

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7680453号
(P7680453)

(45)発行日 令和7年5月20日(2025.5.20)

(24)登録日 令和7年5月12日(2025.5.12)

(51)国際特許分類

F I

H 0 4 N 19/70 (2014.01)

H 0 4 N 19/105(2014.01)

H 0 4 N 19/174(2014.01)

H 0 4 N 19/70

H 0 4 N 19/105

H 0 4 N 19/174

請求項の数 13 (全45頁)

(21)出願番号	特願2022-537463(P2022-537463)	(73)特許権者	511050697
(86)(22)出願日	令和2年12月22日(2020.12.22)		アリババ グループ ホウルディング リ
(65)公表番号	特表2023-508895(P2023-508895		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6日(2023.3.6)		ジョージ タウン ピーオーボックス 8
(86)国際出願番号	PCT/US2020/066569		4 7 ワン キャピタル プレイス フォー
(87)国際公開番号	WO2021/138143		ス フロア
(87)国際公開日	令和3年7月8日(2021.7.8)	(74)代理人	100079108
審査請求日	令和5年10月18日(2023.10.18)		弁理士 稲葉 良幸
(31)優先権主張番号	62/954,828	(74)代理人	100109346
(32)優先日	令和1年12月30日(2019.12.30)		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】 仮想境界及びラップアラウンド動き補償をシグナリングするための方法

(57)【特許請求の範囲】

【請求項 1】

映像復号化方法であって、
ピクチャのセットに関連するビットストリームを受信すること、
前記受信されたビットストリームに従い、前記ピクチャのセット内の第1のピクチャの解像度が、前記第1のピクチャに関連する参照ピクチャの解像度と異なるかどうかを判定すること、
前記第1のピクチャの前記解像度が、前記第1のピクチャに関連する前記参照ピクチャの前記解像度と異なることに応答して、前記第1のピクチャに関してラップアラウンド動き補償が無効にされると判定すること、
前記受信されたビットストリームに従い、前記ピクチャのセットに関してシーケンスレベルで仮想境界がシグナリングされるかどうかを判定すること、及び
前記シーケンスレベルで前記仮想境界がシグナリングされるかどうかに基づいて、インループフィルタリング演算を制御すること
を含み、
前記インループフィルタリング演算を前記制御することは、
前記シーケンスレベルで前記仮想境界がシグナリングされることに応答して、前記ピクチャのセットに関する前記仮想境界の位置を決定することであって、前記位置は、前記受信されたビットストリーム内でシグナリングされる範囲によって制限されること、及び
前記仮想境界を横断するインループフィルタリング演算を無効にすること

を含み、

前記位置が制限される前記範囲は、垂直範囲又は水平範囲の少なくとも1つを含み、

前記垂直範囲は、前記ピクチャのセットの各ピクチャの最大幅に関連する第1の値以下であり、前記最大幅は、前記受信されたビットストリーム内でシグナリングされ、及び

前記水平範囲は、前記ピクチャのセットの各ピクチャの最大高さに関連する第2の値以下であり、前記最大高さは、前記受信されたビットストリーム内でシグナリングされる、方法。

【請求項2】

前記第1の値は、 $\text{Ceil}(\text{pic_width_max_in_luma_samples} \div 8) - 1$ に等しく、前記第2の値は、 $\text{Ceil}(\text{pic_height_max_in_luma_samples} \div 8) - 1$ に等しく、

$\text{pic_width_max_in_luma_samples}$ は、前記ピクチャのセットの各ピクチャのルマサンプル単位の前記最大幅を表し、 $\text{pic_height_max_in_luma_samples}$ は、前記ピクチャのセットの各ピクチャのルマサンプル単位の前記最大高さを表す、請求項1に記載の方法。

【請求項3】

前記受信されたビットストリームに従い、前記シーケンスレベルで前記仮想境界がシグナリングされるかどうかを判定することは、

前記受信されたビットストリームに従い、前記第1のピクチャの前記解像度の変更が許可されるかどうかを判定すること、及び

前記第1のピクチャの前記解像度の変更が許可されることに応答して、前記シーケンスレベルで前記仮想境界がシグナリングされないことと判定すること

を含む、請求項1に記載の方法。

【請求項4】

前記第1のピクチャの前記解像度の変更が許可されることに応答して、前記ピクチャのセットに関してラップアラウンド動き補償が無効にされると判定すること

をさらに含む、請求項3に記載の方法。

【請求項5】

ピクチャのセットのビットストリームを記憶する方法であって、前記方法は

前記ピクチャのセットの仮想境界を設定すること、

前記仮想境界を横断するインループフィルタリング演算を無効にすること、

前記ピクチャのセット内の第1のピクチャの解像度が、前記第1のピクチャに関連する参照ピクチャの解像度と異なるかどうかを判定すること、

前記第1のピクチャの前記解像度が、前記第1のピクチャに関連する前記参照ピクチャの前記解像度と異なることに応答して、前記第1のピクチャに関してラップアラウンド動き補償が無効にすること、

前記仮想境界の位置が制限される範囲を示す最大値をシグナリングする符号化された情報を含むビットストリームを生成すること、及び

前記ビットストリームを非一時的コンピュータ可読媒に記憶すること

を含み、

前記範囲は、垂直範囲又は水平範囲の少なくとも1つを含み、前記最大値は、前記ピクチャのセットの各ピクチャの最大幅と、前記ピクチャのセットの各ピクチャの最大高さとのうち少なくとも1つを含み、

前記垂直範囲は、前記最大幅に関連する第1の値以下であり、

前記水平範囲は、前記最大高さに関連する第2の値以下である、方法。

【請求項6】

前記第1の値は、 $\text{Ceil}(\text{pic_width_max_in_luma_samples} \div 8) - 1$ に等しく、前記第2の値は、 $\text{Ceil}(\text{pic_height_max_in_luma_samples} \div 8) - 1$ に等しく、

$\text{pic_width_max_in_luma_samples}$ は、前記ピクチャのセットの各ピクチャのルマサンプル単位の前記最大幅を表し、 $\text{pic_height_max_in_luma_samples}$ は、前記ピクチャのセットの各ピクチャのルマサンプル単位の前記最大高さを表す、請求項5に記載の方法。

【請求項 7】

前記方法は、

前記第 1 のピクチャの前記解像度の変更が許可されるかどうかを示すフラグを判定することをさらに含み、

前記第 1 のピクチャの前記解像度の変更が許可される場合、前記符号化された情報は、前記ピクチャのセットのシーケンスレベルで前記仮想境界をシグナリングせず、
前記第 1 のピクチャの前記解像度の変更が許可されない場合、前記符号化された情報は、前記シーケンスレベルで前記仮想境界をシグナリングする、請求項 5 に記載の方法。

【請求項 8】

前記方法は、

前記第 1 のピクチャの前記解像度の変更が許可されることに~~応答して~~、前記ピクチャのセットに関して前記ラップアラウンド動き補償を無効にすること
をさらに含む、請求項 7 に記載の方法。

【請求項 9】

前記第 1 のピクチャの前記解像度の変更が許可される場合、前記符号化された情報は、記セット内の 1 つ以上のピクチャに関してピクチャレベルで前記仮想境界をシグナリングする、請求項 7 に記載の方法。

【請求項 10】

映像符号化方法であって、

ピクチャのセットに関する仮想境界を設定すること、

前記仮想境界を横断するインループフィルタリング演算を無効にすること

前記ピクチャのセット内の第 1 のピクチャの解像度が、前記第 1 のピクチャに関連する参照ピクチャの解像度と異なるかどうかを判定すること、

前記第 1 のピクチャの前記解像度が、前記第 1 のピクチャに関連する前記参照ピクチャの前記解像度と異なることに~~応答して~~、前記第 1 のピクチャに関してラップアラウンド動き補償が無効にすること、

ビットストリーム内で、前記仮想境界の位置を制限する範囲を指示する最大値をシグナリングすること、
を含み、

前記範囲は、垂直範囲又は水平範囲の少なくとも 1 つを含み、前記最大値は、前記ピクチャのセットの各ピクチャの最大幅又は前記ピクチャのセットの各ピクチャの最大高さの少なくとも 1 つを含み、

前記垂直範囲は、前記最大幅に関連する第 1 の値以下であり、

前記水平範囲は、前記最大高さに関連する第 2 の値以下である、

方法。

【請求項 11】

前記第 1 の値は、 $\text{Ceil}(\text{pic_width_max_in_luma_samples} \div 8) - 1$ に等しく、前記第 2 の値は、 $\text{Ceil}(\text{pic_height_max_in_luma_samples} \div 8) - 1$ に等しく、

$\text{pic_width_max_in_luma_samples}$ は、前記ピクチャのセットの各ピクチャのルマサンプル単位の前記最大幅を表し、 $\text{pic_height_max_in_luma_samples}$ は、前記ピクチャのセットの各ピクチャのルマサンプル単位の前記最大高さを表す、請求項 10 に記載の方法。

【請求項 12】

前記第 1 のピクチャの前記解像度の変更が許可されるかどうかを判定すること、

前記第 1 のピクチャの前記解像度の変更が許可されないことに~~応答して~~、前記ピクチャのセットのシーケンスレベルで前記仮想境界をシグナリングすること

前記第 1 のピクチャの前記解像度の変更が許可されることに~~応答して~~、前記シーケンスレベルで前記仮想境界を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、請求項 10 に記載の方法。

【請求項 13】

前記第 1 のピクチャの前記解像度の変更が許可されることに応答して、前記ピクチャのセットに関してラップアラウンド動き補償を無効にすること
をさらに含む、請求項 1 2 に記載の方法。

【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

[001] 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30日に出願された米国仮特許出願第62/954,828号に対する優先権を主張する。

【 0 0 0 2 】

10

技術分野

[002] 本開示は、概して、映像処理に関し、より詳細には、仮想境界及びラップアラウンド動き補償をシグナリングするための方法に関する。

【背景技術】

【 0 0 0 3 】

背景

[003] 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（HEVC/H.265）規格、多用途ビデオコーディング（VVC/H.266）規格、及びAVS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。

20

【発明の概要】

【課題を解決するための手段】

【 0 0 0 4 】

[004] 本開示は、ピクチャ処理のための方法を提供する。この方法は、ピクチャのセットを含むビットストリームを受信すること、受信されたビットストリームに従い、ピクチャのセットに関してシーケンスレベルで仮想境界がシグナリングされるかどうかを判定すること、シーケンスレベルで仮想境界がシグナリングされることに応答して、ピクチャのセットに関する仮想境界の位置を決定することであって、位置は、受信されたビットストリーム内でシグナリングされる範囲によって制限されること、及び仮想境界を横断するインループフィルタリング演算を無効にすることを含み得る。

30

【 0 0 0 5 】

[005] 本開示の実施形態は、ピクチャ処理のための機器をさらに提供する。この機器は、命令のセットを記憶するメモリと、1つ以上のプロセッサとを含み得、1つ以上のプロセッサは、ピクチャのセットを含むビットストリームを受信すること、受信されたビットストリームに従い、ピクチャのセットに関してシーケンスレベルで仮想境界がシグナリングされるかどうかを判定すること、シーケンスレベルで仮想境界がシグナリングされることに応答して、ピクチャのセットに関する仮想境界の位置を決定することであって、位置は、受信されたビットストリーム内でシグナリングされる範囲によって制限されること、及び仮想境界を横断するインループフィルタリング演算を無効にすることを機器に行わせるために命令のセットを実行するように構成される。

40

【 0 0 0 6 】

[006] 本開示の実施形態は、命令のセットを記憶する非一時的コンピュータ可読媒体をさらに提供し、命令のセットは、ピクチャ処理方法をコンピュータに行わせるためにコンピュータの少なくとも1つのプロセッサによって実行可能であり、当該方法は、ピクチャ

50

のセットを含むビットストリームを受信すること、受信されたビットストリームに従い、ピクチャのセットに関してシーケンスレベルで仮想境界がシグナリングされるかどうかを判定すること、シーケンスレベルで仮想境界がシグナリングされることに応答して、ピクチャのセットに関する仮想境界の位置を決定することであって、位置は、受信されたビットストリーム内でシグナリングされる範囲によって制限されること、及び仮想境界を横断するインループフィルタリング演算を無効にすることを含む。

【 0 0 0 7 】

図面の簡単な説明

【 007 】 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。

10

【図面の簡単な説明】

【 0 0 0 8 】

【 図 1 】 【 008 】 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。

【 図 2 A 】 【 009 】 本開示のいくつかの実施形態に従う、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 】 【 010 】 本開示のいくつかの実施形態に従う、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。

【 図 3 A 】 【 011 】 本開示のいくつかの実施形態に従う、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。

20

【 図 3 B 】 【 012 】 本開示のいくつかの実施形態に従う、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 】 【 013 】 本開示のいくつかの実施形態に従う、映像を符号化又は復号化するための例示的な機器のブロック図を示す。

【 図 5 】 【 014 】 本開示のいくつかの実施形態に従う、仮想境界をシグナリングするため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。

【 図 6 】 【 015 】 本開示のいくつかの実施形態に従う、仮想境界をシグナリングするための例示的なピクチャヘッダ (P H) 構文を示す。

【 図 7 A 】 【 016 】 本開示のいくつかの実施形態に従う、正距円筒図法による投影 (E R P) のための例示的な水平ラップアラウンド動き補償を示す。

30

【 図 7 B 】 【 017 】 本開示のいくつかの実施形態に従う、パデッド E R P (P E R P) のための例示的な水平ラップアラウンド動き補償を示す。

【 図 8 】 【 018 】 本開示のいくつかの実施形態に従う、ラップアラウンド動き補償のための例示的な構文を示す。

【 図 9 】 【 019 】 本開示のいくつかの実施形態に従う、ピクチャの幅及び高さの最大値をシグナリングするための例示的な S P S 構文を示す。

【 図 1 0 】 【 020 】 本開示のいくつかの実施形態に従う、ラップアラウンド動き補償をシグナリングするための例示的な構文を示す。

【 図 1 1 】 【 021 】 本開示のいくつかの実施形態に従う、仮想境界をシグナリングするための例示的な S P S 構文を示す。

40

【 図 1 2 】 【 022 】 本開示のいくつかの実施形態に従う、仮想境界をシグナリングするための別の例示的な S P S 構文を示す。

【 図 1 3 】 【 023 】 本開示のいくつかの実施形態に従う、ラップアラウンド動き補償をシグナリングするための例示的な S P S 構文を示す。

【 図 1 4 】 【 024 】 本開示のいくつかの実施形態に従う、ラップアラウンド動き補償をシグナリングするための別の例示的な S P S 構文を示す。

【 図 1 5 】 【 025 】 本開示のいくつかの実施形態に従う、ピクチャ処理のための例示的な方法を示す。

【発明を実施するための形態】

【 0 0 0 9 】

50

詳細な説明

[026] 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、本開示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開示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で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。

【0010】

10

[027] 上述のとおり、映像は、視覚情報を記憶するために時系列で配列されたフレームである。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、一部の用途では、映像取り込みデバイスは、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）にリアルタイムで伝送することができる。

【0011】

[028] 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（DSP）、特定用途向け集積回路（ASIC）、フィールドプログラマブルゲートアレイ（FPGA）、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。一部の用途では、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。

20

30

【0012】

[029] 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。さもないければ、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

【0013】

[030] 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。

【0014】

50

【031】 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。

【0015】

【032】 半分の帯域幅を用いてHEVC/H.265と同じ主観的品質を達成するために、JVEは、共同探索モデル（JEM）参照ソフトウェアを用いて、HEVCを超える技術を開発している。符号化技術がJEMに組み込まれたため、JEMは、HEVCよりも実質的に高い符号化性能を達成した。

10

【0016】

【033】 VVC規格は、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引き続き含んでいる。VVCは、HEVC、H.264/AVC、MPEG2、H.263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。360度映像などのアプリケーションでは、特定の投影フォーマットのレイアウトが複数の面を通常有する。例えば、MPEG-Iパート2：全方向性メディアフォーマット（OMAF）は、面を6つ有するCMPと名付けられたキューブマップベースの投影フォーマットを規格化する。複数の面を含む投影フォーマットに関して、どのような種類のコンパクトフレームパッキング構成が使用されるかに関係なく、フレームパックピクチャ内の2つ以上の隣接する面間に不連続性が生じる。この不連続性にわたってインループフィルタリング演算を実行した場合、レンダリング後の再構成映像内で面の継ぎ目のアーティファクトが見えるようになり得る。面の継ぎ目のアーティファクトを軽減するために、フレームパックピクチャ内の不連続性にわたるインループフィルタリング演算を無効にしなければならない。仮想境界を横断するインループフィルタリングを無効にしてアーティファクトを防ぐために、仮想境界は、360度映像のための符号化ツールの1つとして設定され得る。

20

【0017】

【034】 図1は、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。

30

【0018】

【035】 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。一部の実施形態では、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。

40

【0019】

【036】 典型的には、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

50

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。一部の実施形態では、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随意的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。

【0020】

[037] 基本処理ユニットは、コンピュータメモリ内(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)、及び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。

【0021】

[038] 映像符号化は、複数の演算段階を有し、図2A~2B及び3A~3Bにその例が詳細に示される。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。一部の実施形態では、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(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できることに留意されたい。

【0022】

[039] 例えば、モード決定段階(図2Bにその例が詳細に示される)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。

【0023】

10

20

30

40

50

[040] 別の例として、予測段階（図2A～2Bにその例が詳細に示される）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。

【0024】

[041] 別の例として、変換段階（図2A～2Bにその例が詳細に示される）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。

【0025】

[042] 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。

【0026】

[043] 一部の実装形態では、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らしために、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、そうすることで、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、コーデックは、破損した又は失われたデータを依存することなく、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、そうすることで、誤り耐性の能力をもたらし。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し得ることに留意されたい。

【0027】

[044] 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。

【0028】

[045] 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。一部の実施形態では、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 を反復的な方法で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。一部の実施形態では、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。

【0029】

[046] 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 BPU 208 を生成することができる。符号器は、予測 BPU 208 を原 BPU から減算し、残差 BPU 210 を生成することができる。符号器は、残差 BPU 210 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 中、量子化段階 214 後、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 BPU 222 を生成することができる。符号器は、再構成残差 BPU 222 を予測 BPU 208 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。

【0030】

[047] 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。

【0031】

[048] 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方法による任意の行為を指すことができる。

【0032】

[049] 予測段階 204 では、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。

【0033】

[050] 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 B

P Uを再構成するために用いられ得る。そのため、原B P Uは、圧縮される。

【0034】

[051] 残差B P U 2 1 0をさらに圧縮するために、変換段階2 1 2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B P U 2 1 0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B P U 2 1 0のサイズ）を有することができる。各基底パターンは、残差B P U 2 1 0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。基底パターンのいずれも、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B P U 2 1 0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。

10

【0035】

[052] 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2 1 2において用いることができる。変換段階2 1 2における変換は、逆演算可能である。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。

20

【0036】

[053] 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、これにより、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。

30

40

【0037】

[054] 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。典型的には、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は、少ないビットを必要とし得る。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。

【0038】

[055] 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

50

06及び量子化変換係数216を符号化することができる。一部の実施形態では、予測データ206及び量子化変換係数216の他に、符号器は、例えば、予測段階204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212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226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226の出力データを用いて映像ビットストリーム228を生成することができる。一部の実施形態では、映像ビットストリーム228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。

【0039】

10

[056] プロセス200Aの再構成経路を参照すると、逆量子化段階218において、符号器は、量子化変換係数216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220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の次の反復において用いられることになる予測基準224を生成することができる。

【0040】

[057] 映像シーケンス202を符号化するために、プロセス200Aの他の変形形態を用いることもできることに留意されたい。一部の実施形態では、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。一部の実施形態では、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。一部の実施形態では、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。一部の実施形態では、プロセス200Aは、追加の段階を含むことができる。一部の実施形態では、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。

20

【0041】

[058] 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。

30

【0042】

[059] 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。

40

【0043】

[060] 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

50

る。符号器は、隣接 BPU を外挿することによって予測 BPU 208 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。一部の実施形態では、符号器は、予測 BPU 208 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。

10

【0044】

[061] 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。一部の実施形態では、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 BPU 222 を予測 BPU 208 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の再構成 BPU 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（「探索窓」と称される）範囲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 BPU 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 BPU 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に拡張され得る。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 BPU 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 BPU と異なる（例えば、原 BPU 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 BPU の場所に「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。（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、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。一部の実施形態では、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。

20

30

【0045】

[062] 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。

40

【0046】

[063] 予測 BPU 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 BPU 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、動きベクトルに従い、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 BPU 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、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

50

域のピクセル値を平均することができる。一部の実施形態では、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。

【0047】

[064] 一部の実施形態では、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。

10

【0048】

[065] 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、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候補予測モ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BPU208及び予測データ206を生成することができる。

20

【0049】

[066] プロセス200B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224(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BPU)を生成した後、符号器は、予測基準224を後の使用のために(例えば、現在のピクチャの次のBPUの外挿のために)空間的予測段階2042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224(例えば、全てのBPU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224をループフィルタ段階232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224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232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202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234(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2044において用いるためにバッファ234内に記憶することができる。一部の実施形態では、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216、予測データ206、及び他の情報と共に、2値符号化段階226において符号化することができる。

30

40

【0050】

[067] 図3Aは、本開示の実施形態に従う、例示的な復号化プロセス300Aの概略図を示す。プロセス300Aは、図2Aにおける圧縮プロセス200Aに対応する復元プロセスであり得る。一部の実施形態では、プロセス300Aは、プロセス200Aの再構成経路と似たものであり得る。復号器は、プロセス300Aに従って映像ビットストリーム228を映像ストリーム304に復号化することができる。映像ストリーム304は、映像シーケンス202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2A~2Bにおける量子化段階214)における情報損失のため、概して、映像ストリーム304は、映像シーケンス202と同一ではない。図2A~2Bにおけるプロセス200A及び200Bと同様に、復号器は、映像ビットストリーム228内に符号化され

50

たピクチャごとに基本処理ユニット（ＢＰＵ）のレベルでプロセス３００Ａを遂行することができる。例えば、復号器は、プロセス３００Ａを反復的な方法で遂行することができ、その場合、復号器は、基本処理ユニットをプロセス３００Ａの１回の反復において復号化することができる。一部の実施形態では、復号器は、プロセス３００Ａを、映像ビットストリーム２２８内に符号化された各ピクチャの領域（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。

【００５１】

[068] 図３Ａ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ＢＰＵ」と称される）に関連付けられた映像ビットストリーム２２８の部分で２値復号化段階３０２に供給することができる。２値復号化段階３０２において、復号器は、当該部分を予測データ２０６及び量子化変換係数２１６に復号化することができる。復号器は、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。復号器は、予測データ２０６を予測段階２０４に供給し、予測ＢＰＵ２０８を生成することができる。復号器は、再構成残差ＢＰＵ２２２を予測ＢＰＵ２０８に加算し、予測基準２２４を生成することができる。一部の実施形態では、予測基準２２４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３００Ａの次の反復において遂行するために予測基準２２４を予測段階２０４に供給することができる。

【００５２】

[069] 復号器は、符号化ピクチャの各符号化ＢＰＵを復号化し、符号化ピクチャの次の符号化ＢＰＵを符号化するための予測基準２２４を生成するために、プロセス３００Ａ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ＢＰＵ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３０４に出力し、映像ビットストリーム２２８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。

【００５３】

[070] ２値復号化段階３０２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。一部の実施形態では、予測データ２０６及び量子化変換係数２１６の他に、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３０２において復号化することができる。一部の実施形態では、映像ビットストリーム２２８がネットワークを通してパケットの形式で伝送される場合、復号器は、映像ビットストリーム２２８を、それを２値復号化段階３０２に供給する前にデパケット化することができる。

【００５４】

[071] 図３Ｂ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３００Ｂの概略図を示す。プロセス３００Ｂは、プロセス３００Ａから変更され得る。例えば、プロセス３００Ｂは、ハイブリッド映像符号化規格（例えば、Ｈ．２６×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３００Ａと比べて、プロセス３００Ｂは、予測段階２０４を空間的予測段階２０４２及び時間的予測段階２０４４に追加的に分割し、ループフィルタ段階２３２及びバッファ２３４を追加的に含む。

【００５５】

[072] プロセス３００Ｂ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ＢＰＵ」と称される）のために、復号器によって２値復号化段階３０２から復号化された予測データ２０６は、いかなる予測モードが符号器によって現在のＢＰＵ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ＢＰＵを符号化するために用いられた場合、予測データ２０６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BPUの場所(例えば、座標)、隣接BPUのサイズ、外挿のパラメータ、原BPUに対する隣接BPU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BPUを符号化するために用いられた場合、予測データ206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BPU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。

10

【0056】

[073] 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2042において空間的予測(例えば、イントラ予測)を遂行すべきか、又は時間的予測段階2044において時間的予測(例えば、インター予測)を遂行す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2Bにおいて説明されており、以下で繰り返されない。このような空間的予測又は時間的予測を遂行した後、復号器は、予測BPU208を生成することができる。復号器は、図3Aにおいて説明されたように、予測BPU208及び再構成残差BPU222を加算し、予測基準224を生成することができる。

【0057】

20

[074] プロセス300Bにおいて、復号器は、予測演算をプロセス300Bの次の反復において遂行するために、予測基準224を空間的予測段階2042又は時間的予測段階2044に供給することができる。例えば、現在のBPUが空間的予測段階2042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224(例えば、復号化された現在のBPU)を生成した後、復号器は、予測基準224を後の使用のために(例えば、現在のピクチャの次のBPUの外挿のために)空間的予測段階2042に直接供給することができる。現在のBPUが時間的予測段階2044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224(例えば、全てのBPUが復号化された参照ピクチャ)を生成した後、符号器は、予測基準224をループフィルタ段階232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2Bにおいて説明されたとおりの方法でループフィルタを予測基準224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228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234(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2044において用いられるためにバッファ234内に記憶することができる。一部の実施形態では、予測データ206の予測モードインジケータが、現在のBPUを符号化するためにインター予測が用いられたことを指示するとき、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。

30

【0058】

40

[075] 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400のブロック図である。図4に示されるように、機器400は、プロセッサ402を含むことができる。プロセッサ402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400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402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402は、中央処理装置(又は「CPU」)、グラフィック処理装置(又は「GPU」)、ニューラル処理装置(「NPU」)、マイクロコントローラユニット(「MCU」)、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(IP)コア、プログラマブル論理アレイ(PLA)、プログラマブルアレイ論理(PAL)、ジェネリック

50

アレイ論理（GAL）、複合プログラマブル論理装置（CPLD）、フィールドプログラマブルゲートアレイ（FPGA）、システムオンチップ（SoC）、特定用途向け集積回路（ASIC）、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。一部の実施形態では、プロセッサ402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402は、プロセッサ402a、プロセッサ402b、及びプロセッサ402nを含む複数のプロセッサを含むことができる。

【0059】

[076] 機器400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404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200A、200B、300A、又は300Bにおける段階を実施するためのプログラム命令）、及び処理のためのデータ（例えば、映像シーケンス202、映像ビットストリーム228、又は映像ストリーム304）を含むことができる。プロセッサ402は、（例えば、バス410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。一部の実施形態では、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。

【0060】

[077] 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。

【0061】

[078] 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。

【0062】

[079] 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。一部の実施形態では、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。

【0063】

[080] 一部の実施形態では、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4A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

10

20

30

40

50

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。

【0064】

[081] 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。

10

【0065】

[082] 開示する実施形態によれば、360度映像又は漸進的復号化リフレッシュ（GDR）を符号化するためにいくつかの符号化ツールが使用され得る。仮想境界は、これらの符号化ツールの1つである。360度映像などのアプリケーションでは、特定の投影フォーマットのレイアウトが複数の面を通常有する。例えば、MPEG-Iパート2：全方向性メディアフォーマット（OMAF）は、面を6つ有するCMPと名付けられたキューブマップベースの投影フォーマットを規格化する。複数の面を含む投影フォーマットに関して、どのような種類のコンパクトフレームパッキング構成が使用されるかに関係なく、フレームパックピクチャ内の2つ以上の隣接する面間に不連続性が生じる。この不連続性にわたってインループフィルタリング演算を実行した場合、レンダリング後の再構成映像内で面の継ぎ目のアーティファクトが見えるようになり得る。面の継ぎ目のアーティファクトを軽減するために、フレームパックピクチャ内の不連続性にわたるインループフィルタリング演算を無効にしなければならない。従って、開示する実施形態に従い、横断するようなループフィルタリング演算が無効にされる仮想境界と呼ばれる概念を使用することができる。符号器は、不連続境界を仮想境界として設定することができ、それにより不連続境界上でループフィルタを適用しないことができる。仮想境界の位置は、ビットストリーム内でシグナリングされ、符号器は、現在の投影フォーマットに従って仮想境界の位置を変更することができる。

20

30

【0066】

[083] 360度映像の他に、仮想境界は、超低遅延アプリケーション内で主に使用される漸進的復号化リフレッシュ（GDR）にも使用することができる。超低遅延アプリケーションでは、ランダムアクセスポイントピクチャとしてイントラ符号化ピクチャを挿入することは、イントラ符号化ピクチャのサイズが大きくなることにより、許容できない伝送レイテンシを引き起こす場合がある。レイテンシを減らすために、B/Pピクチャ内にイントラ符号化領域を挿入することによってピクチャが漸進的にリフレッシュされるGDRが採用される。誤り伝播を防ぐために、ピクチャ内のリフレッシュ領域内のピクセルは、現在のピクチャ又は参照ピクチャの未リフレッシュ領域内のものを参照することができない。従って、リフレッシュ領域及び未リフレッシュ領域の境界にわたってループフィルタリングを適用することができない。上記の仮想境界の方式により、符号器は、リフレッシュ領域及び未リフレッシュ領域の境界を仮想境界として設定することができ、その場合、その境界を横断してループフィルタリング演算を適用することができない。

40

【0067】

[084] いくつかの実施形態によれば、仮想境界は、シーケンスパラメータセット（SPS）、又はピクチャヘッダ（PH）内でシグナリングすることができる。ピクチャヘッダは、特定のピクチャに関する情報を運び、同じピクチャに属する全てのスライスに共通の情報を含む。PHは、仮想境界に関係する情報を含み得る。ピクチャごとに1つのPHが設定される。シーケンスパラメータセットは、符号化レイヤ映像シーケンス（CLVS）

50

に関する構文要素を含む。S P S は、全符号化レイヤ映像シーケンス (C L V S) 内の全てのピクチャによって共有されるシーケンスレベル情報を含み、ビットストリームが何を含むか、及びビットストリーム内の情報をどのように使用できるかについての全体像を与えることができる。S P S では、仮想境界存在フラグ「sps_virtual_boundaries_present_flag」が最初にシグナリングされる。フラグが真である場合、仮想境界の数及び各仮想境界の位置が、S P S を参照するピクチャについてシグナリングされる。「sps_virtual_boundaries_present_flag」が偽である場合、別の仮想境界存在フラグ「ph_virtual_boundaries_present_flag」が P H 内でシグナリングされ得る。同様に、「ph_virtual_boundaries_present_flag」が真である場合、仮想境界の数及び各仮想境界の位置が、P H に関連するピクチャについてシグナリングされ得る。

10

【 0 0 6 8 】

[085] 図 5 の表 1 に仮想境界の S P S 構文を示す。図 5 の S P S 構文のセマンティクスを以下のように示す。

【 0 0 6 9 】

[086] 1 に等しい「sps_virtual_boundaries_present_flag」は、仮想境界の情報が S 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「sps_virtual_boundaries_present_flag」は、仮想境界の情報が S P S 内でシグナリングされないことを指定する。1 つ以上の仮想境界が S P S 内でシグナリングされる場合、S P S を参照するピクチャ内の仮想境界にわたるインループフィルタリング演算が無効にされる。インループフィルタリング演算は、デブロッキングフィルタ、サンプル適応オフセットフィルタ、及び適応ループフィルタ演算を含む。

20

【 0 0 7 0 】

[087] 「sps_num_ver_virtual_boundaries」は、S P S 内にある「sps_virtual_boundaries_pos_x[i]」構文要素の数を指定する。「sps_num_ver_virtual_boundaries」がない場合、その値は、0 に等しいと推論される。

【 0 0 7 1 】

[088] 「sps_virtual_boundaries_pos_x[i]」は、8 で割ったルマサンプル単位の i 番目の垂直仮想境界の位置を指定する。「sps_virtual_boundaries_pos_x[i]」の値は、1 ~ $\text{Ceil}(\text{pic_width_in_luma_samples} \div 8) - 1$ の範囲内にある。

【 0 0 7 2 】

30

[089] 「sps_num_hor_virtual_boundaries」は、S P S 内にある「sps_virtual_boundaries_pos_y[i]」構文要素の数を指定する。「sps_num_hor_virtual_boundaries」がない場合、その値は、0 に等しいと推論される。

【 0 0 7 3 】

[090] 「sps_virtual_boundaries_pos_y[i]」は、8 で割ったルマサンプル単位の i 番目の水平仮想境界の位置を指定する。「sps_virtual_boundaries_pos_y[i]」の値は、1 ~ $\text{Ceil}(\text{pic_height_in_luma_samples} \div 8) - 1$ の範囲内にある。

【 0 0 7 4 】

[091] 図 6 の表 2 に仮想境界の P H 構文を示す。図 6 の P H 構文のセマンティクスを以下に示す。

40

【 0 0 7 5 】

[092] 「ph_num_ver_virtual_boundaries」は、P H 内にある「ph_virtual_boundaries_pos_x[i]」構文要素の数を指定する。「ph_num_ver_virtual_boundaries」がない場合、その値は、0 に等しいと推論される。

【 0 0 7 6 】

[093] パラメータVirtualBoundariesNumVerは、以下のように導出される。

VirtualBoundariesNumVer=sps_virtual_boundaries_present_flag?

sps_num_ver_virtual_boundaries:ph_num_ver_virtual_boundaries

【 0 0 7 7 】

[094] 「ph_virtual_boundaries_pos_x[i]」は、8 で割ったルマサンプル単位の i

50

番目の垂直仮想境界の位置を指定する。「ph_virtual_boundaries_pos_x[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_width_in_luma_samples} \div 8) - 1$ の範囲内にある。

【0078】

[095] ルマサンプル単位の垂直仮想境界「VirtualBoundariesPosX[i]」の位置は、以下のように導出される。

$$\text{VirtualBoundariesPosX}[i] = (\text{sps_virtual_boundaries_present_flag?} \\ \text{sps_virtual_boundaries_pos_x}[i] : \text{ph_virtual_boundaries_pos_x}[i]) * 8$$

【0079】

[096] 任意の2つの垂直仮想境界間の距離は、CtbSizeYルマサンプル以上であり得る。

【0080】

[097] 「ph_num_hor_virtual_boundaries」は、PH内にある「ph_virtual_boundaries_pos_y[i]」構文要素の数を指定する。「ph_num_hor_virtual_boundaries」がない場合、その値は、0に等しいと推論される。

【0081】

[098] パラメータVirtualBoundariesNumHorは、以下のように導出される。

$$\text{VirtualBoundariesNumHor} = \text{sps_virtual_boundaries_present_flag?} \\ \text{sps_num_hor_virtual_boundaries} : \text{ph_num_hor_virtual_boundaries}$$

【0082】

[099] 「ph_virtual_boundaries_pos_y[i]」は、8で割ったルマサンプル単位のi番目の水平仮想境界の位置を指定する。「ph_virtual_boundaries_pos_y[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_height_in_luma_samples} \div 8) - 1$ の範囲内にある。

【0083】

[100] ルマサンプル単位の水平仮想境界「VirtualBoundariesPosY[i]」の位置は、以下のように導出される。

$$\text{VirtualBoundariesPosY}[i] = (\text{sps_virtual_boundaries_present_flag?} \\ \text{sps_virtual_boundaries_pos_y}[i] : \text{ph_virtual_boundaries_pos_y}[i]) * 8$$

【0084】

[101] 任意の2つの水平仮想境界間の距離は、CtbSizeYルマサンプル以上であり得る。

【0085】

[102] ラップアラウンド動き補償は、別の360度映像符号化ツールである。従来の動き補償では、動きベクトルが参照ピクチャのピクチャ境界を超えてサンプルを参照する場合、対応するピクチャ境界上の最近傍からコピーすることによって境界外のサンプルの値を導出するために反復パディングが適用される。360度映像では、この反復パディングの方法は、適しておらず、再構成されたビューポート映像において「シームアーティファクト」と呼ばれる視覚的アーティファクトを引き起こし得る。360度映像は、球上で捕捉され、元来「境界」を有さないため、投影領域内の参照ピクチャの境界の外側の参照サンプルは、球領域内の隣接サンプルから常に得ることができる。2Dから3D及び3Dから2Dへの座標変換並びに端数サンプル位置のためのサンプル補間を含むため、一般的な投影フォーマットでは、球領域内の対応する隣接サンプルを導出することが困難であり得る。左側のピクチャ境界の外側の球近傍を右側のピクチャ境界の内側のサンプルから得ることができ、その逆も同様であるため、この問題は、正距円筒図法による投影(ERP)又はパデッドERP(PERP)フォーマットの左右の境界に関して非常に単純である。ERP又はPERP投影フォーマットの広範な使用、及び実装が相対的に容易であることを所与として、ERP投影フォーマットで符号化された360度映像の視覚的品質を改善するために水平ラップアラウンド動き補償を使用することができる。

【0086】

[103] 図7Aに水平ラップアラウンド動き補償プロセスを示す。参照ブロックの一部が投影領域内の参照ピクチャの左側(又は右側)の境界の外側にある場合、反復パディングの代わりに、投影領域内の右側(又は左側)の境界の方に参照ピクチャ内で位置する対応する球近傍から「境界外」の部分が取られる。反復パディングは、上部及び下部のピクチャ

10

20

30

40

50

ャ境界にのみ使用される。図 7 B に示すように、水平ラップアラウンド動き補償は、360 度映像符号化で多くの場合に使用される非規範的パディング方法 (non-normative padding method) と組み合わせることができる。これは、パディングの前の E R P ピクチャの幅に設定されるべきラップアラウンドオフセットを示すための高レベル構文要素をシグナリングすることによって実現され得る。この構文は、水平ラップアラウンドの位置を適宜調節するために使用される。この構文は、左側及び右側のピクチャ境界上の特定のパディング量の影響を受けず、従って (左右のパディングが異なる) E R P ピクチャの非対称パディングを当然にサポートする。水平ラップアラウンド動き補償は、参照サンプルが参照ピクチャの左側及び右側の境界の外側にある場合に動き補償にとってより有意味の情報を提供する。このツールは、レート - 歪み性能に関してだけでなく、再構成される 360 度映像のシームアーティファクトを低減すること及び主観的品質を改善することに関しても圧縮性能を改善する。水平ラップアラウンド動き補償は、調節された等積投影など、水平方向に一定のサンプリング密度を有する他の単一面投影フォーマットにも使用することができる。

10

【0087】

[104] いくつかの実施形態によれば、ラップアラウンド動き補償は、S P S 内でシグナリングされる。まず、有効化フラグがシグナリングされる。有効化フラグが真である場合、ラップアラウンドオフセットがシグナリングされる。S P S 構文を図 8 に示し、対応するセマンティクスを以下で与える。

【0088】

20

[105] 1 に等しい「sps_ref_wraparound_enabled_flag」は、水平ラップアラウンド動き補償がインター予測内で適用されることを指定する。0 に等しい「sps_ref_wraparound_enabled_flag」は、水平ラップアラウンド動き補償が適用されないことを指定する。(CtbSizeY / MinCbSizeY + 1) の値が (pic_width_in_luma_samples / MinCbSizeY - 1) よりも大きく、「pic_width_in_luma_samples」が、S P S を参照する任意のピクチャパラメータセット (P P S) 内の「pic_width_in_luma_samples」の値である場合、「sps_ref_wraparound_enabled_flag」の値は、0 に等しい。

【0089】

[106] 「sps_ref_wraparound_offset_minus1」プラス 1 は、MinCbSizeY ルマサンプル単位の水平ラップアラウンド位置を計算するために使用されるオフセットを指定する。「ref_wraparound_offset_minus1」の値は (CtbSizeY / MinCbSizeY) + 1 ~ (pic_width_in_luma_samples / MinCbSizeY) - 1 の範囲内にあり、「pic_width_in_luma_samples」は、S P S を参照する任意の P P S 内の「pic_width_in_luma_samples」の値である。

30

【0090】

[107] CtbSizeY は、符号化ツリーブロック (C T B) のルマサイズであり、MinCbSizeY は、ルマ符号化ブロックの最小サイズであり、「pic_width_in_luma_samples」は、ルマサンプル単位のピクチャ幅である。

【0091】

[108] いくつかの実施形態によれば、ピクチャシーケンス内の全てのピクチャの幅及び高さの最大値が S P S 内でシグナリングされ、次いで現在のピクチャに関してピクチャの幅及び高さが各 P P S 内でシグナリングされる。ピクチャの幅及び高さの最大値をシグナリングするための構文を図 9 の表 4 に示し、ピクチャの幅及び高さをシグナリングするための構文を図 10 の表 5 に示す。図 9 及び図 10 に対応するセマンティクスを以下で与える。

40

【0092】

[109] 1 に等しい「ref_pic_resampling_enabled_flag」は、S P S を参照する符号化レイヤ映像シーケンス (C L V S) 内の符号化ピクチャを復号化するとき、参照ピクチャ再サンプリングを適用できることを指定する。0 に等しい ref_pic_resampling_enabled_flag は、S P S を参照する C L V S 内のピクチャを復号化するとき、参照ピクチャ

50

再サンプリングが適用されないことを指定する。例えば、復号化プログラムは、フレームのそれぞれを復号化することができる。現在のフレームの解像度が参照ピクチャの解像度と異なると復号化プログラムが判定する場合、復号化プログラムは、参照ピクチャに対して適切な再サンプリングを行い、生成した再サンプリング参照ピクチャを現在のフレームのための参照ピクチャとして使用することができる。すなわち、ピクチャの空間解像度が映像シーケンス内で変更されることが許可される場合、参照ピクチャの再サンプリングが必要である。参照ピクチャの適切な再サンプリングは、参照ピクチャのダウンサンプリング又はアップサンプリングであり得る。

【 0 0 9 3 】

[110] 「pic_width_max_in_luma_samples」は、S P Sを参照する各復号化ピクチャのルマサンプル単位の最大幅を指定する。「pic_width_max_in_luma_samples」は、0 に等しくなくてもよく、 $\text{Max}(8, \text{MinCbSizeY})$ の整数倍であり得る。

10

【 0 0 9 4 】

[111] 「pic_height_max_in_luma_samples」は、S P Sを参照する各復号化ピクチャのルマサンプル単位の最大高さを指定する。「pic_height_max_in_luma_samples」は、0 に等しくなくてもよく、 $\text{Max}(8, \text{MinCbSizeY})$ の整数倍であり得る。

【 0 0 9 5 】

[112] 「pic_width_in_luma_samples」は、P P Sを参照する各復号化ピクチャのルマサンプル単位の幅を指定する。「pic_width_in_luma_samples」は、0 に等しくなくてもよい。むしろ、「pic_width_in_luma_samples」は、 $\text{Max}(8, \text{MinCbSizeY})$ の整数倍であり得、「pic_width_max_in_luma_samples」以下であり得る。

20

【 0 0 9 6 】

[113] 「subpics_present_flag」が1 に等しいか又は「ref_pic_resampling_enabled_flag」が0 に等しい場合、「pic_width_in_luma_samples」の値は、「pic_width_max_in_luma_samples」に等しい。

【 0 0 9 7 】

[114] 「pic_height_in_luma_samples」は、P P Sを参照する各復号化ピクチャのルマサンプル単位の高さを指定する。「pic_height_in_luma_samples」は、0 に等しくなくてもよい。むしろ、「pic_height_in_luma_samples」は、 $\text{Max}(8, \text{MinCbSizeY})$ の整数倍であり得、「pic_height_max_in_luma_samples」以下であり得る。

30

【 0 0 9 8 】

[115] 「subpics_present_flag」が1 に等しいか又は「ref_pic_resampling_enabled_flag」が0 に等しい場合、「pic_height_in_luma_samples」の値は、「pic_height_max_in_luma_samples」に等しい。

【 0 0 9 9 】

[116] 上述の仮想境界のS P Sシグナリングは、いくらかの曖昧さを引き起こし得る。具体的には、S P S内でシグナリングされる「sps_virtual_boundaries_pos_x[i]」及び「sps_virtual_boundaries_pos_y[i]」の範囲は、それぞれ $0 \sim \text{Ceil}(\text{pic_width_in_luma_samples} \div 8) - 1$ 及び $0 \sim \text{Ceil}(\text{pic_height_in_luma_samples} \div 8) - 1$ である。しかし、上記で説明したように、「pic_width_in_luma_samples」及び「pic_height_in_luma_samples」は、P P S内でシグナリングされ、それらは、P P Sごとに異なり得る。同じS P Sを参照する複数のP P Sがあり得るため、「pic_width_in_luma_samples」及び「pic_height_in_luma_samples」が「sps_virtual_boundaries_pos_x[i]」及び「sps_virtual_boundaries_pos_y[i]」の上限として設定されるべきであるかどうかは、明らかではない。

40

【 0 1 0 0 】

[117] さらに、上述のラップアラウンド動き補償のS P Sシグナリングは、いくつかの問題を有し得る。具体的には、「sps_ref_wraparound_enabled_flag」及び「sps_ref_wraparound_offset_minus1」は、S P S内でシグナリングされる構文要素であるが、P P S上でシグナリングされる「pic_width_in_luma_samples」に依存する「sps_ref

50

「sps_ref_wraparound_enabled_flag」の適合制約がある。「sps_ref_wraparound_offset_minus1」の範囲も、PPS上でシグナリングされる「pic_width_in_luma_samples」に依存する。これらの依存関係は、いくらかの問題を引き起こす。まず、関連するPPSの全てにおいて構文要素によってSPS構文要素の値を制限することは、効率的な方法ではなく、なぜなら、SPSは、PPSよりも高いレベルであるためである。さらに、高レベル構文が低レベル構文を参照すべきでないことが通常理解される。第2に、現在の設計によれば、「sps_ref_wraparound_enabled_flag」は、SPSを参照するシーケンス内の全てのピクチャの幅が制約条件を満たす場合にのみ真であり得る。従って、たとえ全シーケンス内の1フレームのみでも制約条件を満たさない場合、ラップアラウンド動き補償を使用することができない。従って、全シーケンスに対するラップアラウンド動き補償の利益が1つのフレームのみを理由に失われる。

10

【0101】

[118] 本開示は、仮想境界又はラップアラウンド動き補償をシグナリングすることに関連する上記の問題を解決するための方法を提供する。開示する方法に従ういくつかの例示の実施形態を以下で詳細に説明する。

【0102】

[119] いくつかの例示の実施形態では、仮想境界のシグナリングに関連する上記の問題を解決するために、「sps_virtual_boundaries_pos_x[i]」及び「sps_virtual_boundaries_pos_y[i]」の値の上限をシーケンス内のピクチャの幅及び高さの最小値に変更する。従って、各ピクチャについて、SPS内でシグナリングされる仮想境界の位置（「sps_virtual_boundaries_pos_x[i]」及び「sps_virtual_boundaries_pos_y[i]」）がピクチャ境界を越えない。

20

【0103】

[120] これらの実施形態に従うセマンティクスを以下に記載する。

【0104】

[121] 「sps_virtual_boundaries_pos_x[i]」は、8で割ったルマサンプル単位のi番目の垂直仮想境界の位置を指定する。「sps_virtual_boundaries_pos_x[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_width_in_luma_samples} \div 8) - 1$ の範囲内にあり、「pic_width_in_luma_samples」は、SPSを参照する任意のPPS内の「pic_width_in_luma_samples」の値である。

30

【0105】

[122] 「sps_virtual_boundaries_pos_y[i]」は、8で割ったルマサンプル単位のi番目の水平仮想境界の位置を指定する。「sps_virtual_boundaries_pos_y[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_height_in_luma_samples} \div 8) - 1$ の範囲内にあり、「pic_height_in_luma_samples」は、SPSを参照する任意のPPS内の「pic_width_in_luma_samples」の値である。

【0106】

[123] いくつかの例示の実施形態では、仮想境界のシグナリングに関連する上記の問題を解決するために、「sps_virtual_boundaries_pos_x[i]」及び「sps_virtual_boundaries_pos_y[i]」の値の上限を、シーケンス内のピクチャの幅及び高さそれぞれの最大値である「pic_width_max_in_luma_samples」及び「pic_height_max_in_luma_samples」にそれぞれ変更する。各ピクチャについて、SPS内でシグナリングされる仮想境界の位置（「sps_virtual_boundaries_pos_x[i]」及び「sps_virtual_boundaries_pos_y[i]」）がピクチャ境界を越える場合、仮想境界を境界内で切り取るか又は破棄する。

40

【0107】

[124] これらの実施形態に従うセマンティクスを以下に記載する。

【0108】

[125] 「sps_virtual_boundaries_pos_x[i]」は、8で割ったルマサンプル単位のi番目の垂直仮想境界の位置を指定する。「sps_virtual_boundaries_pos_x[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_width_max_in_luma_samples} \div 8) - 1$ の範囲内にある。

50

【 0 1 0 9 】

[126] 「sps_virtual_boundaries_pos_y[i]」は、8で割ったルマサンプル単位のi番目の水平仮想境界の位置を指定する。「sps_virtual_boundaries_pos_y[i]」の値は、1 ~ $\text{Ceil}(\text{pic_height_max_in_luma_samples} \div 8) - 1$ の範囲内にある。

【 0 1 1 0 】

[127] 一例として、各ピクチャについて、SPS内でシグナリングされる仮想境界の位置が現在のピクチャ境界内で切り取られる。この例では、導出される仮想境界の位置VirtualBoundariesPosX[i]及びVirtualBoundariesPosY[i]並びに仮想境界の数VirtualBoundariesNumVer及びVirtualBoundariesNumHorが以下のように導出される。

```
VirtualBoundariesNumVer=sps_virtual_boundaries_present_flag?
    sps_num_ver_virtual_boundaries:ph_num_ver_virtual_boundaries
VirtualBoundariesPosX[i]=(sps_virtual_boundaries_present_flag?
    min(Ceil(pic_width_in_luma_samples ÷ 8)-1,sps_virtual_boundaries_pos_x[
i]):ph_virtual_boundaries_pos_x[i])*8
VirtualBoundariesNumHor=sps_virtual_boundaries_present_flag?
    sps_num_hor_virtual_boundaries:ph_num_hor_virtual_boundaries
VirtualBoundariesPosY[i]=(sps_virtual_boundaries_present_flag?
    min(Ceil(pic_width_in_luma_samples ÷ 8)-1,sps_virtual_boundaries_pos_y[
i]):ph_virtual_boundaries_pos_y[i])*8
```

【 0 1 1 1 】

[128] 任意の2つの垂直仮想境界間の距離は、0又はCtbSizeYルマサンプル以上であり得る。

【 0 1 1 2 】

[129] 任意の2つの水平仮想境界間の距離は、0又はCtbSizeYルマサンプル以上であり得る。

【 0 1 1 3 】

[130] 別の例として、各ピクチャについて、SPS内でシグナリングされる仮想境界の位置が現在のピクチャ境界を越える場合、現在のピクチャ内で仮想境界は、使用されない。導出される仮想境界の位置VirtualBoundariesPosX[i]及びVirtualBoundariesPosY[i]並びに仮想境界の数VirtualBoundariesNumVer及びVirtualBoundariesNumHorは、以下のように導出される。

```
VirtualBoundariesNumVer=sps_virtual_boundaries_present_flag?
    sps_num_ver_virtual_boundaries:ph_num_ver_virtual_boundaries
VirtualBoundariesPosXInPic[i]=(sps_virtual_boundaries_present_flag?
    sps_virtual_boundaries_pos_x[i]:ph_virtual_boundaries_pos_x[i])*8,(i=0..
VirtualBoundariesNumVer)
VirtualBoundariesNumHor=sps_virtual_boundaries_present_flag?
    sps_num_hor_virtual_boundaries:ph_num_hor_virtual_boundaries
VirtualBoundariesPosYInPic[i]=(sps_virtual_boundaries_present_flag?
    sps_virtual_boundaries_pos_y[i]:ph_virtual_boundaries_pos_y[i])*8,(i=0..
VirtualBoundariesNumHor)
for(i=0,j=0;i VirtualBoundariesNumVer;i++){
    if(VirtualBoundariesPosXInPic[i] =Ceil(pic_width_in_luma_samples ÷ 8)-1
){
    VirtualBoundariesPosX[j++]=VirtualBoundariesPosXInPic[i]
    }
}
VirtualBoundariesNumVer=j
for(i=0,j=0;i VirtualBoundariesNumHor;i++){
    if(VirtualBoundariesPosYInPic[i] =Ceil(pic_height_in_luma_samples ÷ 8)-
```

```

1){
    VirtualBoundariesPosY[j++]=VirtualBoundariesPosYInPic[i]
}
}
VirtualBoundariesNumHor=j
【 0 1 1 4 】
[131] 代わりに、導出される仮想境界の位置「VirtualBoundariesPosX[i]」、「VirtualBoundariesPosY[i]」並びに仮想境界の数VirtualBoundariesNumVer及びVirtualBoundariesNumHorは、以下のように導出される。
    for(i=0,j=0;i<sps_num_ver_virtual_boundaries;i++){
        if(sps_virtual_boundaries_pos_x[i]>=Ceil(pic_width_in_luma_samples÷8)-1){
            VirtualBoundariesPosX[j++]=sps_virtual_boundaries_pos_x[i]
        }
    }
    VirtualBoundariesNumVer=j
    for(i=0,j=0;i<sps_num_hor_virtual_boundaries;i++){
        if(sps_virtual_boundaries_pos_y[i]>=Ceil(pic_height_in_luma_samples÷8)-1){
            VirtualBoundariesPosY[j++]=sps_virtual_boundaries_pos_y[i]
        }
    }
    VirtualBoundariesNumHor=j
    VirtualBoundariesNumVer=sps_virtual_boundaries_present_flag?
        VirtualBoundariesNumVer:ph_num_ver_virtual_boundaries
    VirtualBoundariesPosX[i]=(sps_virtual_boundaries_present_flag?
        VirtualBoundariesPosX[i]:ph_virtual_boundaries_pos_x[i])*8,(i=0..VirtualBoundariesNumVer)
    VirtualBoundariesNumHor=sps_virtual_boundaries_present_flag?
        VirtualBoundariesNumHor:ph_num_hor_virtual_boundaries
    VirtualBoundariesPosY[i]=(sps_virtual_boundaries_present_flag?
        VirtualBoundariesPosY[i]:ph_virtual_boundaries_pos_y[i])*8,(i=0..VirtualBoundariesNumHor)
【 0 1 1 5 】
[132] いくつかの例示的实施形態では、仮想境界のシグナリングに関連する上記の問題を解決するために、「sps_virtual_boundaries_pos_x[i]」及び「sps_virtual_boundaries_pos_y[i]」の値の上限を、シーケンス内のピクチャの幅及び高さそれぞれの最大値である「pic_width_max_in_luma_samples」及び「pic_height_max_in_luma_samples」にそれぞれ変更する。各ピクチャについて、SPS内でシグナリングされる仮想境界の位置（「sps_virtual_boundaries_pos_x[i]」及び「sps_virtual_boundaries_pos_y[i]」）が、SPS内でシグナリングされる最大ピクチャ幅及び高さ、PPS内でシグナリングされる現在のピクチャの幅及び高さとの間の比率に従ってスケールされる。
【 0 1 1 6 】
[133] これらの実施形態に従うセマンティクスを以下に記載する。
【 0 1 1 7 】
[134] 「sps_virtual_boundaries_pos_x[i]」は、8で割ったルマサンプル単位のi番目の垂直仮想境界の位置を指定する。「sps_virtual_boundaries_pos_x[i]」の値は、1～Ceil(pic_width_max_in_luma_samples÷8)-1の範囲内にある。
【 0 1 1 8 】
[135] 「sps_virtual_boundaries_pos_y[i]」は、8で割ったルマサンプル単位のi

```

番目の水平仮想境界の位置を指定する。「sps_virtual_boundaries_pos_y[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_height_max_in_luma_samples} \div 8) - 1$ の範囲内にある。

【0 1 1 9】

[136] ピクチャごとの仮想境界の位置を導出するために、スケーリング比を最初に計算し、次いでSPS内でシグナリングされる仮想境界の位置を以下のようにスケーリングする。

$$\text{VBScaleX} = ((\text{pic_width_max_in_luma_samples} - 14) + (\text{pic_width_in_luma_samples} - 1)) / \text{pic_width_in_luma_samples}$$

$$\text{VBScaleY} = ((\text{pic_height_max_in_luma_samples} - 14) + (\text{pic_height_in_luma_samples} - 1)) / \text{pic_height_in_luma_samples}$$

$$\text{SPSVirtualBoundariesPosX}[i] = (\text{sps_virtual_boundaries_pos_x}[i] \times \text{VBScaleX} + (1 - 13)) \div 14$$

$$\text{SPSVirtualBoundariesPosY}[i] = (\text{sps_virtual_boundaries_pos_y}[i] \times \text{VBScaleY} + (1 - 13)) \div 14$$

【0 1 2 0】

[137] 例えば「SPSVirtualBoundariesPosX[i]」及び「SPSVirtualBoundariesPosY[i]」は、以下のように8ピクセルグリッドにさらに丸めることができる。

$$\text{SPSVirtualBoundariesPosX}[i] = ((\text{SPSVirtualBoundariesPosX}[i] + 4) \div 3) \times 3$$

$$\text{SPSVirtualBoundariesPosY}[i] = ((\text{SPSVirtualBoundariesPosY}[i] + 4) \div 3) \times 3$$

【0 1 2 1】

[138] 最後に、導出される仮想境界の位置「VirtualBoundariesPosX[i]」及び「VirtualBoundariesPosY[i]」並びに仮想境界の数「VirtualBoundariesNumVer」及び「VirtualBoundariesNumHor」は、以下のように導出される。

$$\text{VirtualBoundariesNumVer} = \text{sps_virtual_boundaries_present_flag}?$$

$$\text{sps_num_ver_virtual_boundaries} : \text{ph_num_ver_virtual_boundaries}$$

$$\text{VirtualBoundariesPosX}[i] = (\text{sps_virtual_boundaries_present_flag}?$$

$$\text{SPSVirtualBoundariesPosX}[i] : \text{ph_virtual_boundaries_pos_x}[i]) \times 8, (i = 0 \dots \text{VirtualBoundariesNumVer})$$

$$\text{VirtualBoundariesNumHor} = \text{sps_virtual_boundaries_present_flag}?$$

$$\text{sps_num_hor_virtual_boundaries} : \text{ph_num_hor_virtual_boundaries}$$

$$\text{VirtualBoundariesPosY}[i] = (\text{sps_virtual_boundaries_present_flag}?$$

$$\text{SPSVirtualBoundariesPosY}[i] : \text{ph_virtual_boundaries_pos_y}[i]) \times 8, (i = 0 \dots \text{VirtualBoundariesNumHor})$$

【0 1 2 2】

[139] いくつかの例示的实施形態では、仮想境界のシグナリングに関連する上記の問題を解決するために、シーケンスレベルの仮想境界のシグナリングと、ピクチャの幅及び高さの変更とを相互排他的に使用する。例えば、仮想境界がSPS内でシグナリングされる場合、ピクチャの幅及び高さをシーケンス内で変更することができない。ピクチャの幅及び高さがシーケンス内で変更される場合、仮想境界をSPS内でシグナリングすることができない。

【0 1 2 3】

[140] 本発明の実施形態では、「sps_virtual_boundaries_pos_x[i]」及び「sps_virtual_boundaries_pos_y[i]」の値の上限を、シーケンス内のピクチャの幅及び高さそれぞれの最大値である「pic_width_max_in_luma_samples」及び「pic_height_max_in_luma_samples」にそれぞれ変更する。

【0 1 2 4】

[141] 本発明の実施形態に従う「sps_virtual_boundaries_pos_x[i]」及び「sps_virtual_boundaries_pos_y[i]」のセマンティクスは、下記のとおりである。

【0 1 2 5】

[142] 「sps_virtual_boundaries_pos_x[i]」は、8で割ったルマサンプル単位の i

10

20

30

40

50

番目の垂直仮想境界の位置を指定する。「sps_virtual_boundaries_pos_x[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_width_max_in_luma_samples} \div 8) - 1$ の範囲内にある。

【0126】

[143] 「sps_virtual_boundaries_pos_y[i]」は、8で割ったルマサンプル単位のi番目の水平仮想境界の位置を指定する。「sps_virtual_boundaries_pos_y[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_height_max_in_luma_samples} \div 8) - 1$ の範囲内にある。

【0127】

[144] 一例として、「pic_width_in_luma_samples」及び「pic_height_in_luma_samples」のビットストリーム適合要件は、以下のように課することができる。

【0128】

[145] (1) 「subpics_present_flag」が1に等しいか、又は(2) 「ref_pic_resampling_enabled_flag」が0に等しいか、又は(3) 「sps_virtual_boundaries_present_flag」が1に等しい場合、「pic_width_in_luma_samples」の値は、「pic_width_max_in_luma_samples」に等しい。この制約の第3の条件(すなわち「sps_virtual_boundaries_present_flag」が1に等しいこと)に従い、仮想境界がSPS内にある場合、シーケンス内の各ピクチャは、シーケンス内のピクチャの最大幅に等しい同じ幅を有する。

【0129】

[146] (1) 「subpics_present_flag」が1に等しいか、又は(2) 「ref_pic_resampling_enabled_flag」が0に等しいか、又は(3) 「sps_virtual_boundaries_present_flag」が1に等しい場合、「pic_height_in_luma_samples」の値は、「pic_height_max_in_luma_samples」に等しい。この制約の第3の条件(すなわち「sps_virtual_boundaries_present_flag」が1に等しいこと)に従い、仮想境界がSPS内にある場合、シーケンス内の各ピクチャは、シーケンス内のピクチャの最大高さに等しい同じ高さを有する。

【0130】

[147] 別の例として、「pic_width_in_luma_samples」及び「pic_height_in_luma_samples」のビットストリーム適合要件は、以下のように課することができる。

【0131】

[148] (1) 「subpics_present_flag」が1に等しいか、又は(2) 「ref_pic_resampling_enabled_flag」が0に等しいか、又は(3) 「sps_num_ver_virtual_boundaries」が0に等しくない場合、「pic_width_in_luma_samples」の値は、「pic_width_max_in_luma_samples」に等しい。この制約の第3の条件(すなわち「sps_num_ver_virtual_boundaries」が0に等しくないこと)に従い、垂直仮想境界の数が0を上回る(すなわち少なくとも1つの垂直仮想境界がある)場合、シーケンス内の各ピクチャは、シーケンス内のピクチャの最大幅に等しい同じ幅を有する。

【0132】

[149] (1) 「subpics_present_flag」が1に等しいか、又は(2) 「ref_pic_resampling_enabled_flag」が0に等しいか、又は(3) 「sps_num_hor_virtual_boundaries」が0に等しくない場合、「pic_height_in_luma_samples」の値は、「pic_height_max_in_luma_samples」に等しい。この制約の第3の条件(すなわち「sps_num_hor_virtual_boundaries」が0に等しくないこと)に従い、垂直仮想境界の数が0を上回る(すなわち少なくとも1つの垂直仮想境界がある)場合、シーケンス内の各ピクチャは、シーケンス内のピクチャの最大幅に等しい同じ幅を有する。

【0133】

[150] 別の例として、「sps_virtual_boundaries_present_flag」のビットストリーム適合要件は、以下のように課することができる: 「ref_pic_resampling_enabled_flag」が1である場合に「sps_virtual_boundaries_present_flag」が0であることは、ビットストリーム適合の要件である。この制約に従い、参照ピクチャ再サンプリングが有効にされる場合、仮想境界は、SPS内にあるべきではない。この制約の原理は、以下の

10

20

30

40

50

とおりである。参照ピクチャ再サンプリングは、現在のピクチャが、参照ピクチャと異なる解像度を有する場合に使用される。ピクチャの解像度がシーケンス内で変更されることが許可される場合、異なるピクチャは、異なる解像度を有し得るため、仮想境界の位置もピクチャごとに変更することができる。従って、仮想境界の位置をピクチャレベル（例えば、PPS）でシグナリングすることができる。シーケンスレベルの（すなわちSPS内でシグナリングされる）仮想境界のシグナリングは、適切ではない。

【0134】

[151] 別の例として、「sps_virtual_boundaries_present_flag」は、「ref_pic_resampling_enabled_flag」に基づいて条件付きでシグナリングされる。例えば、0に等しい「ref_pic_resampling_enabled_flag」は、SPSを参照するCLVS内のピクチャを復号化するとき、参照ピクチャ再サンプリングが適用されないことを指定し、「sps_virtual_boundaries_present_flag」は、「!ref_pic_resampling_enabled_flag」の値が1であることに基づいてシグナリングされる。この構文を図11に示し、表1（図5）内の構文に対する変更をイタリック体で示す。関連するセマンティクスを以下のとおり記載する。

【0135】

[152] 1に等しい「sps_virtual_boundaries_present_flag」は、仮想境界の情報がSPS内でシグナリングされることを指定する。0に等しい「sps_virtual_boundaries_present_flag」は、仮想境界の情報がSPS内でシグナリングされないことを指定する。1つ以上の仮想境界がSPS内でシグナリングされる場合、SPSを参照するピクチャ内の仮想境界を横断するインループフィルタリング演算が無効にされる。「sps_virtual_boundaries_present_flag」がない場合、その値は、0であると推論される。インループフィルタリング演算は、デブロッキングフィルタ、サンプル適応オフセットフィルタ、及び適応ループフィルタ演算を含む。

【0136】

[153] 別の例として、「ref_pic_resampling_enabled_flag」は、「sps_virtual_boundaries_present_flag」に基づいて条件付きでシグナリングされる。この構文を図12に示し、表1（図5）内の構文に対する変更を、イタリック体及び取り消し線を用いて示す。関連するセマンティクスを以下に記載する。

【0137】

[154] 1に等しい「ref_pic_resampling_enabled_flag」は、SPSを参照するCLVS内の符号化ピクチャを復号化するとき、参照ピクチャ再サンプリングを適用できることを指定する。0に等しい「ref_pic_resampling_enabled_flag」は、SPSを参照するCLVS内のピクチャを復号化するとき、参照ピクチャ再サンプリングが適用されないことを指定する。「ref_pic_resampling_enabled_flag」がない場合、その値は、0であると推論される。

【0138】

[155] さらに、ラップアラウンド動き補償のシグナリングに関連する上記の問題を解決するために、「sps_ref_wraparound_enabled_flag」は、SPSを参照するシーケンス内の全てのピクチャの幅が制約条件を満たす場合にのみ真であり得、本開示によって以下の実施形態が与えられる。

【0139】

[156] 一部の実施形態では、ラップアラウンド動き補償有効化フラグ及びラップアラウンドオフセットに対する制約は、SPS内でシグナリングされる最大ピクチャ幅に依存するように変更される。ピクチャレベルにおいて、ピクチャ幅のチェックが導入される。ラップアラウンドは、条件を満たす幅を有するピクチャにのみ適用することができる。その幅が条件を満たさないピクチャでは、たとえ「sps_ref_wraparound_enabled_flag」が真でもラップアラウンドがオフにされる。そうすることにより、PPS内でシグナリングされるピクチャ幅を使用してSPS内でシグナリングされる「sps_ref_wraparound_enabled_flag」及び「sps_ref_wraparound_offset_minus1」を制限する必要がない。

構文は、変更されず、「sps_ref_wraparound_enabled_flag」及び「sps_ref_wraparound_offset_minus1」のセマンティクスは、下記のとおりである。

【 0 1 4 0 】

[157] 1 に等しい「sps_ref_wraparound_enabled_flag」は、水平ラップアラウンド動き補償がインター予測内で適用され得ることを指定する。0 に等しいsps_ref_wraparound_enabled_flagは、水平ラップアラウンド動き補償が適用されないことを指定する。 $(CtbSizeY / MinCbSizeY + 1)$ の値が $(pic_width_max_in_luma_samples / MinCbSizeY - 1)$ を上回る場合、sps_ref_wraparound_enabled_flagの値は、0 に等しいものとする。

【 0 1 4 1 】

[158] 「sps_ref_wraparound_offset_minus1」プラス1は、MinCbSizeYルマサンプル単位の水平ラップアラウンド位置を計算するために使用されるオフセットの最大値を指定する。sps_ref_wraparound_offset_minus1の値は $(CtbSizeY / MinCbSizeY) + 1 \sim (pic_width_max_in_luma_samples / MinCbSizeY) - 1$ の範囲内にあるものとする。

【 0 1 4 2 】

[159] V V Cドラフト7のとおり、pic_width_max_in_luma_samplesは、S P Sを参照する各復号化ピクチャのルマサンプル単位の最大幅である。CtbSizeY及びMinCbSizeYは、V V Cドラフト7で定められているとおりである。

【 0 1 4 3 】

[160] シーケンスの各ピクチャについて、変数「PicRefWraparoundEnableFlag」が以下のように定められる。

【 0 1 4 4 】

[161] $PicRefWraparoundEnableFlag = sps_ref_wraparound_enabled_flag \& \& sps_ref_wraparound_offset_minus1 \& ((pic_width_in_luma_samples / MinCbSizeY - 1))$

【 0 1 4 5 】

[162] ここで、「pic_width_in_luma_samples」は、V V Cドラフト7のとおり、「pic_width_in_luma_samples」がシグナリングされるP P Sを参照するピクチャの幅である。CtbSizeY及びMinCbSizeYは、V V Cドラフト7で定められているとおりである。

【 0 1 4 6 】

[163] 変数「PicRefWraparoundEnableFlag」は、現在のピクチャに対してラップアラウンドMCを有効にできるかどうかを判定するために使用される。

【 0 1 4 7 】

[164] 代替的な方法では、シーケンスの各ピクチャについて、2つの変数「PicRefWraparoundEnableFlag」及び「PicRefWraparoundOffset」が以下のように定められる。

【 0 1 4 8 】

[165] $PicRefWraparoundEnableFlag = sps_ref_wraparound_enabled_flag \& \& ((CtbSizeY / MinCbSizeY + 1) \& ((pic_width_in_luma_samples / MinCbSizeY - 1)))$

【 0 1 4 9 】

[166] $PicRefWraparoundOffset = \min(sps_ref_wraparound_offset_minus1 + 1, (pic_width_in_luma_samples / MinCbSizeY))$

【 0 1 5 0 】

[167] ここで、「pic_width_in_luma_samples」は、V V Cドラフト7のとおり、「pic_width_in_luma_samples」がシグナリングされるP P Sを参照するピクチャの幅である。CtbSizeY及びMinCbSizeYは、V V Cドラフト7で定められているとおりである。

【 0 1 5 1 】

10

20

30

40

50

[168] 変数「PicRefWraparoundEnableFlag」は、現在のピクチャに対してラップアラウンドMCを有効にできるかどうかを判定するために使用される。ラップアラウンドMCを有効にできる場合、オフセット「PicRefWraparoundOffset」を動き補償プロセス内で使用することができる。

【0152】

[169] 一部の実施形態では、ラップアラウンド動き補償及びシーケンスレベルの仮想境界のシグナリングとピクチャ幅の変更との相互排他的な使用を提供する。ラップアラウンド動き補償が有効にされる場合、ピクチャ幅をシーケンス内で変更することができない。ピクチャ幅がシーケンス内で変更される場合、ラップアラウンド動き補償が無効にされる。

【0153】

[170] 一例として、本発明の実施形態に従う「sps_ref_wraparound_enabled_flag」及び「sps_ref_wraparound_offset_minus1」のセマンティクスを以下のとおり記載する。

【0154】

[171] 1に等しい「sps_ref_wraparound_enabled_flag」は、水平ラップアラウンド動き補償がインター予測内で適用されることを指定する。0に等しい「sps_ref_wraparound_enabled_flag」は、水平ラップアラウンド動き補償が適用されないことを指定する。 $(CtbSizeY / MinCbSizeY + 1)$ の値が $(pic_width_max_in_luma_samples / MinCbSizeY - 1)$ を上回る場合、sps_ref_wraparound_enabled_flagの値は、0に等しくてもよい。

【0155】

[172] 「sps_ref_wraparound_offset_minus1」プラス1は、MinCbSizeYルマサンプル単位の水平ラップアラウンド位置を計算するために使用されるオフセットを指定する。「ref_wraparound_offset_minus1」の値は、 $(CtbSizeY / MinCbSizeY) + 1 \sim (pic_width_max_in_luma_samples / MinCbSizeY) - 1$ の範囲内にあり得る。

【0156】

[173] 「pic_width_in_luma_samples」及び「pic_height_in_luma_samples」のビットストリーム適合要件は、以下のように課すことができる。

【0157】

[174] 「subpics_present_flag」が1に等しいか、又は「ref_pic_resampling_enabled_flag」が0に等しいか、又は「sps_ref_wraparound_enabled_flag」が1に等しい場合、「pic_width_in_luma_samples」の値は、「pic_width_max_in_luma_samples」に等しい。

【0158】

[175] 別の例として、「sps_ref_wraparound_enabled_flag」のビットストリーム適合要件を課すことができる。本発明の実施形態に従う「sps_ref_wraparound_enabled_flag」及び「sps_ref_wraparound_offset_minus1」のセマンティクスを以下のとおり記載する。

【0159】

[176] 1に等しい「sps_ref_wraparound_enabled_flag」は、水平ラップアラウンド動き補償がインター予測内で適用されることを指定する。0に等しい「sps_ref_wraparound_enabled_flag」は、水平ラップアラウンド動き補償が適用されないことを指定する。 $(CtbSizeY / MinCbSizeY + 1)$ の値が $(pic_width_max_in_luma_samples / MinCbSizeY - 1)$ を上回る場合、sps_ref_wraparound_enabled_flagの値は、0に等しい。

【0160】

[177] ref_pic_resampling_enabled_flagが1である場合、「sps_ref_wraparound_enabled_flag」の値は、0であり得る。ref_pic_resampling_enabled_flagは、参照再サンプリングが有効にされるかどうかを指定する。参照再サンプリングは、参照ピクチャの解像度が現在のピクチャの解像度と異なる場合に参照ピクチャを再サンプリングす

10

20

30

40

50

るために使用される。そのため、ピクチャレベル（例えば、PPS、PH）では、参照ピクチャの解像度が現在のピクチャと異なる場合、現在のピクチャを予測するためにラップアラウンド動き補償は使用されない。

【0161】

[178] 「sps_ref_wraparound_offset_minus1」プラス1は、MinCbSizeYルマサンプル単位の水平ラップアラウンド位置を計算するために使用されるオフセットを指定する。「ref_wraparound_offset_minus1」の値は、 $(CtbSizeY / MinCbSizeY) + 1 \sim (pic_width_max_in_luma_samples / MinCbSizeY) - 1$ の範囲内にある。

【0162】

[179] 別の例として、「sps_ref_wraparound_enabled_flag」は、「ref_pic_resampling_enabled_flag」に基づいて条件付きでシグナリングされる。この構文を図13に示し、表3（図8）内の構文に対する変更をイタリック体で示す。関連するセマンティクスを以下のとおり記載する。

【0163】

[180] 1に等しい「sps_ref_wraparound_enabled_flag」は、水平ラップアラウンド動き補償がインター予測内で適用されることを指定する。0に等しい「sps_ref_wraparound_enabled_flag」は、水平ラップアラウンド動き補償が適用されないことを指定する。 $(CtbSizeY / MinCbSizeY + 1)$ の値が $(pic_width_max_in_luma_samples / MinCbSizeY - 1)$ を上回る場合、「sps_ref_wraparound_enabled_flag」の値は、0に等しい。「sps_ref_wraparound_enabled_flag」がない場合、その値は、0であると推論される。

【0164】

[181] 「sps_ref_wraparound_offset_minus1」プラス1は、MinCbSizeYルマサンプル単位の水平ラップアラウンド位置を計算するために使用されるオフセットを指定する。ref_wraparound_offset_minus1の値は $(CtbSizeY / MinCbSizeY) + 1 \sim (pic_width_max_in_luma_samples / MinCbSizeY) - 1$ の範囲内にある。

【0165】

[182] 一例として、「ref_pic_resampling_enabled_flag」は、「sps_ref_wraparound_enabled_flag」に基づいて条件付きでシグナリングされる。この構文を図14の表9として示し、表3（図8）内の構文に対する変更を、イタリック体及び取り消し線を用いて示す。関連するセマンティクスは、以下のとおりである。

【0166】

[183] 1に等しい「ref_pic_resampling_enabled_flag」は、SPSを参照するCLVS内の符号化ピクチャを復号化するとき、参照ピクチャ再サンプリングを適用できることを指定する。0に等しいref_pic_resampling_enabled_flagは、SPSを参照するCLVS内のピクチャを復号化するとき、参照ピクチャ再サンプリングが適用されないことを指定する。ref_pic_resampling_enabled_flagがない場合、その値は、0であると推論される。

【0167】

[184] 図15は、本開示のいくつかの実施形態に従う例示的な方法1500のフローチャートを示す。一部の実施形態では、方法1500は、符号器、機器（例えば、図4の機器400）の1つ以上のソフトウェア又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1500を実行することができる。一部の実施形態では、方法1500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。

【0168】

[185] 誤り及びアーティファクトを防ぐためにインループフィルタリングを無効にすることができるように、360度映像又は漸進的復号化リフレッシュ（GDR）符号化のた

10

20

30

40

50

めの符号化ツールの1つとして仮想境界を設定することができる。これを、図15に示す方法で例示する。

【0169】

[186] ステップ1501では、ピクチャのシーケンスを含むビットストリームを受信する。ピクチャのシーケンスは、ピクチャのセットであり得る。記載したように、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。

10

【0170】

[187] ステップ1503では、受信されたストリームに従い、シーケンスレベルで(すなわちSPS内で)仮想境界がシグナリングされるかどうかを判定する。仮想境界がシーケンスレベルでシグナリングされる場合、仮想境界の数及び各仮想境界の位置が、SPSを参照するピクチャについてシグナリングされる。

【0171】

[188] 仮想境界は、シーケンスパラメータセット(SPS)又はピクチャヘッダ(PH)内でシグナリングされ得る。SPSでは、仮想境界存在フラグ「sps_virtual_boundaries_present_flag」が最初にシグナリングされる。フラグが真である場合、仮想境界の数及び各仮想境界の位置が、SPSを参照するピクチャについてシグナリングされる。

20

【0172】

[189] ステップ1505では、任意選択的に、仮想境界がピクチャのセットについてシーケンスレベルでシグナリングされないことに応答して、受信されたストリームに従い、ピクチャのセット内のピクチャについて仮想境界がピクチャレベル(例えば、PPS又はPH)でシグナリングされるかどうかを判定する。例えば、「sps_virtual_boundaries_present_flag」が偽である場合、別の仮想境界存在フラグ「ph_virtual_boundaries_present_flag」がPH内でシグナリングされ得る。同様に、「ph_virtual_boundaries_present_flag」が真である場合、仮想境界の数及び各仮想境界の位置が、PHに関連するピクチャについてシグナリングされ得る。ステップ1505は、任意選択的であり、一部の実施形態では、本方法は、仮想境界がシーケンスレベルでシグナリングされないことに応答してステップ1505を含み得る。一部の実施形態では、仮想境界がシーケンスレベルでシグナリングされないことに応答して、本方法は、ステップ1505を行うことなく終了することができる。仮想境界がピクチャレベルでシグナリングされることに応答して、本方法は、仮想境界の位置を決定するためのステップ1507に進む。

30

【0173】

[190] 一部の実施形態では、仮想境界がシーケンスレベルでシグナリングされていないことを示す任意の条件が満たされているかどうかを判定することができる。例で上述したように、図5の表1内の「sps_virtual_boundaries_present_flag」の値を決定することができる。値が1であることは、仮想境界がシーケンスレベルでシグナリングされることを示す。しかし、仮想境界がシーケンスレベルで適用されないことを示す他の条件がある。シーケンスレベルの仮想境界のシグナリングがないことを示す第1の条件は、参照再サンプリングが有効にされることである。第2の条件は、ピクチャのセット内のピクチャの解像度の変更が許可されることである。これらの2つの条件のいずれかが満たされる場合、仮想境界は、シーケンスレベルでシグナリングされない。これらの2つの条件のいずれも満たされない場合、及び本実施例にあるように、「sps_virtual_boundaries_present_flag」の値が1であると決定される場合、仮想信号がピクチャのセットについてシーケンスレベルでシグナリングされると決定される。この方法は、仮想境界の位置を決定するためのステップ1507に進む。

40

50

【 0 1 7 4 】

[191] 一部の実施形態では、受信されたビットストリームに従い、ピクチャのシーケンスに関して参照再サンプリングが有効にされるかどうかを判定する。ピクチャのシーケンスに関して参照再サンプリングが有効にされることに応答して、仮想境界がシーケンスレベルでシグナリングされない。

【 0 1 7 5 】

[192] 復号化プログラムは、フレームのそれぞれを復号化することができる。現在のフレームの解像度が参照ピクチャの解像度と異なると復号化プログラムが判定する場合、復号化プログラムは、参照ピクチャに対して適切な再サンプリングを行い、生成した再サンプリングされた参照ブロックを、現在のフレーム内のブロックを復号化するための参照ブロックとして使用することができる。ピクチャの空間解像度が映像シーケンス内で変更されることが許可される場合、参照ピクチャの再サンプリングが必要である。再サンプリングが有効にされる場合、解像度の変更を許可しても又はしなくてもよい。ピクチャの解像度の変更が許可される場合、再サンプリングが必要であるため、再サンプリングが有効にされる。参照ピクチャの適切な再サンプリングは、参照ピクチャのダウンサンプリング又はアップサンプリングであり得る。例えば、図 9 に示すように、1 に等しい「ref_pic_resampling_enabled_flag」は、S P S を参照する C L V S 内の符号化ピクチャを復号化するとき、参照ピクチャ再サンプリングを適用できることを指定する。0 に等しいref_pic_resampling_enabled_flagは、S P S を参照する C L V S 内のピクチャを復号化するとき、参照ピクチャ再サンプリングが適用されないことを指定する。

【 0 1 7 6 】

[193] 例えば、図 5 の表 1 内の「sps_virtual_boundaries_present_flag」は、「ref_pic_resampling_enabled_flag」に基づいて条件付きでシグナリングされ得る。「ref_pic_resampling_enabled_flag」の値を決定することができる。0 に等しい「ref_pic_resampling_enabled_flag」は、S P S を参照する C L V S 内のピクチャを復号化するとき、参照ピクチャ再サンプリングが適用されないことを指定する。「ref_pic_resampling_enabled_flag」が 1 である場合、「sps_virtual_boundaries_present_flag」は、0 である。

【 0 1 7 7 】

[194] 一部の実施形態では、ピクチャのシーケンス内のピクチャの解像度の変更が許可されるかどうかを判定する。解像度の変更が許可されることに応答して、仮想境界がシーケンスレベルでシグナリングされないと判定される。

【 0 1 7 8 】

[195] 参照ピクチャ再サンプリングは、現在のピクチャが、参照ピクチャと異なる解像度を有する場合に使用される。ピクチャの解像度がシーケンス内で変更されることが許可される場合、異なるピクチャは、異なる解像度を有し得るため、仮想境界の位置もピクチャごとに変更することができる。従って、仮想境界の位置をピクチャレベル（例えば、P P S 及び P H ）でシグナリングすることができる一方、シーケンスレベルの（すなわち S P S 内でシグナリングされる）仮想境界のシグナリングは、適切ではない。従って、ピクチャの解像度の変更が許可される場合、仮想境界がシーケンスレベルでシグナリングされないと判定される。同時に、再サンプリングが許可されるが、ピクチャの解像度の変更が許可されない場合、仮想境界のシグナリングに対する制約は、課されない。

【 0 1 7 9 】

[196] 一部の実施形態では、ピクチャのシーケンスに関して参照再サンプリングが有効にされることに応答して、ピクチャのシーケンスに関してラップアラウンド動き補償が無効にされると判定される。

【 0 1 8 0 】

[197] 例えば、図 1 3 内の「ref_pic_resampling_enabled_flag」の値が 1 であり、「sps_ref_wraparound_enabled_flag」が 0 であり、水平ラップアラウンド動き補償が適用されないことが決定される。「ref_pic_resampling_enabled_flag」は、参照再

サンプリングが有効にされるかどうかを指定する。参照再サンプリングは、参照ピクチャの解像度が現在のピクチャの解像度と異なる場合に参照ピクチャを再サンプリングするために使用される。ピクチャレベル（例えば、P P S）において、この例では参照ピクチャの解像度が現在のピクチャと異なる場合、現在のピクチャを予測するためにラップアラウンド動き補償が使用されない。「ref_pic_resampling_enabled_flag」の値が0であると判定される場合、「sps_ref_wraparound_enabled_flag」は1であり、水平ラップアラウンド動き補償が適用される。

【0181】

[198] 一部の実施形態では、ピクチャのシーケンス内のピクチャの解像度の変更が許可されることに応答して、ピクチャのシーケンスに関してラップアラウンド動き補償が無効にされると判定される。

10

【0182】

[199] ステップ1507では、仮想境界がシーケンスレベルでシグナリングされることに応答して、シーケンスの仮想境界の位置を決定する。位置は、シーケンス内のピクチャの幅及び高さに基づく範囲によって制限される。仮想境界は、垂直境界及び水平境界を含み得る。仮想境界の位置は、垂直境界の位置点及び水平境界の位置点を含み得る。例えば、図5の表1に示すように、「sps_virtual_boundaries_pos_x[i]」は、8で割ったルマサンプル単位のi番目の垂直仮想境界の位置を指定する一方、「sps_virtual_boundaries_pos_y[i]」は、8で割ったルマサンプル単位のi番目の水平仮想境界の位置を指定する。

20

【0183】

[200] 一部の実施形態では、この位置の垂直範囲は、シーケンス内のピクチャの各ピクチャについて許可された最大幅以下であり、水平範囲は、セット内のピクチャのそれぞれについて許可された最大高さ以下である。最大幅及び最大高さは、受信されたストリーム内でシグナリングされる。セマンティクスにおける例は、以下のとおりである。

【0184】

[201] 「sps_virtual_boundaries_pos_x[i]」は、8で割ったルマサンプル単位のi番目の垂直仮想境界の位置を指定する。「sps_virtual_boundaries_pos_x[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_width_max_in_luma_samples} \div 8) - 1$ の範囲内にある。

【0185】

30

[202] 「sps_virtual_boundaries_pos_y[i]」は、8で割ったルマサンプル単位のi番目の水平仮想境界の位置を指定する。「sps_virtual_boundaries_pos_y[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_height_max_in_luma_samples} \div 8) - 1$ の範囲内にある。

【0186】

[203] 仮想境界がS P S内にある、すなわち「sps_virtual_boundaries_present_flag」が1に等しい場合、シーケンス内の各ピクチャは、シーケンス内のピクチャの最大幅に等しい同じ幅を有する。

【0187】

[204] 一部の実施形態では、ピクチャのセット内の1つのピクチャについて仮想境界がピクチャレベルでシグナリングされることに応答して、その1つのピクチャの仮想境界の位置を決定することができる。セット内の1つ以上のピクチャに関して1つ以上の境界をシグナリングすることができる。例えば、図6のP H構文のセマンティクスに示すように、「ph_num_ver_virtual_boundaries」は、P H内にある「ph_virtual_boundaries_pos_x[i]」構文要素の数を指定する。「ph_num_ver_virtual_boundaries」がない場合、その値は、0に等しいと推論される。

40

【0188】

[205] 「ph_virtual_boundaries_pos_x[i]」は、8で割ったルマサンプル単位のi番目の垂直仮想境界の位置を指定する。「ph_virtual_boundaries_pos_x[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_width_in_luma_samples} \div 8) - 1$ の範囲内にある。

【0189】

50

[206] 「ph_virtual_boundaries_pos_y[i]」は、8で割ったルマサンプル単位の i 番目の水平仮想境界の位置を指定する。「ph_virtual_boundaries_pos_y[i]」の値は、 $1 \sim \text{Ceil}(\text{pic_height_in_luma_samples} \div 8) - 1$ の範囲内にある。

【0190】

[207] 記載したように、360度映像などのアプリケーションでは、特定の投影フォーマットのレイアウトが複数の面を通常有する。例えば、MPEG-Iパート2：全方向性メディアフォーマット(OMAF)は、面を6つ有するCMPと名付けられたキューブマップベースの投影フォーマットを規格化する。複数の面を含む投影フォーマットに関して、どのような種類のコンパクトフレームパッキング構成が使用されるかに関係なく、フレームバックピクチャ内の2つ以上の隣接する面間に不連続性が生じる。この不連続性にわたってインループフィルタリング演算を実行した場合、レンダリング後の再構成映像内で面の継ぎ目のアーティファクトが見えるようになり得る。面の継ぎ目のアーティファクトを軽減するために、フレームバックピクチャ内の不連続性にわたるインループフィルタリング演算を無効にしなければならない。横断するようなループフィルタリング演算が無効にされる仮想境界を使用することができる。符号器は、不連続境界を仮想境界として設定することができ、それにより不連続境界上でループフィルタを適用しないことができる。360度映像の他に、仮想境界は、超低遅延アプリケーション内で主に使用される漸進的復号化リフレッシュ(GDR)にも使用することができる。超低遅延アプリケーションでは、ランダムアクセスポイントピクチャとしてイントラ符号化ピクチャを挿入することは、イントラ符号化ピクチャのサイズが大きくなることにより、許容できない伝送レイテンシを引き起こす場合がある。レイテンシを減らすために、B/Pピクチャ内にイントラ符号化領域を挿入することによってピクチャが漸進的にリフレッシュされるGDRが採用される。誤り伝播を防ぐために、ピクチャ内のリフレッシュ領域内のピクセルは、現在のピクチャ又は参照ピクチャの未リフレッシュ領域内のものを参照することができない。従って、リフレッシュ領域及び未リフレッシュ領域の境界を横断するループフィルタリングを適用することができない。上記の仮想境界の方式により、符号器は、リフレッシュ領域及び未リフレッシュ領域の境界を仮想境界として設定することができ、従ってその境界を横断してループフィルタリング演算を適用することができない。

【0191】

[208] ステップ1509では、ピクチャの仮想境界を横断するインループフィルタリング演算(例えば、図3A内のルックフィルタ段階232)を無効にする。1つ以上の仮想境界がSPS内でシグナリングされる場合、SPSを参照するピクチャ内の仮想境界を横断するインループフィルタリング演算が無効にされ得る。インループフィルタリング演算は、デブロッキングフィルタ、サンプル適応オフセットフィルタ、及び適応ループフィルタ演算を含む。

【0192】

[209] 本開示のいくつかの実施形態に従い、別の例示的方法1600を提供する。この方法は、符号器、機器(例えば、図4の機器400)の1つ以上のソフトウェア又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1600を実行することができる。一部の実施形態では、方法1600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。この方法は、以下のステップを含むことができる。

【0193】

[210] ステップ1601では、ピクチャのシーケンスを含むビットストリームを受信する。ピクチャのシーケンスは、ピクチャのセットであり得る。記載したように、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

10

20

30

40

50

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。

【0194】

[211] ステップ1603では、受信されたビットストリームに従い、ピクチャのシーケンスに関して参照再サンプリングが有効にされるかどうかを判定する。

【0195】

[212] ステップ1605では、ピクチャのシーケンスに関して参照再サンプリングが有効にされることに応答して、ピクチャのシーケンスに関してラップアラウンド動き補償が無効にされると判定する。

10

【0196】

[213] 例えば、図13内の「ref_pic_resampling_enabled_flag」の値が1であり、「sps_ref_wraparound_enabled_flag」が0であり、水平ラップアラウンド動き補償が適用されないことが決定される。「ref_pic_resampling_enabled_flag」は、参照再サンプリングが有効にされるかどうかを指定する。参照再サンプリングは、参照ピクチャの解像度が現在のピクチャの解像度と異なる場合に、参照ピクチャを再サンプリングするために使用される。ピクチャレベル（例えば、PPS）において、この例では参照ピクチャの解像度が現在のピクチャと異なる場合、現在のピクチャを予測するためにラップアラウンド動き補償が使用されない。「ref_pic_resampling_enabled_flag」の値が0であると判定される場合、「sps_ref_wraparound_enabled_flag」は1であり、水平ラップアラウンド動き補償が適用される。

20

【0197】

[214] 従って、ステップ1603の代替的ステップであるステップ1703では、ピクチャのシーケンスの参照ピクチャの解像度がピクチャのシーケンスの現在のピクチャの解像度と異なるかどうかを判定する。

【0198】

[215] ステップ1605の代替的ステップであるステップ1705では、ピクチャのシーケンスの参照ピクチャの解像度が、ピクチャのシーケンスの現在のピクチャの解像度と異なることに応答して、現在のピクチャに関してラップアラウンド動き補償が無効にされると判定する。

30

【0199】

[216] 一部の実施形態では、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、CD-ROM、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、RAM、PROM及びEPROM、FLASH（登録商標）-EPROM、又は任意の他のフラッシュメモリ、NVRAM、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（CPU）、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。

40

【0200】

[217] 開示する実施形態は、以下の条項を用いてさらに記述することができる：

1. ピクチャのセットを受信すること、
セット内のピクチャの幅及び高さを決定すること、及び
ピクチャのセットに関する仮想境界の位置を前記幅及び高さに基づいて決定することを含む方法。

2. ピクチャの決定された幅及び高さの最小値を決定すること

50

をさらに含み、ピクチャのセットに関する仮想境界の位置を前記幅及び高さに基づいて決定することは、

ピクチャのセットに関する仮想境界の位置を前記最小値に基づいて決定すること
をさらに含む、条項 1 に記載の方法。

3 . ピクチャの決定された幅及び高さの最大値を決定すること
をさらに含み、ピクチャのセットに関する仮想境界の位置を前記幅及び高さに基づいて決定することは、

ピクチャのセットに関する仮想境界の位置を前記最大値に基づいて決定すること
をさらに含む、条項 1 に記載の方法。

4 . セット内の第 1 のピクチャの幅が所与の条件を満たすかどうかを判定すること、及び 10
第 1 のピクチャの幅が所与の条件を満たすという判定に応答して、第 1 のピクチャに対する水平ラップアラウンド動き補償を無効にすること

をさらに含む、条項 1 に記載の方法。

5 . ピクチャのセットの仮想境界を横断するインループフィルタリング演算を無効にすること

をさらに含む、条項 1 に記載の方法。

6 . セット内の第 1 のピクチャに対して再サンプリングが行われているかどうかを判定すること、

第 1 のピクチャに対する再サンプリングが行われていないという判定に応答して、仮想境界又はラップアラウンド動き補償のシグナリングの少なくとも 1 つを第 1 のピクチャに 20
関して無効にすること

をさらに含む、条項 1 に記載の方法。

7 . 仮想境界の位置は、シーケンスパラメータセット (S P S) 又はピクチャヘッダ (P H) の少なくとも 1 つにおいてシグナリングされる、条項 1 に記載の方法。

8 . 命令のセットを記憶するメモリと、

1 つ以上のプロセッサとを含み、1 つ以上のプロセッサは、

ピクチャのセットを受信すること、

セット内のピクチャの幅及び高さを決定すること、及び

ピクチャのセットに関する仮想境界の位置を前記幅及び高さに基づいて決定すること

を機器に行わせるために命令のセットを実行するように構成される、機器。 30

9 . 1 つ以上のプロセッサは、

ピクチャの決定された幅及び高さの最小値を決定すること

を機器にさらに行わせるために命令のセットを実行するように構成され、ピクチャのセットに関する仮想境界の位置を前記幅及び高さに基づいて決定することは、

ピクチャのセットに関する仮想境界の位置を前記最小値に基づいて決定すること

をさらに含む、条項 8 に記載の機器。

10 . 1 つ以上のプロセッサは、

ピクチャの決定された幅及び高さの最大値を決定すること

を機器にさらに行わせるために命令のセットを実行するように構成され、ピクチャのセットに関する仮想境界の位置を前記幅及び高さに基づいて決定することは、 40

ピクチャのセットに関する仮想境界の位置を前記最大値に基づいて決定すること

をさらに含む、条項 8 に記載の機器。

11 . 1 つ以上のプロセッサは、

セット内の第 1 のピクチャの幅が所与の条件を満たすかどうかを判定すること、及び

第 1 のピクチャの幅が所与の条件を満たすという判定に応答して、第 1 のピクチャに対する水平ラップアラウンド動き補償を無効にすること

を機器にさらに行わせるために命令のセットを実行するように構成される、条項 8 に記載の機器。

12 . 1 つ以上のプロセッサは、

ピクチャのセットの仮想境界にわたるインループフィルタリング演算を無効にすること 50

を機器にさらに行わせるために命令のセットを実行するように構成される、条項 8 に記載の機器。

13. 1つ以上のプロセッサは、

セット内の第1のピクチャに対して再サンプリングが行われているかどうかを判定すること、

第1のピクチャに対する再サンプリングが行われていないという判定に応答して、仮想境界又はラップアラウンド動き補償のシグナリングの少なくとも1つを第1のピクチャに関して無効にすること

を機器にさらに行わせるために命令のセットを実行するように構成される、条項 8 に記載の機器。

10

14. 仮想境界の位置は、シーケンスパラメータセット (SPS) 又はピクチャヘッダ (PH) の少なくとも1つにおいてシグナリングされる、条項 8 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媒体であって、命令のセットは、映像処理方法をコンピュータに行わせるためにコンピュータの少なくとも1つのプロセッサによって実行可能であり、方法は、

ピクチャのセットを受信すること、

セット内のピクチャの幅及び高さを決定すること、及び

ピクチャのセットに関する仮想境界の位置を前記幅及び高さに基づいて決定することを含む、非一時的コンピュータ可読媒体。

16. 命令のセットは、

20

ピクチャの決定された幅及び高さの最小値を決定すること

をコンピュータにさらに行わせるためにコンピュータによって実行可能であり、ピクチャのセットに関する仮想境界の位置を前記幅及び高さに基づいて決定することは、

ピクチャのセットに関する仮想境界の位置を前記最小値に基づいて決定すること

をさらに含む、条項 15 に記載の非一時的コンピュータ可読媒体。

17. 命令のセットは、

ピクチャの決定された幅及び高さの最大値を決定すること

をコンピュータにさらに行わせるためにコンピュータによって実行可能であり、ピクチャのセットに関する仮想境界の位置を前記幅及び高さに基づいて決定することは、

ピクチャのセットに関する仮想境界の位置を前記最大値に基づいて決定すること

をさらに含む、条項 15 に記載の非一時的コンピュータ可読媒体。

30

18. 命令のセットは、

セット内の第1のピクチャの幅が所与の条件を満たすかどうかを判定すること、及び

第1のピクチャの幅が所与の条件を満たすという判定に応答して、第1のピクチャに対する水平ラップアラウンド動き補償を無効にすること

をコンピュータにさらに行わせるためにコンピュータによって実行可能である、条項 15 に記載の非一時的コンピュータ可読媒体。

19. 命令のセットは、

ピクチャのセットの仮想境界にわたるインループフィルタリング演算を無効にすること

をコンピュータにさらに行わせるためにコンピュータによって実行可能である、条項 15 に記載の非一時的コンピュータ可読媒体。

40

20. 命令のセットは、

セット内の第1のピクチャに対して再サンプリングが行われているかどうかを判定すること、

第1のピクチャに対する再サンプリングが行われていないという判定に応答して、仮想境界又はラップアラウンド動き補償のシグナリングの少なくとも1つを第1のピクチャに関して無効にすること

をコンピュータにさらに行わせるためにコンピュータによって実行可能である、条項 15 に記載の非一時的コンピュータ可読媒体。

21. 仮想境界の位置は、シーケンスパラメータセット (SPS) 又はピクチャヘッダ

50

(PH)の少なくとも1つにおいてシグナリングされる、条項15に記載の非一時的コンピュータ可読媒体。

【0201】

[218] 「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举であることを意味されないか、又は列举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。

10

【0202】

[219] 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。

【0203】

20

[220] 上述の実施形態は、ハードウェア、又はソフトウェア(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、又はソフトウェア、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することも理解するであろう。

【0204】

30

[221] 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、例示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。

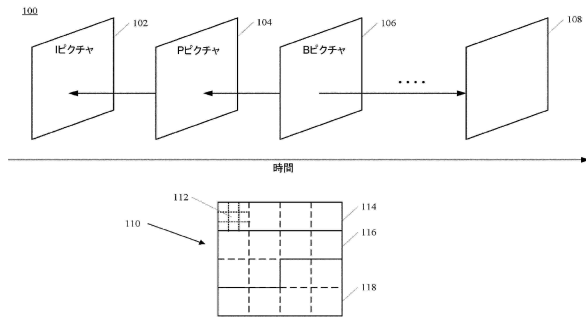
【0205】

40

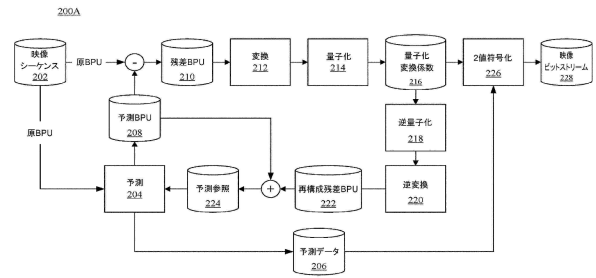
[222] 図面及び本明細書において、例示的な実施形態が開示された。しかし、これらの実施形態に対する多くの変形形態及び変更形態がなされ得る。従って、特定の用語が採用されていても、これらは、単に一般的な説明の意味で使用されているにすぎず、限定を目的として使用されているものではない。

【図面】

【図 1】

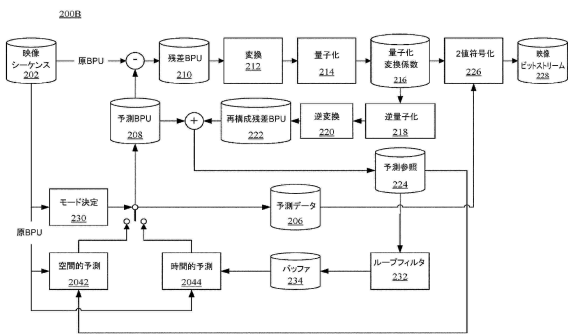


【図 2 A】

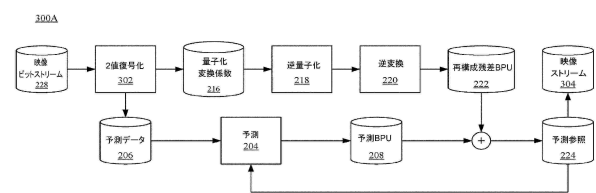


10

【図 2 B】



【図 3 A】



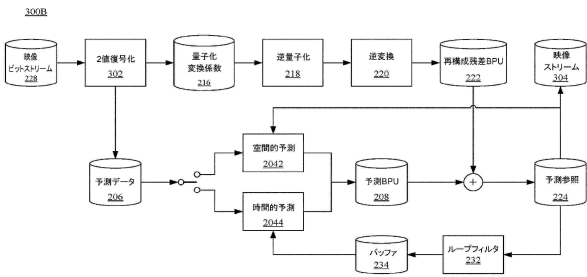
20

30

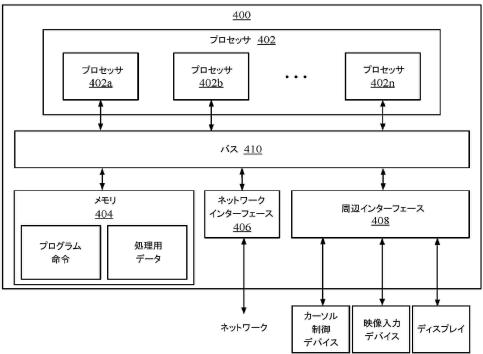
40

50

【図 3 B】



【図 4】



【図 5】

表1:仮想境界のための例示的なSPS構文

seq_parameter_set_rbsp() {	記述子
....	
sps_virtual_boundaries_present_flag	u(1)
if(sps_virtual_boundaries_present_flag) {	
sps_num_ver_virtual_boundaries	u(2)
for(i = 0; i < sps_num_ver_virtual_boundaries; i++)	
sps_virtual_boundaries_pos_x[i]	u(13)
sps_num_hor_virtual_boundaries	u(2)
for(i = 0; i < sps_num_hor_virtual_boundaries; i++)	
sps_virtual_boundaries_pos_y[i]	u(13)
}	

【図 6】

表2:仮想境界のための例示的なPH構文

picture_header_rbsp() {	記述子
...	
if(!sps_virtual_boundaries_present_flag) {	
ph_virtual_boundaries_present_flag	u(1)
if(ph_virtual_boundaries_present_flag) {	
ph_num_ver_virtual_boundaries	u(2)
for(i = 0; i < ph_num_ver_virtual_boundaries; i++)	
ph_virtual_boundaries_pos_x[i]	u(13)
ph_num_hor_virtual_boundaries	u(2)
for(i = 0; i < ph_num_hor_virtual_boundaries; i++)	
ph_virtual_boundaries_pos_y[i]	u(13)
}	
}	
}	

10

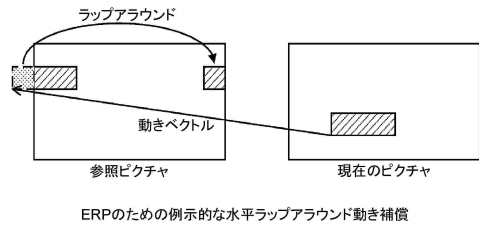
20

30

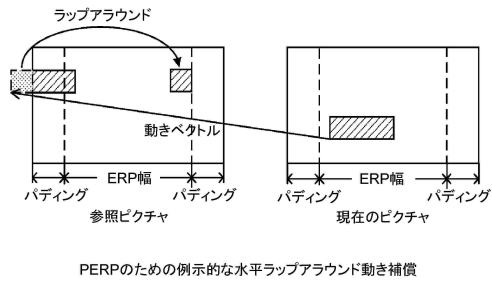
40

50

【図 7 A】



【図 7 B】



10

【図 8】

表3:ラップアラウンド動き補償のための例示的な構文

seq_parameter_set_rbsp() {	記述子
...	
sps_ref_wraparound_enabled_flag	u(1)
if(sps_ref_wraparound_enabled_flag)	
sps_ref_wraparound_offset_minus1	ue(v)
...	u(1)

【図 9】

表4:ピクチャの幅及び高さの最大値を信号通知するための例示的なSPS構文

seq_parameter_set_rbsp() {	記述子
...	
ref_pic_resampling_enabled_flag	u(1)
pic_width_max_in_luma_samples	ue(v)
pic_height_max_in_luma_samples	ue(v)
...	u(2)

20

30

40

50

【図 1 0】

表5:ピクチャの幅及び高さを信号通知するための例示的なPPS構文

pic_parameter_set_rbsp() {	記述子
...	
pic_width_in_luma_samples	ue(v)
pic_height_in_luma_samples	ue(v)
...	

【図 1 1】

表6:例示的なSPS構文

seq_parameter_set_rbsp() {	記述子
...	
ref_pic_resampling_enabled_flag	u(1)
....	
if(!ref_pic_resampling_enabled_flag){	
sps_virtual_boundaries_present_flag	u(1)
if(sps_virtual_boundaries_present_flag) {	
sps_num_ver_virtual_boundaries	u(2)
for(i = 0; i < sps_num_ver_virtual_boundaries; i++)	
sps_virtual_boundaries_pos_x[i]	u(13)
sps_num_hor_virtual_boundaries	u(2)
for(i = 0; i < sps_num_hor_virtual_boundaries; i++)	
sps_virtual_boundaries_pos_y[i]	u(13)
}	
}	

10

20

【図 1 2】

表7:例示的なSPS構文

seq_parameter_set_rbsp() {	記述子
...	
ref_pic_resampling_enabled_flag	u(1)
....	
sps_virtual_boundaries_present_flag	
if(sps_virtual_boundaries_present_flag) {	u(1)
sps_num_ver_virtual_boundaries	
for(i = 0; i < sps_num_ver_virtual_boundaries; i++)	u(2)
sps_virtual_boundaries_pos_x[i]	
sps_num_hor_virtual_boundaries	u(13)
for(i = 0; i < sps_num_hor_virtual_boundaries; i++)	u(2)
sps_virtual_boundaries_pos_y[i]	
}	u(13)
...	
if(!sps_virtual_boundaries_present_flag)	
ref_pic_resampling_enabled_flag	u(1)

【図 1 3】

表8:例示的なSPS構文

seq_parameter_set_rbsp() {	記述子
...	
ref_pic_resampling_enabled_flag	u(1)
....	
if(!ref_pic_resampling_enabled_flag){	
sps_ref_wraparound_enabled_flag	u(1)
if(sps_ref_wraparound_enabled_flag)	
sps_ref_wraparound_offset_minus1	ue(v)
}	

30

40

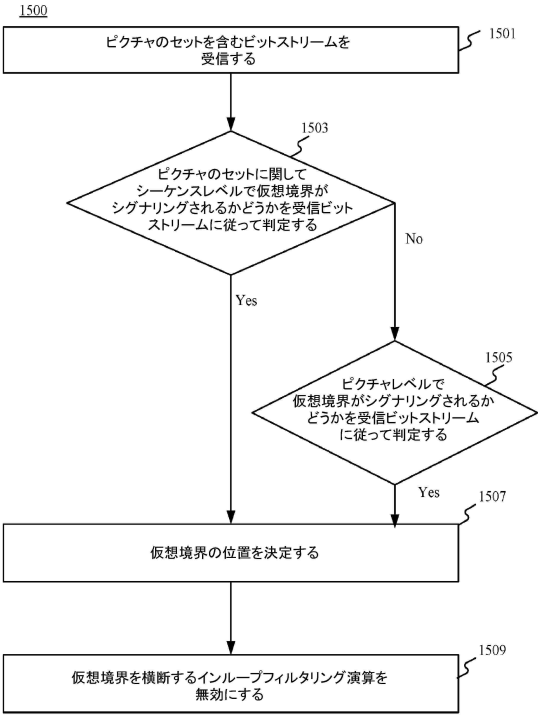
50

【図 14】

表9:例示的なSPS構文

seq_parameter_set_rbsp() {	記述子
...	
ref_pic_resampling_enabled_flag	u(1)
....	
sps_ref_wraparound_enabled_flag	u(1)
if(sps_ref_wraparound_enabled_flag)	
sps_ref_wraparound_offset_minus1	ue(v)
...	
if(!sps_ref_wraparound_enabled_flag)	
ref_pic_resampling_enabled_flag	u(1)

【図 15】



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, ベイジン 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 リャオ, ル - リン
 中華人民共和国, ベイジン 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 フィ タワー
- (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
- 審査官 松永 隆志
- (56)参考文献 国際公開第 2 0 1 9 / 0 6 0 4 4 3 (W O , A 1)
 特表 2 0 1 9 - 5 2 5 5 6 3 (J P , A)
 特表 2 0 1 8 - 5 3 4 8 2 7 (J P , A)
 米国特許出願公開第 2 0 1 8 / 0 0 5 4 6 1 3 (U S , A 1)
- (58)調査した分野 (Int.Cl., D B 名)
 H 0 4 N 1 9 / 7 0
 H 0 4 N 1 9 / 1 0 5
 H 0 4 N 1 9 / 1 7 4