubpicIdListの導出規則は以下に示す構文(4)に従い（イタリック体で強調）、構文要素ph_subpic_idに対して推論規則が与えられる。

```
for(i=0;i =sps_num_subpics_minus1;i++)
```

50

```

SubpicIdList[i]=sps_subpic_id_present_flag? 構文 ( 4 )
(sps_subpic_id_signaling_present_flag?sps_subpic_id[i]:
(pps_subpic_id_signaling_present_flag?pps_subpic_id[i]:ph_subpic_id[i]
)):i

```

【 0 1 3 2 】

[00159] 構文 (4) に従うセマンティクス (イタリック体で強調) は次のとおりである :

ph_subpic_id[i] は i 番目のサブピクチャのサブピクチャ ID を指定する。構文要素 ph_subpic_id[i] の長さは ph_subpic_id_len_minus1 + 1 ビットである。ない場合、構文要素 ph_subpic_id[i] の値は、0 ~ ph_num_subpics_minus1 の範囲内の各 i について i であると推論される。

10

【 0 1 3 3 】

[00160] 別の例として、導出規則 SubpicIDList は以下に示す構文 (5) に従い (イタリック体で強調)、この例では構文要素 ph_subpic_id に対する追加の推論規則はない。

```

for(i=0;i =sps_num_subpics_minus1;i++)
SubpicIdList[i]=sps_subpic_id_present_flag? 構文 ( 5 )
(sps_subpic_id_signaling_present_flag?sps_subpic_id[i]:
(pps_subpic_id_signaling_present_flag?pps_subpic_id[i]:
(ph_subpic_id_signaling_present_flag?ph_subpic_id[i]:i)):i

```

【 0 1 3 4 】

20

[00161] ビットストリーム内でサブピクチャ ID を全くシグナリングしなくても SubpicIdList を正しく導出できることを保証するために、構文要素 ph_subpic_id 又は SubpicIdList の導出プロセスに推論規則が与えられる。従って推論規則によって推論される既定のサブピクチャ ID が使用される場合、サブピクチャのシグナリングに充てられるビットを節約することができる。

【 0 1 3 5 】

[00162] 本開示のいくつかの実施形態では、サブピクチャの数が 1 に等しいときサブピクチャに関してシグナリングされる冗長情報を除去することができる。

【 0 1 3 6 】

[00163] 一例として、図 17A の表 7A (ボックス 1701 ~ 1702 内に強調を示しイタリック体で強調表示する) 又は図 17B の表 7B (ボックス 1711 ~ 1712 内に強調を示しイタリック体で強調表示する) に SPS 構文を示す。表 7A 及び表 7B は等価であることが理解されよう。表 7A 及び表 7B 内の構文に従うセマンティクス (イタリック体で強調) を以下に示す。

30

【 0 1 3 7 】

[00164] subpic_ctu_top_left_x[i] は CtbSizeY 単位の i 番目のサブピクチャの左上 CTU の水平位置を指定する。この構文要素の長さは $\text{Ceil}(\text{Log}_2(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素 subpic_ctu_top_left_x[i] の値は 0 に等しいと推論される。

【 0 1 3 8 】

40

[00165] subpic_ctu_top_left_y[i] は CtbSizeY 単位の i 番目のサブピクチャの左上 CTU の垂直位置を指定する。この構文要素の長さは $\text{Ceil}(\text{Log}_2(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素 subpic_ctu_top_left_y[i] の値は 0 に等しいと推論される。

【 0 1 3 9 】

[00166] subpic_width_minus1[i] プラス 1 は CtbSizeY 単位の i 番目のサブピクチャの幅を指定する。この構文要素の長さは $\text{Ceil}(\text{Log}_2(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素 subpic_width_minus1[i] の値は $\text{Ceil}(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}) - 1$ に等しいと推論される。ここでは「Ceil ()」は最も近い整数に切り上げるための関数である。従って Ceil (pic_

50

$\text{width_max_in_luma_samples} \div \text{CtbSizeY} - 1$ は $(\text{pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) / \text{CtbSizeY} - 1$ に等しく、「/」は整数除算である。

【 0 1 4 0 】

[00167] $\text{subpic_height_minus1}[i]$ プラス 1 は CtbSizeY 単位の i 番目のサブピクチャの高さを指定する。この構文要素の長さは $\text{Ceil}(\text{Log2}(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素 $\text{subpic_height_minus1}[i]$ の値は $\text{Ceil}(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}) - 1$ に等しいと推論される。ここでは「 $\text{Ceil}()$ 」は最も近い整数に切り上げるための関数である。従って $\text{Ceil}(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}) - 1$ は $(\text{pic_height_max_in_luma_samples} + \text{CtbSizeY} - 1) / \text{CtbSizeY} - 1$ に等しく、「/」は整数除算である。

10

【 0 1 4 1 】

[00168] 1 に等しい $\text{subpic_treated_as_pic_flag}[i]$ は、CLVS 内の各符号化ピクチャの i 番目のサブピクチャが、インループフィルタリング演算を除外する復号化プロセス内のピクチャとして扱われることを指定する。0 に等しい構文要素 $\text{subpic_treated_as_pic_flag}[i]$ は、CLVS 内の各符号化ピクチャの i 番目のサブピクチャが、インループフィルタリング演算を除外する復号化プロセス内のピクチャとして扱われないことを指定する。ないとき、構文要素 $\text{subpics_present_flag}$ が 1 に等しく、構文要素 $\text{sps_num_subpics_minus1}$ が 0 に等しい場合、構文要素 $\text{subpic_treated_as_pic_flag}[i]$ の値は 1 に等しいと推論され、さもなければ構文要素 $\text{subpic_treated_as_pic_flag}[i]$ の値は 0 に等しいと推論される。

20

【 0 1 4 2 】

[00169] 1 に等しい $\text{loop_filter_across_subpic_enabled_flag}[i]$ は、CLVS 内の各符号化ピクチャ内の i 番目のサブピクチャの境界にわたってインループフィルタリング演算を行うことができることを指定する。0 に等しい構文要素 $\text{loop_filter_across_subpic_enabled_flag}[i]$ は、CLVS 内の各符号化ピクチャ内の i 番目のサブピクチャの境界にわたってインループフィルタリング演算が行われないことを指定する。ないとき、構文要素 $\text{subpics_present_flag}$ が 1 に等しく、構文要素 $\text{sps_num_subpics_minus1}$ が 0 に等しい場合、構文要素 $\text{loop_filter_across_subpic_enabled_flag}[i]$ の値は 0 に等しいと推論され、さもなければ構文要素 $\text{loop_filter_across_subpic_enabled_pic_flag}[i]$ の値は 1 に等しいと推論される。

30

【 0 1 4 3 】

[00170] 別の例として、図 18 の表 8 (ボックス 1801 ~ 1802 内に強調を示しイタリック体で強調表示する) に SPS 構文を示す。表 8 内の構文に従うセマンティクス (イタリック体で強調) を以下に示す。

【 0 1 4 4 】

[00171] $\text{subpic_ctu_top_left_x}[i]$ は CtbSizeY 単位の i 番目のサブピクチャの左上 CTU の水平位置を指定する。この構文要素の長さは $\text{Ceil}(\text{Log2}(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素 $\text{subpic_ctu_top_left_x}[i]$ の値は 0 に等しいと推論される。

【 0 1 4 5 】

40

[00172] $\text{subpic_ctu_top_left_y}[i]$ は CtbSizeY 単位の i 番目のサブピクチャの左上 CTU の垂直位置を指定する。この構文要素の長さは $\text{Ceil}(\text{Log2}(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素 $\text{subpic_ctu_top_left_y}[i]$ の値は 0 に等しいと推論される。

【 0 1 4 6 】

[00173] $\text{subpic_width_minus1}[i]$ プラス 1 は CtbSizeY 単位の i 番目のサブピクチャの幅を指定する。この構文要素の長さは $\text{Ceil}(\text{Log2}(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素 $\text{subpic_width_minus1}[i]$ の値は $\text{Ceil}((\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}) - 1)$ に等しいと推論される。ここでは「 $\text{Ceil}()$ 」は最も近い整数に切り上げるための関数である。従って $\text{Ceil}(\text{pic}$

50

$\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}) - 1$ は $(\text{pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) / \text{CtbSizeY} - 1$ に等しく、「/」は整数除算である。

【0147】

[00174] $\text{subpic_height_minus1}[i]$ プラス 1 は CtbSizeY 単位の i 番目のサブピクチャの高さを指定する。この構文要素の長さは $\text{Ceil}(\text{Log2}(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素 $\text{subpic_height_minus1}[i]$ の値は $\text{Ceil}(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}) - 1$ に等しいと推論される。ここでは「 $\text{Ceil}()$ 」は最も近い整数に切り上げるための関数である。従って $\text{Ceil}(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}) - 1$ は $(\text{pic_height_max_in_luma_samples} + \text{CtbSizeY} - 1) / \text{CtbSizeY} - 1$ に等しく、「/」は整数除算である。

10

【0148】

[00175] 本開示のいくつかの実施形態では、最後のサブピクチャの位置及び/又はサイズ情報をスキップし、全ピクチャのサイズ並びに前の全てのサブピクチャのサイズ及び位置から導出することができる。図19は、本開示のいくつかの実施形態に係る、別の例示的なSPS構文を示す例示的な表9を示す。表9(ボックス1901~1902内に強調を示しイタリック体で強調表示する)では、構文要素 $\text{sps_num_subpics_minus1}$ に等しいインデックスを有するサブピクチャである最後のサブピクチャの幅及び高さがスキップされる。

【0149】

[00176] 最後のサブピクチャの幅及び高さは、全ピクチャの幅及び高さ並びに最後のサブピクチャの左上位置から導出される。

20

【0150】

[00177] 以下は表9のセマンティクスに従うセマンティクス(イタリック体で強調)である。

【0151】

[00178] $\text{subpic_ctu_top_left_x}[i]$ は CtbSizeY 単位の i 番目のサブピクチャの左上CTUの水平位置を指定する。この構文要素の長さは $\text{Ceil}(\text{Log2}(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素 $\text{subpic_ctu_top_left_x}[i]$ の値は 0 に等しいと推論される。

【0152】

30

[00179] $\text{subpic_ctu_top_left_y}[i]$ は CtbSizeY 単位の i 番目のサブピクチャの左上CTUの垂直位置を指定する。この構文要素の長さは $\text{Ceil}(\text{Log2}(\text{pic_height_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素 $\text{subpic_ctu_top_left_y}[i]$ の値は 0 に等しいと推論される。

【0153】

[00180] $\text{subpic_width_minus1}[i]$ プラス 1 は CtbSizeY 単位の i 番目のサブピクチャの幅を指定する。この構文要素の長さは $\text{Ceil}(\text{Log2}(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY}))$ ビットである。ない場合、構文要素 $\text{subpic_width_minus1}[i]$ の値は $\text{Ceil}((\text{pic_width_max_in_luma_samples}) \div \text{CtbSizeY}) - 1 - (i == \text{sps_num_subpics_minus1} ? \text{subpic_ctu_top_left_x}[\text{sps_num_subpics_minus1}] : 0)$ に等しいと推論される。ここでは「 $\text{Ceil}()$ 」は最も近い整数に切り上げるための関数である。つまり $\text{Ceil}(\text{pic_width_max_in_luma_samples} \div \text{CtbSizeY})$ は $(\text{pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) / \text{CtbSizeY} - 1$ に等しく、「/」は整数除算である。「 $\text{sps_num_subpics_minus1}$ 」はピクチャ内のサブピクチャの数である。ピクチャ内の最後のサブピクチャに関して、 i は $\text{sps_num_subpics_minus1}$ に等しく、この事例では $\text{subpic_width_minus1}[i]$ は $(\text{pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) / \text{CtbSizeY} - 1 - \text{subpic_ctu_top_left_x}[\text{sps_num_subpics_minus1}]$ に等しいと推論される。ピクチャ内に1つのみのサブピクチャがある場合、 i は、0であることのみが可能であり、 $\text{subpic_ctu_top_left_x}[0]$ は、0である。従って、 $\text{subpic_width_minus1}[i]$ は、 $(\text{pic_width_max_in_luma_samples} + \text{CtbSizeY} - 1) / \text{CtbSizeY} - 1 - \text{subpic_ctu_t}$

40

50

op_left_x[0]又は $(pic_width_max_in_luma_samples + CtbSizeY - 1) / CtbSizeY - 1$ に等しいと推論される。

【0154】

[00181] subpic_height_minus1[i]プラス1はCtbSizeY単位のi番目のサブピクチャの高さを指定する。この構文要素の長さは $Ceil(\log_2(pic_height_max_in_luma_samples \div CtbSizeY))$ ビットである。ない場合、構文要素subpic_height_minus1[sps_num_subpics_minus1]の値は $(Ceil(pic_height_max_in_luma_samples) \div CtbSizeY) - 1 - (i == sps_num_subpics_minus1 ? subpic_ctu_top_left_y[i]:0)$ に等しいと推論される。ここでは「Ceil()」は最も近い整数に切り上げるための関数である。つまり $Ceil(pic_height_max_in_luma_samples \div CtbSizeY)$ は $(pic_height_max_in_luma_samples + CtbSizeY - 1) / CtbSizeY - 1$ に等しく、「/」は整数除算である。「sps_num_subpics_minus1」はピクチャ内のサブピクチャの数である。ピクチャ内の最後のサブピクチャに関して、iはsps_num_subpics_minus1に等しく、この事例ではsubpic_height_minus1[i]は $(pic_height_max_in_luma_samples + CtbSizeY - 1) / CtbSizeY - 1 - subpic_ctu_top_left_y[sps_num_subpics_minus1]$ に等しいと推論される。ピクチャ内に1つのみのサブピクチャがある場合、iは、0であることのみが可能であり、subpic_ctu_top_left_y[0]は、0である。従って、subpic_width_minus1[i]は、 $(pic_height_max_in_luma_samples + CtbSizeY - 1) / CtbSizeY - 1 - subpic_ctu_top_left_y[0]$ 又は $(pic_height_max_in_luma_samples + CtbSizeY - 1) / CtbSizeY - 1$ に等しいと推論される。

【0155】

[00182] 本開示のいくつかの実施形態では、サブピクチャがある場合にのみサブピクチャIDがシグナリングされる。図20は、本開示のいくつかの実施形態に係る、別の例示的なSPS構文(ボックス2001~2002内に強調を示しイタリック体で強調表示する)を示す例示的な表10を示す。

【0156】

[00183] 図21は、本開示のいくつかの実施形態に係る、例示的な映像処理方法2100のフローチャートを示す。方法2100は、符号器によって(例えば図2Aのプロセス200A又は図2Bのプロセス200Bにより)、復号器によって(例えば図3Aのプロセス300A又は図3Bのプロセス300Bにより)、又は機器(例えば図4の機器400)の1つ若しくは複数のソフトウェア若しくはハードウェアコンポーネントによって実行され得る。例えばプロセッサ(例えば図4のプロセッサ402)が方法2100を実行することができる。いくつかの実施形態では、方法2100は、コンピュータ(例えば図4の機器400)によって実行されるプログラムコードなどのコンピュータ実行可能命令を含むコンピュータ可読媒体内に具体化されるコンピュータプログラム製品によって実装することができる。

【0157】

[00184] ステップ2101で、サブピクチャIDマッピングがビットストリーム内にあるかどうかを判定することができる。いくつかの実施形態では、方法2100は、サブピクチャIDマッピングがビットストリーム内にあるかどうかを示すフラグをシグナリングすることを含み得る。例えばこのフラグは、図11の表2、図13の表4、図15の表5又は図16の表6に示すsps_subpic_id_present_flagであり得る。

【0158】

[00185] ステップ2103で、1つ以上のサブピクチャIDが第1の構文又は第2の構文内でシグナリングされるかどうかを判定することができる。ステップ2105で、サブピクチャIDマッピングがあると判定され、及び1つ以上のサブピクチャIDが第1の構文及び第2の構文内でシグナリングされないと判定されることに応答して、1つ以上のサブピクチャIDを第3の構文内でシグナリングする。第1の構文、第2の構文、又は第3の構文は、SPS、PPS及びPHの1つである。例えば、第1の構文、第2の構文、及び第3の構文は、SPS、PPS及びPHのそれぞれである。従ってサブピクチャID

マッピングがある（例えばsps_subpic_id_present_flag = 1である）場合、サブピクチャIDは、SPS、PPS、又はPH内でシグナリングされることを強いられ得る。

【0159】

[00186] いくつかの実施形態では、方法2100は、1つ以上のサブピクチャIDが第1の構文（例えば図11の表2に示すSPS）内でシグナリングされることを示す第1のフラグ（例えば図11の表2、図15の表5又は図16の表6に示すsps_subpic_id_signaling_present_flag）をシグナリングすることを含み得る。いくつかの実施形態では、方法2100は、1つ以上のサブピクチャIDが第2の構文（例えば図12の表3に示すPPS）内でシグナリングされることを示す第2のフラグ（例えば図12の表3、図15の表5又は図16の表6に示すpps_subpic_id_signaling_present_flag）をシグナリングすることを含み得る。いくつかの実施形態では、方法2100は、1つ以上のサブピクチャIDが第3の構文（例えば図15の表5に示すPH）内でシグナリングされることを示す第3のフラグ（例えば図15の表5に示すph_subpic_id_signaling_present_flag）をシグナリングすることを含み得る。

10

【0160】

[00187] いくつかの実施形態では、方法2100は、1つ以上のサブピクチャIDが第3の構文内でシグナリングされることを示す第3のフラグをビットストリームが含むかどうかを判定すること、及び第3のフラグをビットストリームが含まないことに応答して、1つ以上のサブピクチャIDを第3の構文内でシグナリングすることを含み得る。例えば第3の構文はPHとすることができ、第3のフラグはph_subpic_id_signaling_present_flagであり得る。ph_subpic_id_signaling_present_flagがPH内でシグナリングされない場合、ph_subpic_id_signaling_present_flagは1であると推論することができ、1つ以上のサブピクチャIDがPH内でシグナリングされる。

20

【0161】

[00188] いくつかの実施形態では、方法2100は、1つ以上のサブピクチャIDが第1の構文内でシグナリングされないことと判定されることに応答して、1つ以上のサブピクチャIDを第2の構文及び第3の構文（例えば図15の表5）内でシグナリングすることを含み得る。

【0162】

[00189] 図22は、本開示のいくつかの実施形態に係る、例示的な映像処理方法2200のフローチャートを示す。方法2200は、符号器によって（例えば図2Aのプロセス200A又は図2Bのプロセス200Bにより）、復号器によって（例えば図3Aのプロセス300A又は図3Bのプロセス300Bにより）、又は機器（例えば図4の機器400）の1つ若しくは複数のソフトウェア若しくはハードウェアコンポーネントによって実行され得る。例えばプロセッサ（例えば図4のプロセッサ402）が方法2200を実行することができる。いくつかの実施形態では、方法2200は、コンピュータ（例えば図4の機器400）によって実行されるプログラムコードなどのコンピュータ実行可能命令を含むコンピュータ可読媒体内に具体化されるコンピュータプログラム製品によって実装することができる。

30

【0163】

[00190] ステップ2201で、1つ以上のサブピクチャIDが、SPS、PH、又はPPSの少なくとも1つにおいてシグナリングされるかどうかを判定することができる。いくつかの実施形態では、方法2200は、1つ以上のサブピクチャIDがPPS内でシグナリングされるかどうかを判定する前に、1つ以上のサブピクチャIDがPH内でシグナリングされるかどうかを判定することを含み得る。いくつかの実施形態では、方法2200は、1つ以上のサブピクチャIDがPH内でシグナリングされるかどうかを判定する前に、1つ以上のサブピクチャIDがPPS内でシグナリングされるかどうかを判定することを含み得る。

40

【0164】

[00191] ステップ2203で、1つ以上のサブピクチャIDがSPS、PH、及びP

50

P S 内でシグナリングされないとは判定されることに応答して、1つ以上のサブピクチャIDが既定値を有すると判定することができる。

【0165】

【00192】 図23は、本開示のいくつかの実施形態に係る、例示的な映像処理方法2300のフローチャートを示す。方法2300は、符号器によって（例えば図2Aのプロセス200A又は図2Bのプロセス200Bにより）、復号器によって（例えば図3Aのプロセス300A又は図3Bのプロセス300Bにより）、又は機器（例えば図4の機器400）の1つ若しくは複数のソフトウェア若しくはハードウェアコンポーネントによって実行され得る。例えばプロセッサ（例えば図4のプロセッサ402）が方法2300を実行することができる。いくつかの実施形態では、方法2300は、コンピュータ（例えば図4の機器400）によって実行されるプログラムコードなどのコンピュータ実行可能命令を含むコンピュータ可読媒体内に具体化されるコンピュータプログラム製品によって実装することができる。

10

【0166】

【00193】 ステップ2301で、符号化ピクチャのサブピクチャの数が1に等しいかどうかを判定することができる。例えば、図18の表8に示すように、`sps_num_subpic_minus1`が0よりも大きいかどうかを判定することができる。

【0167】

【00194】 ステップ2303で、サブピクチャの数が1に等しいと判定されることに応答して、符号化ピクチャのサブピクチャを復号化プロセス内のピクチャとして扱うことができる。例えばサブピクチャの数が1に等しいと判定されることに応答して、フラグ`subpic_treated_as_pic_flag[i]`は1に等しいと推論され得る。いくつかの実施形態では、サブピクチャの数が1に等しいと判定されることに応答して、インループフィルタリング演算を除外することができる。

20

【0168】

【00195】 図24は、本開示のいくつかの実施形態に係る、例示的な映像処理方法2400のフローチャートを示す。方法2400は、符号器によって（例えば図2Aのプロセス200A又は図2Bのプロセス200Bにより）、復号器によって（例えば図3Aのプロセス300A又は図3Bのプロセス300Bにより）、又は機器（例えば図4の機器400）の1つ若しくは複数のソフトウェア若しくはハードウェアコンポーネントによって実行され得る。例えばプロセッサ（例えば図4のプロセッサ402）が方法2400を実行することができる。いくつかの実施形態では、方法2400は、コンピュータ（例えば図4の機器400）によって実行されるプログラムコードなどのコンピュータ実行可能命令を含むコンピュータ可読媒体内に具体化されるコンピュータプログラム製品によって実装することができる。

30

【0169】

【00196】 ステップ2401で、サブピクチャがピクチャの最後のサブピクチャかどうかを判定することができる。ステップ2403で、サブピクチャが最後のサブピクチャであると判定されることに応答して、サブピクチャの位置又はサイズの情報をピクチャのサイズ並びにピクチャの前のサブピクチャのサイズ及び位置から導出することができる。

40

【0170】

【00197】 図25は、本開示のいくつかの実施形態に係る、例示的な映像処理方法2500のフローチャートを示す。方法2500は、符号器によって（例えば図2Aのプロセス200A又は図2Bのプロセス200Bにより）、復号器によって（例えば図3Aのプロセス300A又は図3Bのプロセス300Bにより）、又は機器（例えば図4の機器400）の1つ若しくは複数のソフトウェア若しくはハードウェアコンポーネントによって実行され得る。例えばプロセッサ（例えば図4のプロセッサ402）が方法2500を実行することができる。いくつかの実施形態では、方法2500は、コンピュータ（例えば図4の機器400）によって実行されるプログラムコードなどのコンピュータ実行可能命令を含むコンピュータ可読媒体内に具体化されるコンピュータプログラム製品によって実

50

装することができる。

【 0 1 7 1 】

[00198] ステップ 2 5 0 1 で、サブピクチャがピクチャ内にあるかどうかを判定することができる。例えばこの判定はフラグ（例えば図 2 0 の表 1 0 に示す subpics_present_flag）に基づいて行うことができる。

【 0 1 7 2 】

[00199] ステップ 2 5 0 3 で、1 つ以上のサブピクチャがピクチャ内にあると判定されることに応答して、サブピクチャ ID マッピングが S P S 内にあるかどうかを示す第 1 のフラグをシグナリングすることができる。例えば第 1 のフラグは図 2 0 の表 1 0 に示す sps_subpic_id_present_flag であり得る。

10

【 0 1 7 3 】

[00200] いくつかの実施形態では、方法 2 5 0 0 は、サブピクチャ ID マッピングが S P S 内にあることを第 1 のフラグが示すことに応答して、サブピクチャ ID マッピングが S P S 内でシグナリングされるかどうかを示す第 2 のフラグをシグナリングすることを含み得る。サブピクチャ ID マッピングが S P S 内でシグナリングされることを第 2 のフラグが示すことに応答して、1 つ以上のサブピクチャのサブピクチャ ID を S P S 内でシグナリングすることができる。例えば第 2 のフラグは図 2 0 の表 1 0 に示す sps_subpic_id_signaling_present_flag であり得る。sps_subpic_id_signaling_present_flag が真である場合、sps_subpic_id[i] をシグナリングすることができる。

【 0 1 7 4 】

20

[00201] 実施形態は、以下の条項を用いてさらに記述することができる：

1 . 映像処理方法であって、
サブピクチャ ID マッピングがビットストリーム内にあるかどうかを判定すること、
1 つ以上のサブピクチャ ID が第 1 の構文又は第 2 の構文内でシグナリングされるかどうかを判定すること、及び
サブピクチャ ID マッピングがあると判定され、及び 1 つ以上のサブピクチャ ID が第 1 の構文及び第 2 の構文内でシグナリングされないと判定されることに応答して、1 つ以上のサブピクチャ ID を第 3 の構文内でシグナリングすること
を含む映像処理方法。

2 . 第 1 の構文、第 2 の構文、又は第 3 の構文は、シーケンスパラメータセット（S P S）、ピクチャパラメータセット（P P S）及びピクチャヘッダ（P H）の 1 つである、
条項 1 に記載の方法。

30

3 . 1 つ以上のサブピクチャ ID が第 1 の構文内でシグナリングされることを示す第 1 のフラグをシグナリングすること、又は

1 つ以上のサブピクチャ ID が第 2 の構文内でシグナリングされることを示す第 2 のフラグをシグナリングすること

をさらに含む、条項 1 及び 2 に記載の方法。

4 . 1 つ以上のサブピクチャ ID が第 3 の構文内でシグナリングされることを示す第 3 のフラグをシグナリングすること

をさらに含む、条項 1 ~ 3 の何れか一項に記載の方法。

40

5 . 1 つ以上のサブピクチャ ID が第 3 の構文内でシグナリングされることを示す第 3 のフラグをビットストリームが含むかどうかを判定すること、及び

第 3 のフラグをビットストリームが含まないことに応答して、1 つ以上のサブピクチャ ID を第 3 の構文内でシグナリングすること

をさらに含む、条項 1 ~ 4 の何れか一項に記載の方法。

6 . 1 つ以上のサブピクチャ ID が第 1 の構文内でシグナリングされないと判定されることに応答して、1 つ以上のサブピクチャ ID を第 2 の構文及び第 3 の構文内でシグナリングすること

をさらに含む、条項 1 ~ 5 の何れか一項に記載の方法。

7 . サブピクチャ ID マッピングがビットストリーム内にあるかどうかを示す第 4 のフ

50

ラグをシグナリングすること

をさらに含む、条項 1 ~ 6 の何れか一項に記載の方法。

8 . 映像処理機器であって、

命令を記憶するための少なくとも 1 つのメモリと、

少なくとも 1 つのプロセッサとを含み、少なくとも 1 つのプロセッサは、

サブピクチャ ID マッピングがビットストリーム内にあるかどうかを判定すること、

1 つ以上のサブピクチャ ID が第 1 の構文又は第 2 の構文内でシグナリングされるかどうかを判定すること、及び

サブピクチャ ID マッピングがあると判定され、及び 1 つ以上のサブピクチャ ID が第 1 の構文及び第 2 の構文内でシグナリングされないと判定されることに応答して、1 つ以上のサブピクチャ ID を第 3 の構文内でシグナリングすること

10

を機器に行わせるために命令を実行するように構成される、映像処理機器。

9 . 第 1 の構文、第 2 の構文、又は第 3 の構文は、シーケンスパラメータセット (S P S)、ピクチャパラメータセット (P P S) 及びピクチャヘッダ (P H) の 1 つである、条項 8 に記載の機器。

10 . 少なくとも 1 つのプロセッサは、

1 つ以上のサブピクチャ ID が第 1 の構文内でシグナリングされることを示す第 1 のフラグをシグナリングすること、又は

1 つ以上のサブピクチャ ID が第 2 の構文内でシグナリングされることを示す第 2 のフラグをシグナリングすること

20

を機器に行わせるために命令を実行するように構成される、条項 8 及び 9 に記載の機器。

11 . 少なくとも 1 つのプロセッサは、

1 つ以上のサブピクチャ ID が第 3 の構文内でシグナリングされることを示す第 3 のフラグをシグナリングすること

を機器に行わせるために命令を実行するように構成される、条項 8 ~ 10 の何れか一項に記載の機器。

12 . 少なくとも 1 つのプロセッサは、

1 つ以上のサブピクチャ ID が第 3 の構文内でシグナリングされることを示す第 3 のフラグをビットストリームが含むかどうかを判定すること、及び

第 3 のフラグをビットストリームが含まないことに応答して、1 つ以上のサブピクチャ ID を第 3 の構文内でシグナリングすること

30

を機器に行わせるために命令を実行するように構成される、条項 8 ~ 11 の何れか一項に記載の機器。

13 . 少なくとも 1 つのプロセッサは、

1 つ以上のサブピクチャ ID が第 1 の構文内でシグナリングされないと判定されることに応答して、1 つ以上のサブピクチャ ID を第 2 の構文及び第 3 の構文内でシグナリングすること

を機器に行わせるために命令を実行するように構成される、条項 8 ~ 12 の何れか一項に記載の機器。

14 . 少なくとも 1 つのプロセッサは、

サブピクチャ ID マッピングがビットストリーム内にあるかどうかを示す第 4 のフラグをシグナリングすること

40

を機器に行わせるために命令を実行するように構成される、条項 8 ~ 13 の何れか一項に記載の機器。

15 . 命令のセットを記憶する非一時的コンピュータ可読記憶媒体であって、命令のセットは、

サブピクチャ ID マッピングがビットストリーム内にあるかどうかを判定すること、

1 つ以上のサブピクチャ ID が第 1 の構文又は第 2 の構文内でシグナリングされるかどうかを判定すること、及び

サブピクチャ ID マッピングがあると判定され、及び 1 つ以上のサブピクチャ ID が第

50

1 の構文及び第 2 の構文内でシグナリングされないと判定されることに応答して、1 つ以上のサブピクチャ ID を第 3 の構文内でシグナリングすることを含む方法を映像処理機器に行わせるために 1 つ以上の処理装置によって実行可能である、非一時的コンピュータ可読記憶媒体。

16 . 第 1 の構文、第 2 の構文、又は第 3 の構文は、シーケンスパラメータセット (S P S)、ピクチャパラメータセット (P P S) 及びピクチャヘッダ (P H) の 1 つである、条項 15 に記載の非一時的コンピュータ可読記憶媒体。

17 . 命令のセットは、

1 つ以上のサブピクチャ ID が第 1 の構文内でシグナリングされることを示す第 1 のフラグをシグナリングすること、又は

1 つ以上のサブピクチャ ID が第 2 の構文内でシグナリングされることを示す第 2 のフラグをシグナリングすること

を映像処理機器に行わせるために 1 つ以上の処理装置によって実行可能である、条項 15 及び 16 に記載の非一時的コンピュータ可読記憶媒体。

18 . 命令のセットは、

1 つ以上のサブピクチャ ID が第 3 の構文内でシグナリングされることを示す第 3 のフラグをシグナリングすること

を映像処理機器に行わせるために 1 つ以上の処理装置によって実行可能である、条項 15 ~ 17 の何れか一項に記載の非一時的コンピュータ可読記憶媒体。

19 . 命令のセットは、

1 つ以上のサブピクチャ ID が第 3 の構文内でシグナリングされることを示す第 3 のフラグをビットストリームが含むかどうかを判定すること、及び

第 3 のフラグをビットストリームが含まないことに応答して、1 つ以上のサブピクチャ ID を第 3 の構文内でシグナリングすること

を映像処理機器に行わせるために 1 つ以上の処理装置によって実行可能である、条項 15 ~ 18 の何れか一項に記載の非一時的コンピュータ可読記憶媒体。

20 . 命令のセットは、

1 つ以上のサブピクチャ ID が第 1 の構文内でシグナリングされないと判定されることに応答して、1 つ以上のサブピクチャ ID を第 2 の構文及び第 3 の構文内でシグナリングすること

を映像処理機器に行わせるために 1 つ以上の処理装置によって実行可能である、条項 15 ~ 19 の何れか一項に記載の非一時的コンピュータ可読記憶媒体。

21 . 命令のセットは、

サブピクチャ ID マッピングがビットストリーム内にあるかどうかを示す第 4 のフラグをシグナリングすること

を映像処理機器に行わせるために 1 つ以上の処理装置によって実行可能である、条項 15 ~ 20 の何れか一項に記載の非一時的コンピュータ可読記憶媒体。

22 . 映像処理方法であって、

シーケンスパラメータセット (S P S)、ピクチャヘッダ (P H)、又はピクチャパラメータセット (P P S) の少なくとも 1 つにおいて、1 つ以上のサブピクチャ ID がシグナリングされるかどうかを判定すること、及び

1 つ以上のサブピクチャ ID が S P S、P H 及び P P S 内でシグナリングされないと判定されることに応答して、1 つ以上のサブピクチャ ID が既定値を有すると判定することを含む映像処理方法。

23 . 1 つ以上のサブピクチャ ID が P P S 内でシグナリングされるかどうかを判定する前に、1 つ以上のサブピクチャ ID が P H 内でシグナリングされるかどうかを判定することが行われる、条項 22 に記載の方法。

24 . 1 つ以上のサブピクチャ ID が P H 内でシグナリングされるかどうかを判定する前に、1 つ以上のサブピクチャ ID が P P S 内でシグナリングされるかどうかを判定することが行われる、条項 22 に記載の方法。

10

20

30

40

50

25. 映像処理方法であって、
符号化ピクチャのサブピクチャの数が1に等しいかどうかを判定すること、及び
サブピクチャの数が1に等しいと判定されることに応答して、符号化ピクチャのサブピクチャを復号化プロセス内のピクチャとして扱うこと
を含む映像処理方法。

26. サブピクチャの数が1に等しいと判定されることに応答して、インループフィルタリング演算を除外すること
をさらに含む、条項25に記載の方法。

27. 映像処理方法であって、
サブピクチャがピクチャの最後のサブピクチャかどうかを判定すること、及び
サブピクチャが最後のサブピクチャであると判定されることに応答して、サブピクチャの位置又はサイズの情報をピクチャのサイズ並びにピクチャの前のサブピクチャのサイズ及び位置から導出すること
を含む映像処理方法。

28. 映像処理方法であって、
サブピクチャがピクチャ内にあるかどうかを判定すること、及び
1つ以上のサブピクチャがピクチャ内にあると判定されることに応答して、サブピクチャIDマッピングがシーケンスパラメータセット(SPS)内にあるかどうかを示す第1のフラグをシグナリングすること
を含む映像処理方法。

29. サブピクチャIDマッピングがSPS内にあることを第1のフラグが示すことに
応答して、サブピクチャIDマッピングがSPS内でシグナリングされるかどうかを示す
第2のフラグをシグナリングすること
をさらに含む、条項28に記載の方法。

30. サブピクチャIDマッピングがSPS内でシグナリングされることを第2のフラグが示すことに応答して、1つ以上のサブピクチャのサブピクチャIDをSPS内でシグナリングすること
をさらに含む、条項29に記載の方法。

31. 映像処理機器であって、
命令を記憶するための少なくとも1つのメモリと、
少なくとも1つのプロセッサとを含み、少なくとも1つのプロセッサは、
シーケンスパラメータセット(SPS)、ピクチャヘッダ(PH)、又はピクチャパラメータセット(PPS)の少なくとも1つにおいて、1つ以上のサブピクチャIDがシグナリングされるかどうかを判定すること、及び

1つ以上のサブピクチャIDがSPS、PH及びPPS内でシグナリングされないと判定されることに応答して、1つ以上のサブピクチャIDが既定値を有すると判定することを機器に行わせるために命令を実行するように構成される、映像処理機器。

32. 少なくとも1つのプロセッサは、
1つ以上のサブピクチャIDがPPS内でシグナリングされるかどうかを判定する前に、
1つ以上のサブピクチャIDがPH内でシグナリングされるかどうかを判定すること
を機器に行わせるために命令を実行するように構成される、条項31に記載の機器。

33. 少なくとも1つのプロセッサは、
1つ以上のサブピクチャIDがPH内でシグナリングされるかどうかを判定する前に、
1つ以上のサブピクチャIDがPPS内でシグナリングされるかどうかを判定すること
を機器に行わせるために命令を実行するように構成される、条項31に記載の機器。

34. 映像処理機器であって、
命令を記憶するための少なくとも1つのメモリと、
少なくとも1つのプロセッサとを含み、少なくとも1つのプロセッサは、
符号化ピクチャのサブピクチャの数が1に等しいかどうかを判定すること、及び
サブピクチャの数が1に等しいと判定されることに応答して、符号化ピクチャのサブピ

10

20

30

40

50

クチャを復号化プロセス内のピクチャとして扱うこと

を機器に行わせるために命令を実行するように構成される、映像処理機器。

35. 少なくとも1つのプロセッサは、

サブピクチャの数が1に等しいと判定されることに応答して、インループフィルタリング演算を除外すること

を機器に行わせるために命令を実行するように構成される、条項34に記載の機器。

36. 映像処理機器であって、

命令を記憶するための少なくとも1つのメモリと、

少なくとも1つのプロセッサとを含み、少なくとも1つのプロセッサは、

サブピクチャがピクチャの最後のサブピクチャかどうかを判定すること、及び

サブピクチャが最後のサブピクチャであると判定されることに応答して、サブピクチャの位置又はサイズの情報をピクチャのサイズ並びにピクチャの前のサブピクチャのサイズ及び位置から導出すること

を機器に行わせるために命令を実行するように構成される、映像処理機器。

37. 映像処理機器であって、

命令を記憶するための少なくとも1つのメモリと、

少なくとも1つのプロセッサとを含み、少なくとも1つのプロセッサは、

サブピクチャがピクチャ内にあるかどうかを判定すること、及び

1つ以上のサブピクチャがピクチャ内にあると判定されることに応答して、サブピクチャIDマッピングがシーケンスパラメータセット(SPS)内にあるかどうかを示す第1のフラグをシグナリングすること

を機器に行わせるために命令を実行するように構成される、映像処理機器。

38. 少なくとも1つのプロセッサは、

サブピクチャIDマッピングがSPS内にあることを第1のフラグが示すことに応答して、サブピクチャIDマッピングがSPS内でシグナリングされるかどうかを示す第2のフラグをシグナリングすること

を機器に行わせるために命令を実行するように構成される、条項37に記載の機器。

39. 少なくとも1つのプロセッサは、

サブピクチャIDマッピングがSPS内でシグナリングされることを第2のフラグが示すことに応答して、1つ以上のサブピクチャのサブピクチャIDをSPS内でシグナリングすること

を機器に行わせるために命令を実行するように構成される、条項38に記載の機器。

40. 命令のセットを記憶する非一時的コンピュータ可読記憶媒体であって、命令のセットは、

シーケンスパラメータセット(SPS)、ピクチャヘッダ(PH)、又はピクチャパラメータセット(PPS)の少なくとも1つにおいて、1つ以上のサブピクチャIDがシグナリングされるかどうかを判定すること、及び

1つ以上のサブピクチャIDがSPS、PH及びPPS内でシグナリングされないことと判定されることに応答して、1つ以上のサブピクチャIDが既定値を有すると判定することを含む方法を映像処理機器に行わせるために1つ以上の処理装置によって実行可能である、非一時的コンピュータ可読記憶媒体。

41. 命令のセットは、

1つ以上のサブピクチャIDがPPS内でシグナリングされるかどうかを判定する前に、1つ以上のサブピクチャIDがPH内でシグナリングされるかどうかを判定することを映像処理機器に行わせるために1つ以上の処理装置によって実行可能である、条項40に記載の非一時的コンピュータ可読記憶媒体。

42. 命令のセットは、

1つ以上のサブピクチャIDがPH内でシグナリングされるかどうかを判定する前に、1つ以上のサブピクチャIDがPPS内でシグナリングされるかどうかを判定することを映像処理機器に行わせるために1つ以上の処理装置によって実行可能である、条項40

10

20

30

40

50

に記載の非一時的コンピュータ可読記憶媒体。

43．命令のセットを記憶する非一時的コンピュータ可読記憶媒体であって、命令のセットは、

符号化ピクチャのサブピクチャの数が1に等しいかどうかを判定すること、及び

サブピクチャの数が1に等しいと判定されることに応答して、符号化ピクチャのサブピクチャを復号化プロセス内のピクチャとして扱うこと

を含む方法を映像処理機器に行わせるために1つ以上の処理装置によって実行可能である、非一時的コンピュータ可読記憶媒体。

44．命令のセットは、

サブピクチャの数が1に等しいと判定されることに応答して、インループフィルタリング演算を除外すること

を映像処理機器に行わせるために1つ以上の処理装置によって実行可能である、条項43に記載の非一時的コンピュータ可読記憶媒体。

45．命令のセットを記憶する非一時的コンピュータ可読記憶媒体であって、命令のセットは、

サブピクチャがピクチャの最後のサブピクチャかどうかを判定すること、及び

サブピクチャが最後のサブピクチャであると判定されることに応答して、サブピクチャの位置又はサイズの情報をピクチャのサイズ並びにピクチャの前のサブピクチャのサイズ及び位置から導出すること

を含む方法を映像処理機器に行わせるために1つ以上の処理装置によって実行可能である、非一時的コンピュータ可読記憶媒体。

46．命令のセットを記憶する非一時的コンピュータ可読記憶媒体であって、命令のセットは、

サブピクチャがピクチャ内にあるかどうかを判定すること、及び

1つ以上のサブピクチャがピクチャ内にあると判定されることに応答して、サブピクチャIDマッピングがシーケンスパラメータセット(SPS)内にあるかどうかを示す第1のフラグをシグナリングすること

を含む方法を映像処理機器に行わせるために1つ以上の処理装置によって実行可能である、非一時的コンピュータ可読記憶媒体。

47．命令のセットは、

サブピクチャIDマッピングがSPS内にあることを第1のフラグが示すことに応答して、サブピクチャIDマッピングがSPS内でシグナリングされるかどうかを示す第2のフラグをシグナリングすること

を映像処理機器に行わせるために1つ以上の処理装置によって実行可能である、条項46に記載の非一時的コンピュータ可読記憶媒体。

48．命令のセットは、

サブピクチャIDマッピングがSPS内でシグナリングされることを第2のフラグが示すことに応答して、1つ以上のサブピクチャのサブピクチャIDをSPS内でシグナリングすること

を映像処理機器に行わせるために1つ以上の処理装置によって実行可能である、条項47に記載の非一時的コンピュータ可読記憶媒体。

49．映像処理方法であって、

ビットストリーム内でシグナリングされるサブピクチャ情報存在フラグに従って、ビットストリームがサブピクチャ情報を含むかどうかを判定すること、及び

ビットストリームがサブピクチャ情報を含むことに応答して、

ピクチャ内のサブピクチャの数、

ターゲットサブピクチャの幅、高さ、位置及び識別子(ID)マッピング、

subpic_treated_as_pic_flag、及び

loop_filter_across_subpic_enabled_flag

の少なくとも1つをビットストリーム内でシグナリングすること

10

20

30

40

50

を含む映像処理方法。

50．ターゲットサブピクチャの幅、高さ、及び位置の少なくとも1つのシグナリングは、ピクチャ内のサブピクチャの数に基づく、条項49に記載の方法。

51．ピクチャ内に少なくとも2つのサブピクチャがある場合、`subpic_treated_as_pic_flag`、`loop_filter_across_subpic_enabled_flag`、並びにターゲットサブピクチャの幅、高さ及び位置の少なくとも1つをシグナリングすること
をさらに含み、

ピクチャ内に1つのみのサブピクチャがある場合、ターゲットサブピクチャの幅、高さ及び位置の少なくとも1つのシグナリングは、スキップされ、

`subpic_treated_as_pic_flag`は、符号化レイヤ映像シーケンス（CLVS）内の各符号化ピクチャのサブピクチャが、インループフィルタリング演算を除外する復号化プロセス内のピクチャとして扱われるかどうかを示し、及び

`loop_filter_across_subpic_enabled_flag`は、CLVS内の各符号化ピクチャのサブピクチャの境界にわたり、サブピクチャの境界にわたるインループフィルタリング演算が可能にされるかどうかを示す、条項50に記載の方法。

52．ターゲットサブピクチャの幅がシグナリングされない場合、ターゲットサブピクチャの幅の値をピクチャの幅として決定すること、及び

ターゲットサブピクチャの高さがシグナリングされない場合、ターゲットサブピクチャの高さの値をピクチャの高さとして決定すること

をさらに含む、条項51に記載の方法。

53．ターゲットサブピクチャの幅がシグナリングされない場合、符号化ツリーブロック（CTB）サイズ単位のターゲットサブピクチャの幅の値をCTBサイズ単位のピクチャの幅として決定すること、及び

ターゲットサブピクチャの高さがシグナリングされない場合、CTBサイズ単位のターゲットサブピクチャの高さの値をCTBサイズ単位のピクチャの高さとして決定することをさらに含む、条項52に記載の方法。

54．`subpic_treated_as_pic_flag`がビットストリーム内でシグナリングされない場合、`subpic_treated_as_pic_flag`が1の値を有すると判定すること、及び

`loop_filter_across_subpic_enabled_flag`がビットストリーム内でシグナリングされない場合、`loop_filter_across_subpic_enabled_flag`が0の値を有すると判定することをさらに含む、条項51に記載の方法。

55．ターゲットサブピクチャがピクチャ内の最後のサブピクチャである場合、ターゲットサブピクチャの幅及び高さの少なくとも1つのシグナリングをスキップすること
をさらに含む、条項49に記載の方法。

56．ターゲットサブピクチャの幅がシグナリングされない場合、符号化ツリーブロック（CTB）サイズ単位のターゲットサブピクチャの幅の値を、CTBサイズ単位のピクチャの幅からCTBサイズ単位のターゲットサブピクチャの左上符号化ツリーユニット（CTU）の水平位置を引いたものとして決定するか、又はターゲットサブピクチャの幅の値を、ピクチャの幅からターゲットサブピクチャの左上符号化ツリーユニット（CTU）の水平位置を引いたものとして決定すること、及び

ターゲットサブピクチャの高さがシグナリングされない場合、CTBサイズ単位のターゲットサブピクチャの高さの値を、CTBサイズ単位のピクチャの高さからCTBサイズ単位のターゲットサブピクチャの左上CTUの垂直位置を引いたものとして決定するか、又はターゲットサブピクチャの高さの値を、ピクチャの高さからターゲットサブピクチャの左上CTUの垂直位置を引いたものとして決定すること

をさらに含む、条項55に記載の方法。

57．ターゲットサブピクチャのIDマッピングのシグナリングは、
ビットストリーム内で第1のフラグをシグナリングすること、及び

第1のフラグが1に等しいことに応答して、第1のデータ単位又は第2のデータ単位内でターゲットサブピクチャのIDマッピングをシグナリングすること、

10

20

30

40

50

をさらに含み、

0 に等しい第 1 のフラグは、ターゲットサブピクチャの ID マッピングがビットストリーム内でシグナリングされないことを示す、条項 49 に記載の方法。

58 . 第 1 のフラグが 1 に等しく、及びターゲットサブピクチャの ID マッピングが第 1 のデータ単位内でシグナリングされないことに応答して、ターゲットサブピクチャの ID マッピングを第 2 のデータ単位内でシグナリングすること、又は

第 1 のフラグが 0 に等しいか、又はターゲットサブピクチャの ID マッピングが第 1 のデータ単位内でシグナリングされることに応答して、ターゲットサブピクチャの ID マッピングを第 2 のデータ単位内でシグナリングすることをスキップすること

をさらに含む、条項 57 に記載の方法。

10

59 . 第 1 のデータ単位及び第 2 のデータ単位のそれぞれは、シーケンスパラメータセット (S P S)、ピクチャパラメータセット (P P S)、又はピクチャヘッダ (P H) の 1 つである、条項 58 に記載の方法。

60 . 映像処理機器であって、

命令を記憶するための少なくとも 1 つのメモリと、

少なくとも 1 つのプロセッサとを含み、少なくとも 1 つのプロセッサは、

ビットストリーム内でシグナリングされるサブピクチャ情報存在フラグに従って、ビットストリームがサブピクチャ情報を含むかどうかを判定すること、及び

ビットストリームがサブピクチャ情報を含むことに応答して、

ピクチャ内のサブピクチャの数、

20

ターゲットサブピクチャの幅、高さ、位置及び識別子 (I D) マッピング、

subpic_treated_as_pic_flag、及び

loop_filter_across_subpic_enabled_flag

の少なくとも 1 つをビットストリーム内でシグナリングすること

を機器に行わせるために命令を実行するように構成される、映像処理機器。

61 . ターゲットサブピクチャの幅、高さ、及び位置の少なくとも 1 つのシグナリングは、ピクチャ内のサブピクチャの数に基づく、条項 60 に記載の機器。

62 . 少なくとも 1 つのプロセッサは、

ピクチャ内に少なくとも 2 つのサブピクチャがある場合、subpic_treated_as_pic_flag、loop_filter_across_subpic_enabled_flag並びにターゲットサブピクチャの幅、高さ、及び位置の少なくとも 1 つをシグナリングすること

30

を機器に行わせるために命令を実行するように構成され、

ピクチャ内に 1 つのみのサブピクチャがある場合、ターゲットサブピクチャの幅、高さ及び位置の少なくとも 1 つのシグナリングは、スキップされ、

subpic_treated_as_pic_flagは、符号化レイヤ映像シーケンス (C L V S) 内の各符号化ピクチャのサブピクチャが、インループフィルタリング演算を除外する復号化プロセス内のピクチャとして扱われるかどうかを示し、及び

loop_filter_across_subpic_enabled_flagは、C L V S 内の各符号化ピクチャのサブピクチャの境界にわたり、サブピクチャの境界にわたるインループフィルタリング演算が可能にされるかどうかを示す、条項 61 に記載の機器。

40

63 . 少なくとも 1 つのプロセッサは、

ターゲットサブピクチャの幅がシグナリングされない場合、ターゲットサブピクチャの幅の値をピクチャの幅として決定すること、及び

ターゲットサブピクチャの高さがシグナリングされない場合、ターゲットサブピクチャの高さの値をピクチャの高さとして決定すること

を機器に行わせるために命令を実行するように構成される、条項 62 に記載の機器。

64 . 少なくとも 1 つのプロセッサは、

ターゲットサブピクチャの幅がシグナリングされない場合、符号化ツリーブロック (C T B) サイズ単位のターゲットサブピクチャの幅の値を C T B サイズ単位のピクチャの幅として決定すること、及び

50

ターゲットサブピクチャの高さがシグナリングされない場合、C T B サイズ単位のターゲットサブピクチャの高さの値をC T B サイズ単位のピクチャの高さとして決定することを機器に行わせるために命令を実行するように構成される、条項 6 3 に記載の機器。

6 5 . 少なくとも 1 つのプロセッサは、

subpic_treated_as_pic_flag がビットストリーム内でシグナリングされない場合、subpic_treated_as_pic_flag が 1 の値を有すると判定すること、及び

loop_filter_across_subpic_enabled_flag がビットストリーム内でシグナリングされない場合、loop_filter_across_subpic_enabled_flag が 0 の値を有すると判定することを機器に行わせるために命令を実行するように構成される、条項 6 2 に記載の機器。

6 6 . 少なくとも 1 つのプロセッサは、

ターゲットサブピクチャがピクチャ内の最後のサブピクチャである場合、ターゲットサブピクチャの幅及び高さの少なくとも 1 つのシグナリングをスキップすること
を機器に行わせるために命令を実行するように構成される、条項 6 0 に記載の機器。

6 7 . 少なくとも 1 つのプロセッサは、

ターゲットサブピクチャの幅がシグナリングされない場合、符号化ツリーブロック (C T B) サイズ単位のターゲットサブピクチャの幅の値を、C T B サイズ単位のピクチャの幅からC T B サイズ単位のターゲットサブピクチャの左上符号化ツリーユニット (C T U) の水平位置を引いたものとして決定するか、又はターゲットサブピクチャの幅の値を、ピクチャの幅からターゲットサブピクチャの左上符号化ツリーユニット (C T U) の水平位置を引いたものとして決定すること、及び

ターゲットサブピクチャの高さがシグナリングされない場合、C T B サイズ単位のターゲットサブピクチャの高さの値を、C T B サイズ単位のピクチャの高さからC T B サイズ単位のターゲットサブピクチャの左上C T U の垂直位置を引いたものとして決定するか、又はターゲットサブピクチャの高さの値を、ピクチャの高さからターゲットサブピクチャの左上C T U の垂直位置を引いたものとして決定すること
を機器に行わせるために命令を実行するように構成される、条項 6 6 に記載の機器。

6 8 . 命令のセットを記憶する非一時的コンピュータ可読記憶媒体であって、命令のセットは、

ビットストリーム内でシグナリングされるサブピクチャ情報存在フラグに従って、ビットストリームがサブピクチャ情報を含むかどうかを判定すること、及び

ビットストリームがサブピクチャ情報を含むことに応答して、

ピクチャ内のサブピクチャの数、

ターゲットサブピクチャの幅、高さ、位置及び識別子 (I D) マッピング、

subpic_treated_as_pic_flag、及び

loop_filter_across_subpic_enabled_flag

の少なくとも 1 つをビットストリーム内でシグナリングすること

を含む方法を映像処理機器に行わせるために 1 つ以上の処理装置によって実行可能である、非一時的コンピュータ可読記憶媒体。

【 0 1 7 5 】

[00202] いくつかの実施形態では、命令を含む非一時的コンピュータ可読記憶媒体も提供され、命令は、デバイス (本開示の符号器及び復号器など) により、上述の方法を遂行するために実行され得る。一般的な形態の非一時的媒体としては、例えば、フロッピー (登録商標) ディスク、フレキシブルディスク、ハードディスク、ソリッドステートドライブ、磁気テープ、又は任意の他の磁気データ記憶媒体、C D - R O M、任意の他の光学データ記憶媒体、孔のパターンを有する任意の物理媒体、R A M、P R O M 及び E P R O M、F L A S H (登録商標) - E P R O M、又は任意の他のフラッシュメモリ、N V R A M、キャッシュ、レジスタ、任意の他のメモリチップ、又はカートリッジ、並びにこれらのネットワーク化バージョンが挙げられる。デバイスは、1 つ以上のプロセッサ (C P U)、入力 / 出力インターフェース、ネットワークインターフェース、及び / 又はメモリを含むことができる。

【 0 1 7 6 】

[00203] 「第1」及び「第2」などの本明細書における関係語は、単に、実体又は動作を別の実体又は動作と区別するために使用されるにすぎず、これらの実体又は動作間のいかなる実際の関係又は順序も必要とせず、暗示もしないことに留意されたい。さらに、単語「含む」、「有する」、「含有する」及び「包含する」並びに他の同様の形式は、意味が均等であり、これらの単語の任意のものに続く要素若しくは要素群は、このような要素若しくは要素群の限定列挙であることを意味されないか、又は列挙された要素若しくは要素群のみに限定されることを意味されない点でオープンエンドなものであることを意図される。

【 0 1 7 7 】

[00204] 本明細書において使用するとき、別途特に断りのない限り、用語「又は」は、実行不可能な場合を除き、全ての可能な組み合わせを包含する。例えば、データベースがA又はBを含むことができると述べられた場合、このとき、別途特に断りのない限り又は実行不可能でない限り、データベースは、A若しくはB又はA及びBを含むことができる。第2の例として、データベースがA、B又はCを含むことができると述べられた場合、このとき、別途特に断りのない限り又は実行不可能でない限り、データベースは、A、B若しくはC、又はA及びB、A及びC若しくはB及びC、又はA及びB及びCを含むことができる。

【 0 1 7 8 】

[00205] 上述の実施形態は、ハードウェア若しくはソフトウェア（プログラムコード）又はハードウェア及びソフトウェアの組み合わせによって実施され得ることが理解される。ソフトウェアによって実施される場合、それは、上述のコンピュータ可読媒体内に記憶され得る。ソフトウェアは、プロセッサによって実行されたとき、本開示の方法を遂行することができる。本開示において説明される計算ユニット及び他の機能ユニットは、ハードウェア若しくはソフトウェア又はハードウェア及びソフトウェアの組み合わせによって実施され得る。当業者は、上述のモジュール／ユニットの複数のものを1つのモジュール／ユニットとして組み合わせることができ、上述のモジュール／ユニットの各々を複数のサブモジュール／サブユニットにさらに分割できることも理解するであろう。

【 0 1 7 9 】

[00206] 上述の本明細書において、実施形態は、実装形態ごとに異なり得る多くの特定の詳細を参照して説明された。上述の実施形態の特定の適応形態及び変更形態がなされ得る。本明細書の考慮及び本明細書において開示された本発明の実施から、他の実施形態が当業者に明らかになり得る。本明細書及び実施例は、例としてのみ考慮されることが意図されており、本発明の真の範囲及び趣旨は、添付の請求項によって指示される。また、図に示されるステップの配列は、単に例示目的のためのものにすぎず、ステップのいかなる特定の配列にも限定されることを意図されないことも意図される。そのため、当業者は、これらのステップが、同じ方法を実施しながらも、異なる順序で遂行され得ることを理解することができる。

【 0 1 8 0 】

[00207] 図面及び本明細書において、例示的な実施形態が開示された。しかし、これらの実施形態に対する多くの変形形態及び変更形態がなされ得る。従って、特定の用語が採用されていても、これらは、単に、一般的な説明の意味で使用されているにすぎず、限定を目的として使用されているものではない。

10

20

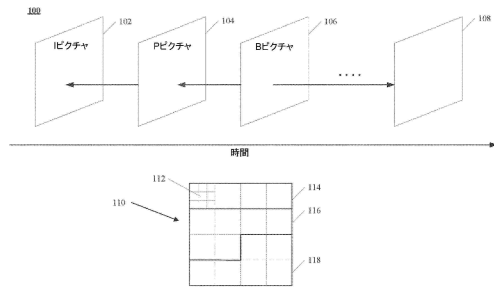
30

40

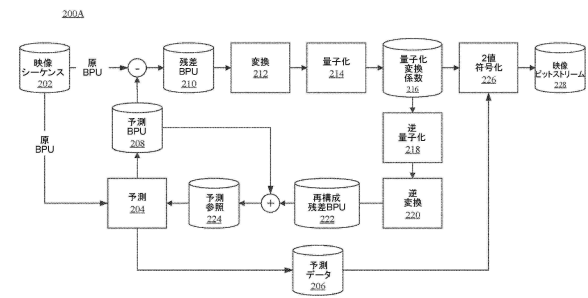
50

【図面】

【図 1】

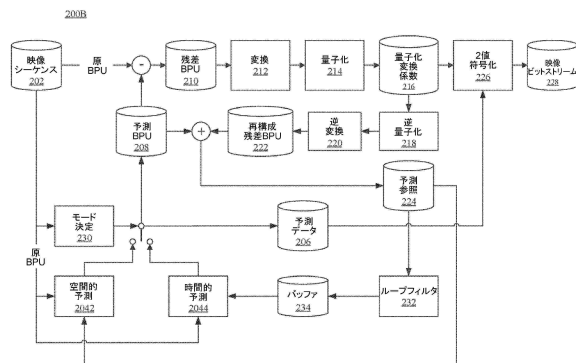


【図 2 A】

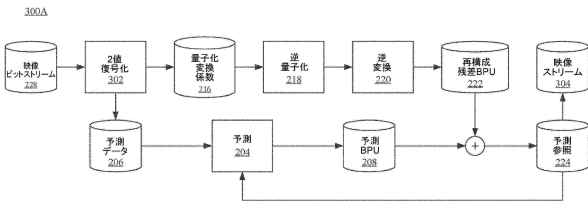


10

【図 2 B】



【図 3 A】



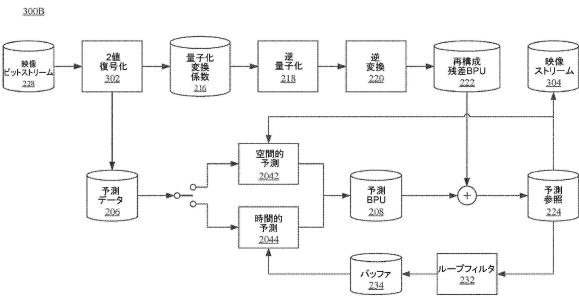
20

30

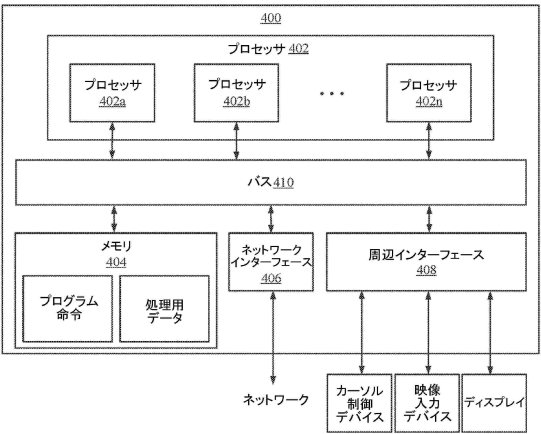
40

50

【図 3 B】

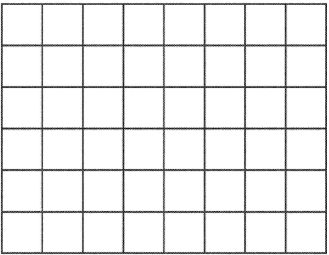


【図 4】



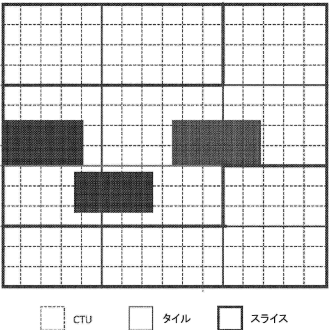
10

【図 5】



CTUに分割されたピクチャの例

【図 6】



タイル及ピラミダスキュンスライスに分割されたピクチャの例

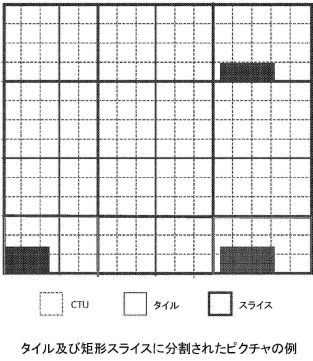
20

30

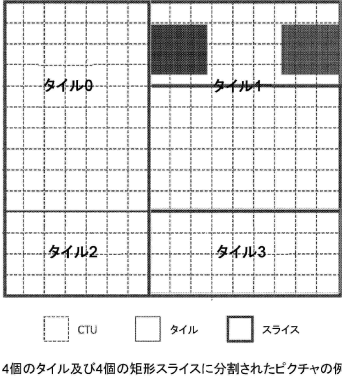
40

50

【図 7】

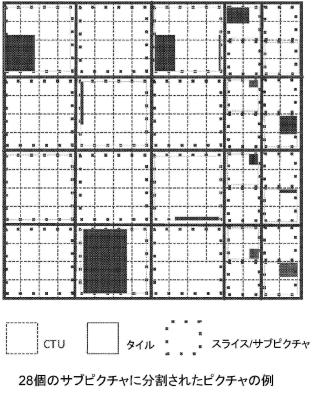


【図 8】



10

【図 9】



【図 10】

表1: サブピクチャ分割の例示的なSPS構文

seq_parameter_set_rbsp() {	記述子
...	
subpics_present_flag	u(1)
if (subpics_present_flag) {	
sps_num_subpics_minus1	u(8)
for (i = 0; i <= sps_num_subpics_minus1; i++) {	
subpic_ctu_top_left_x[i]	u(v)
subpic_ctu_top_left_y[i]	u(v)
subpic_width_minus1[i]	u(v)
subpic_height_minus1[i]	u(v)
subpic_treated_as_pic_flag[i]	u(1)
loop_filter_across_subpic_enabled_flag[i]	u(1)
}	
}	

20

30

40

50

【図 1 1】

表2:サブピクチャ識別子の例示的なSPS構文

	記述子
seq_parameter_set_rbsp() { ... sps_subpic_id_present_flag if sps_subpic_id_present_flag { sps_subpic_id_signalling_present_flag if sps_subpic_id_signalling_present_flag { sps_subpic_id_len_minus1 for (i = 0; i <= sps_num_subpics_minus1; i++) sps_subpic_id[i] } }	u(1) u(1) uc(v) u(v)

【図 1 2】

表3:サブピクチャ識別子の例示的なPPS構文

	記述子
pic_parameter_set_rbsp() { ... pps_subpic_id_signalling_present_flag if pps_subpic_id_signalling_present_flag { pps_num_subpics_minus1 pps_subpic_id_len_minus1 for (i = 0; i <= pps_num_subpics_minus1; i++) pps_subpic_id[i] }	u(1) uc(v) uc(v) u(v)

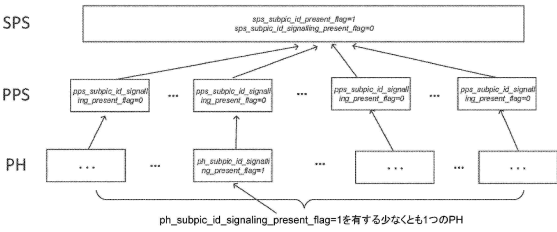
10

【図 1 3】

表4:サブピクチャ識別子の例示的なPH構文

	記述子
picture_header_rbsp() { ... if sps_subpic_id_present_flag && !sps_subpic_id_signalling_present_flag { ph_subpic_id_signalling_present_flag if (ph_subpic_id_signalling_present_flag) { ph_subpic_id_len_minus1 for (i = 0; i <= sps_num_subpics_minus1; i++) ph_subpic_id[i] } }	u(1) uc(v) u(v)

【図 1 4】



20

【図 1 5】

表5:サブピクチャ識別子の例示的なPH構文

1501		
	picture_header_rbsp() {	記述子
	...	
	if sps_subpic_id_present_flag && !sps_subpic_id_signalling_present_flag {	
	ph_subpic_id_signalling_present_flag	u(1)
	if (ph_subpic_id_signalling_present_flag) {	
	ph_subpic_id_len_minus1	uc(v)
	for (i = 0; i <= sps_num_subpics_minus1; i++)	
	ph_subpic_id[i]	u(v)
	}	
	}	

【図 1 6】

表6:サブピクチャ識別子の例示的なPH構文

1601		
	picture_header_rbsp() {	記述子
	...	
	if sps_subpic_id_present_flag && !sps_subpic_id_signalling_present_flag {	
	ph_subpic_id_signalling_present_flag	u(1)
	if (ph_subpic_id_signalling_present_flag) {	
	ph_subpic_id_len_minus1	uc(v)
	for (i = 0; i <= sps_num_subpics_minus1; i++)	
	ph_subpic_id[i]	u(v)
	}	
	}	

30

40

50

【図 17 A】

表7A: 例示的なSPS構文

	seq_parameter_set_rbsp() {	記述子
	...	
	subpics_present_flag	u(1)
	if(subpics_present_flag) {	
	sps_num_subpics_minus1	u(8)
1701	if(sps_num_subpics_minus1 > 0) {	
	for(i = 0; i <= sps_num_subpics_minus1; i++) {	
	subpic_ctu_top_left_x[i]	u(v)
	subpic_ctu_top_left_y[i]	u(v)
	subpic_width_minus1[i]	u(v)
	subpic_height_minus1[i]	u(v)
	subpic_treated_as_pic_flag[i]	u(1)
	loop_filter_across_subpic_enabled_flag[i]	u(1)
	}	
1702	}	
	}	

【図 17 B】

表7B: 例示的なSPS構文

	seq_parameter_set_rbsp() {	記述子
	...	
	subpics_present_flag	u(1)
	if(subpics_present_flag) {	
	sps_num_subpics_minus1	u(8)
1711	for(i = 0; i <= sps_num_subpics_minus1; i++) {	
	if(sps_num_subpics_minus1 > 0) {	
	subpic_ctu_top_left_x[i]	u(v)
	subpic_ctu_top_left_y[i]	u(v)
	subpic_width_minus1[i]	u(v)
	subpic_height_minus1[i]	u(v)
	subpic_treated_as_pic_flag[i]	u(1)
	loop_filter_across_subpic_enabled_flag[i]	u(1)
1712	}	
	}	
	}	
	...	

10

【図 18】

表8: 例示的なSPS構文

	seq_parameter_set_rbsp() {	記述子
	...	
	subpics_present_flag	u(1)
	if(subpics_present_flag) {	
	sps_num_subpics_minus1	u(8)
1801	for(i = 0; i <= sps_num_subpics_minus1; i++) {	
	if(sps_num_subpics_minus1 > 0) {	
	subpic_ctu_top_left_x[i]	u(v)
	subpic_ctu_top_left_y[i]	u(v)
	subpic_width_minus1[i]	u(v)
	subpic_height_minus1[i]	u(v)
1802	}	
	subpic_treated_as_pic_flag[i]	u(1)
	loop_filter_across_subpic_enabled_flag[i]	u(1)
	}	
	}	

【図 19】

表9: 例示的なSPS構文

	seq_parameter_set_rbsp() {	記述子
	...	
	subpics_present_flag	u(1)
	if(subpics_present_flag) {	
	sps_num_subpics_minus1	u(8)
1901	for(i = 0; i <= sps_num_subpics_minus1; i++) {	
	subpic_ctu_top_left_x[i]	u(v)
	subpic_ctu_top_left_y[i]	u(v)
	if(sps_num_subpics_minus1 > 0) {	
	subpic_width_minus1[i]	u(v)
	subpic_height_minus1[i]	u(v)
1902	}	
	subpic_treated_as_pic_flag[i]	u(1)
	loop_filter_across_subpic_enabled_flag[i]	u(1)
	}	
	}	

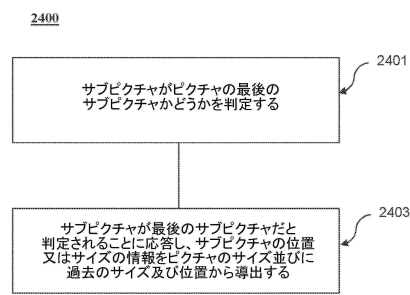
20

30

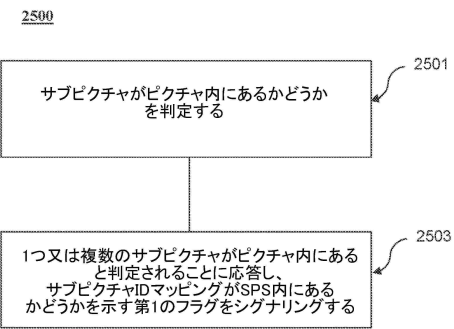
40

50

【図 2 4】



【図 2 5】



10

20

30

40

50

フロントページの続き

- (72)発明者 チェン, ジエ
中華人民共和国, 베이징ン チャオヤン ディストリクト ザ フォース ディストリクト ワンジン
ドン ユアン ビルディング ナンバー 6
- (72)発明者 イエ, ヤン
アメリカ合衆国, カリフォルニア州 9 2 1 3 0 サンディエゴ パールマン ウェイ 5 0 0 1
- (72)発明者 ルオ, ジャンコン
アメリカ合衆国, ニュージャージー州 0 8 5 5 8 スキルマン リッチモンド ドライブ 2 8
- (72)発明者 リャオ, ル - リン
中華人民共和国, 베이징ン チャオヤン ディストリクト ザ フォース ディストリクト ワンジン
ドン ユアン ラディアンス ジン ファイ タワー
- 審査官 田中 純一
- (56)参考文献 国際公開第 2 0 1 9 / 2 4 3 5 3 9 (WO , A 1)
特表 2 0 2 1 - 5 2 8 0 0 3 (J P , A)
特表 2 0 2 2 - 5 5 3 5 9 9 (J P , A)
特表 2 0 2 2 - 5 1 5 9 9 2 (J P , A)
Benjamin Bross, et al. , Versatile Video Coding (Draft 7) [online] , JVET-P JVET-P2001-v
E , ITU-T インターネット < [https://jvet-experts.org/doc_end_user/documents/16_Geneva/
wg11/JVET-P2001-v14.zip](https://jvet-experts.org/doc_end_user/documents/16_Geneva/wg11/JVET-P2001-v14.zip) > < JVET-P2001-vE.docx > , 2019年11月14日 , pp.i-v, pp.1-46
4
- (58)調査した分野 (Int.Cl. , D B 名)
H 0 4 N 7 / 1 2
H 0 4 N 1 9 / 0 0 - 1 9 / 9 8