

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7575478号
(P7575478)

(45)発行日 令和6年10月29日(2024.10.29)

(24)登録日 令和6年10月21日(2024.10.21)

(51)国際特許分類	F I
H 0 4 N 19/13 (2014.01)	H 0 4 N 19/13
H 0 4 N 19/169 (2014.01)	H 0 4 N 19/169 2 0 0
H 0 4 N 19/18 (2014.01)	H 0 4 N 19/18
H 0 4 N 19/70 (2014.01)	H 0 4 N 19/70

請求項の数 18 (全61頁)

(21)出願番号 特願2022-559412(P2022-559412)	(73)特許権者 520476341
(86)(22)出願日 令和3年4月1日(2021.4.1)	北京字節跳動網絡技術有限公司
(65)公表番号 特表2023-519402(P2023-519402 A)	Beijing Bytedance Network Technology Co., Ltd.
(43)公表日 令和5年5月10日(2023.5.10)	中華人民共和國100041北京市石景山区実興大街30号院3号楼2層B-0035房間
(86)国際出願番号 PCT/CN2021/084869	Room B-0035, 2/F, No.3 Building, No.30, Shixing Road, Shijingshan District Beijing 100041 China
(87)国際公開番号 WO2021/197406	
(87)国際公開日 令和3年10月7日(2021.10.7)	
審査請求日 令和4年10月31日(2022.10.31)	
(31)優先権主張番号 PCT/CN2020/082983	(73)特許権者 520477474
(32)優先日 令和2年4月2日(2020.4.2)	バイトダンス インコーポレイテッド
(33)優先権主張国・地域又は機関 中国(CN)	最終頁に続く

(54)【発明の名称】 変換スキップモードにおける係数コーディング

(57)【特許請求の範囲】

【請求項1】

映像処理方法であって、
 映像の現在のブロックと前記映像のビットストリームとの間の変換を実行すること、を有し、
 前記ビットストリームは、前記ビットストリームにおける第1の係数の第1の符号フラグを含むためにコンテキストインクリメントが使用されることを規定する第1の規則に準拠し、
 前記第1の規則は、前記コンテキストインクリメントの値が前記現在のブロックに第1のコーディングモードが適用されるか否かに基づくことを規定し、
 前記ビットストリームは、前記映像に複数の適応パラメータセットネットワーク抽象化レイヤユニットを使用することを規定する第2の規則に準拠し、
 各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する適応パラメータセットタイプ値を有し、
 各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する映像レイヤ識別子に関連付けられ、
 各適応パラメータセットネットワーク抽象化レイヤユニットは、プレフィックスユニットまたはサフィックスユニットであり、
 前記第2の規則は、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットが、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットの複数の映像レ

イヤ識別子とは独立して、および前記複数の適応パラメータセットネットワーク抽象化レイヤユニットが前記プレフィックスユニットまたは前記サフィックスユニットであるか否かとは独立して、特定の適応パラメータセットタイプ値を有することに応じて、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットの適応パラメータセット識別子値が同じ適応パラメータセット識別子値空間に属することを規定する、

方法。

【請求項 2】

前記第 1 の規則は、前記第 1 のコーディングモードが適用されない場合、前記コンテキストインクリメントの前記値が $N_0 \sim N_1$ の第 1 の範囲内にあることを規定し、

前記第 1 の規則は、前記第 1 のコーディングモードが適用される場合、前記コンテキストインクリメントの前記値が $N_2 \sim N_3$ の第 2 の範囲内にあることを規定する、

請求項 1 に記載の方法。

【請求項 3】

前記第 1 のコーディングモードにおいて、パルスコーディング変調表現を用いて、量子化残差と前記量子化残差の予測との差を前記ビットストリームにおいて表現する、

請求項 2 に記載の方法。

【請求項 4】

前記第 1 のコーディングモードは、ブロックベースのデルタパルスコード変調コーディングモードである、

請求項 2 または 3 に記載の方法。

【請求項 5】

N_0 、 N_1 、 N_2 、 N_3 は、それぞれ 0、2、3、5 である、

請求項 2 から 4 のうちいずれか一項に記載の方法。

【請求項 6】

第 1 の隣接符号値および第 2 の隣接符号値が第 1 の条件を満たすことに応じて、前記コンテキストインクリメントの前記値は 0 または 3 に等しく、

前記第 1 の条件は、

(1) 前記第 1 の隣接符号値が 0 に等しく、前記第 2 の隣接符号値が 0 に等しく、
あるいは、

(2) 前記第 1 の隣接符号値が前記第 2 の隣接符号値の負数に等しい、

請求項 1 から 5 のうちいずれか一項に記載の方法。

【請求項 7】

前記第 1 の隣接符号値は、現在の係数スキャン位置の水平座標が 0 であることに応じて 0 に等しく、前記現在の係数スキャン位置の水平座標が 0 でないことに応じて左隣接係数の値に等しく、

前記第 2 の隣接符号値は、前記現在の係数スキャン位置の垂直座標が 0 であることに応じて 0 に等しく、前記現在の係数スキャン位置の垂直座標が 0 でないことに応じて上隣接係数の値に等しい、

請求項 6 に記載の方法。

【請求項 8】

前記コンテキストインクリメントの前記値が、前記第 1 のコーディングモードが前記現在のブロックに適用されないことに応じて 0 に等しく、

前記コンテキストインクリメントの前記値が、前記第 1 のコーディングモードが前記現在のブロックに適用されることに応じて 3 に等しい、

請求項 6 に記載の方法。

【請求項 9】

前記コンテキストインクリメントの前記値が、前記第 1 の隣接符号値および前記第 2 の隣接符号値が前記第 1 の条件を満たさず、かつ、前記第 1 の隣接符号値および前記第 2 の隣接符号値が第 2 の条件を満たすことに応じて 1 または 4 に等しく、

前記第 2 の条件は、

10

20

30

40

50

(1) 前記第 1 の隣接符号値が 0 以上であり、前記第 2 の隣接符号値が 0 以上である、請求項 6 に記載の方法。

【請求項 1 0】

前記コンテキストインクリメントの前記値が、前記第 1 のコーディングモードが前記現在のブロックに適用されないことに応じて 1 に等しく、

前記コンテキストインクリメントの前記値が、前記第 1 のコーディングモードが前記現在のブロックに適用されることに応じて 4 に等しい、

請求項 9 に記載の方法。

【請求項 1 1】

前記コンテキストインクリメントの前記値が、前記第 1 の隣接符号値および前記第 2 の隣接符号値が前記第 1 の条件および前記第 2 の条件を満たさないことに応じて 2 または 5 に等しい、

請求項 9 に記載の方法。

【請求項 1 2】

前記コンテキストインクリメントの前記値が、前記第 1 のコーディングモードが前記現在のブロックに適用されないことに応じて 2 に等しく、

前記コンテキストインクリメントの前記値が、前記第 1 のコーディングモードが前記現在のブロックに適用されることに応じて 5 に等しい、

請求項 1 1 に記載の方法。

【請求項 1 3】

変換スキップ処理が前記現在のブロックに適用される、

請求項 1 から 1 2 のうちいずれか一項に記載の方法。

【請求項 1 4】

前記変換を実行することは、前記映像を前記ビットストリームに符号化することを含む、

請求項 1 から 1 3 のうちいずれか一項に記載の方法。

【請求項 1 5】

前記変換を実行することは、前記映像を前記ビットストリームから復号することを含む、

請求項 1 から 1 3 のうちいずれか一項に記載の方法。

【請求項 1 6】

プロセッサと、命令を有する非一時的メモリと、を備える映像データの処理装置であって、前記命令が前記プロセッサによって実行されると、前記プロセッサに、

映像の現在のブロックと前記映像のビットストリームとの間の変換を実行させ、

前記ビットストリームは、前記ビットストリームにおける第 1 の係数の第 1 の符号フラグを含むためにコンテキストインクリメントが使用されることを規定する第 1 の規則に準拠し、

前記第 1 の規則は、前記コンテキストインクリメントの値が前記現在のブロックに第 1 のコーディングモードが適用されるか否かに基づくことを規定し、

前記ビットストリームは、前記映像に複数の適応パラメータセットネットワーク抽象化レイヤユニットを使用することを規定する第 2 の規則に準拠し、

各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する適応パラメータセットタイプ値を有し、

各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する映像レイヤ識別子に関連付けられ、

各適応パラメータセットネットワーク抽象化レイヤユニットは、プレフィックスユニットまたはサフィックスユニットであり、

前記第 2 の規則は、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットが、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットの複数の映像レイヤ識別子とは独立して、および前記複数の適応パラメータセットネットワーク抽象化レイヤユニットが前記プレフィックスユニットまたは前記サフィックスユニットであるか否かとは独立して、特定の適応パラメータセットタイプ値を有することに応じて、前記複数

10

20

30

40

50

の適応パラメータセットネットワーク抽象化レイヤユニットの適応パラメータセット識別子値が同じ適応パラメータセット識別子値空間に属することを規定する、
装置。

【請求項 17】

命令を記憶する非一時的なコンピュータ可読記憶媒体であって、この命令は、プロセッサに、

映像の現在のブロックと前記映像のビットストリームとの間の変換を実行させ、

前記ビットストリームは、前記ビットストリームにおける第1の係数の第1の符号フラグを含むためにコンテキストインクリメントが使用されることを規定する第1の規則に準拠し、

前記第1の規則は、前記コンテキストインクリメントの値が前記現在のブロックに第1のコーディングモードが適用されるか否かに基づくことを規定し、

前記ビットストリームは、前記映像に複数の適応パラメータセットネットワーク抽象化レイヤユニットを使用することを規定する第2の規則に準拠し、

各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する適応パラメータセットタイプ値を有し、

各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する映像レイヤ識別子に関連付けられ、

各適応パラメータセットネットワーク抽象化レイヤユニットは、プレフィックスユニットまたはサフィックスユニットであり、

前記第2の規則は、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットが、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットの複数の映像レイヤ識別子とは独立して、および前記複数の適応パラメータセットネットワーク抽象化レイヤユニットが前記プレフィックスユニットまたは前記サフィックスユニットであるか否かとは独立して、特定の適応パラメータセットタイプ値を有することに応じて、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットの適応パラメータセット識別子値が同じ適応パラメータセット識別子値空間に属することを規定する、

非一時的なコンピュータ可読記憶媒体。

【請求項 18】

映像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、

現在のブロックを含む前記映像のビットストリームを生成し、

前記ビットストリームを非一時的なコンピュータ可読記憶媒体に記憶し、

前記ビットストリームは、前記ビットストリームにおける第1の係数の第1の符号フラグを含むためにコンテキストインクリメントが使用されることを規定する第1の規則に準拠し、

前記第1の規則は、前記コンテキストインクリメントの値が前記現在のブロックに第1のコーディングモードが適用されるか否かに基づくことを規定し、

前記ビットストリームは、前記映像に複数の適応パラメータセットネットワーク抽象化レイヤユニットを使用することを規定する第2の規則に準拠し、

各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する適応パラメータセットタイプ値を有し、

各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する映像レイヤ識別子に関連付けられ、

各適応パラメータセットネットワーク抽象化レイヤユニットは、プレフィックスユニットまたはサフィックスユニットであり、

前記第2の規則は、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットが、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットの複数の映像レイヤ識別子とは独立して、および前記複数の適応パラメータセットネットワーク抽象化レイヤユニットが前記プレフィックスユニットまたは前記サフィックスユニットであるか否かとは独立して、特定の適応パラメータセットタイプ値を有することに応じて、前記複数

10

20

30

40

50

の適応パラメータセットネットワーク抽象化レイヤユニットの適応パラメータセット識別子値が同じ適応パラメータセット識別子値空間に属することを規定する、

方法。

【発明の詳細な説明】

【技術分野】

【0001】

関連出願の相互参照

本願は、2020年4月2日出願の国際特許出願第PCT/CN2020/082983号の優先権および利益を主張する2021年4月1日出願の国際特許出願第PCT/CN2021/084869号に基づく。上記出願の開示全体は、本明細書の開示として参照によりここに援用される。

10

【0002】

この特許明細書は、画像および映像コーディングおよび復号に関する。

【背景技術】

【0003】

デジタル映像は、インターネットおよび他のデジタル通信ネットワークにおいて最大の帯域幅の使用量を占めている。映像を受信および表示することが可能である接続されたユーザ機器の数が増加するにつれ、デジタル映像の使用に対する帯域幅需要は増大し続けることが予測される。

【発明の概要】

20

【0004】

本願は、コーディングされた表現の復号に有用な制御情報を使用して、映像のコーディングされた表現を処理するために、映像エンコーダおよびデコーダにより使用され得る技術を開示する。

【0005】

1つの例示的な態様において、映像処理方法が開示される。この方法は、映像の現在のブロックとこの映像のビットストリームとの間の変換を行うことを含み、このビットストリームはこのビットストリームに第1の係数の第1の符号フラグを含めるためにコンテキストインデックスオフセットを使用することを規定する規則に準拠し、この規則は、このコンテキストインデックスオフセットの値が、このビットストリームにおけるこの現在のブロックに第1のコーディングモードが適用されるかどうかに基づくことを規定する。

30

【0006】

別の例示的な態様において、映像処理方法が開示される。この方法は、現在の映像のブロックとこの映像のビットストリームとの間の変換を行うことを含み、このビットストリームは、残りのコンテキストコーディングされたピンの数に基づいて、コンテキストモードおよびバイパスモードのうちのいずれか1つを使用して、この現在のブロックの符号フラグがこのビットストリームに含まれていることを規定する規則に準拠する。

【0007】

別の例示的な態様において、映像処理方法が開示される。この方法は、映像の現在のブロックとこの映像のビットストリームとの間の変換を行うことを含み、このビットストリームは、ビットストリームにこの現在のブロックの符号フラグを含めるためにコンテキストインデックスオフセットを使用することを規定する規則に準拠し、この規則は、このコンテキストインデックスオフセットをこの現在のブロックの情報に基づいて決定することを規定する。

40

【0008】

別の例示的な態様において、映像処理方法が開示される。この方法は、規則に従って、1つ以上の映像レイヤを含む映像とこの映像のビットストリームとの間の変換を行うことを含み、この規則は、この映像のために複数の適応パラメータセットネットワーク抽象化レイヤユニットを使用することを規定し、各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する適応パラメータタイプ値を有し、各適応パラメータセットネッ

50

トワーク抽象化レイヤユニットは、対応する映像レイヤ識別子に関連付けられ、各適応パラメータセットネットワーク抽象化レイヤユニットは、プレフィックスユニットまたはサフィックスユニットであり、この規則は、この複数の適応パラメータセットネットワーク抽象化レイヤユニットが同じ適応パラメータタイプ値を共有することに応じて、この複数の適応パラメータセットネットワーク抽象化レイヤユニットの適応パラメータセット識別子値が同じ識別子空間に属することを規定する。

【 0 0 0 9 】

別の例示的な態様において、映像処理方法が開示される。この方法は、映像の現在のブロックとこの映像のビットストリームとの間の変換を行うことを含み、このビットストリームは、特定の特徴を有する第1の補足拡張情報メッセージが、(1)この特定の特徴を有する第2の補足拡張情報メッセージがスライスユニットに含まれていることに応じて、このビットストリームのスライスユニット内で繰り返されること、または(2)この第1の補足拡張メッセージがこのビットストリームのこのスライスユニット内にあることに応じて更新されること、は許可されないことを規定する規則に準拠し、このスライスユニットは、復号順に連続したネットワーク抽象化レイヤユニットのセットを含み、このネットワーク抽象化レイヤのセットは、単一のコーディングされたスライスと、この単一のコーディングされたスライスに関連付けられた1つ以上の非映像コーディングレイヤネットワーク抽象化レイヤユニットとを含む。

10

【 0 0 1 0 】

別の例示的な態様において、映像処理方法が開示される。この方法は、映像の現在のブロックとこの映像のビットストリームとの間の変換を行うことを含み、このビットストリームは、このビットストリームにおける1つのスライスユニットが、(1)このスライスユニットは、第1の非映像コーディングレイヤネットワーク抽象化レイヤユニットと同じ特徴を有する第2の非映像コーディングレイヤネットワーク抽象化レイヤユニットを含むこと、および(2)この第1の非映像コーディングレイヤネットワーク抽象化レイヤユニットはプレフィックス補足拡張情報またはサフィックス補足拡張情報以外のネットワーク抽象化レイヤユニットタイプを有すること、に応じて、第1の非映像コーディングレイヤネットワーク抽象化レイヤユニットの繰り返しを許可しないことを規定する規則に準拠する。

20

【 0 0 1 1 】

別の例示的な態様において、映像処理方法が開示される。この方法は、規則に従って、複数の映像レイヤを含む映像とその映像のビットストリームとの間の変換を行うことを含み、その規則は、複数の適応パラメータセットのうちのどれがその複数の映像レイヤにわたって共有されることは許可されないかを規定する。

30

【 0 0 1 2 】

別の例示的な態様において、映像処理方法が開示される。この方法は、規則に従って、1つ以上の映像レイヤを含む映像とこの映像のビットストリームとの間の変換を行うことを含み、前記ビットストリームは1つ以上の適応ループフィルタ適応パラメータセットを含み、この規則は、この1つ以上の適応ループフィルタ適応パラメータセットを1つのピクチャユニット内で更新するように許可されるかどうかを規定する。

40

【 0 0 1 3 】

別の例示的な態様において、映像処理方法が開示される。この方法は、規則に従って、1つ以上のコーディングされたレイヤ映像シーケンスを含む映像とこの映像のビットストリームとの間の変換を行うことを含み、このビットストリームは適応ループフィルタ適応パラメータセットを含み、この規則は、この適応ループフィルタ適応パラメータセットが1つ以上の特定の特徴を有することに応じて、この適応ループフィルタ適応パラメータセットがこの1つ以上のコーディングされたレイヤ映像シーケンスにわたって共有されることは許可されないことを規定する。

【 0 0 1 4 】

別の例示的な態様において、映像処理方法が開示される。この方法は、映像の現在のブ

50

ロックと映像のコーディングされた表現との間で変換を行うことを含む。このコーディングされた表現は、このコーディングされた表現におけるこの現在のブロックの符号フラグをコーディングするために使用されるコンテキストインデックスオフセットがある範囲内にあることを規定するフォーマット規則に準拠し、この範囲はこのコーディングされた表現におけるこの現在のブロックを表現するために用いられるコーディングモードの範囲関数である。

【 0 0 1 5 】

別の例示的な態様において、別の映像処理方法が開示される。この方法は、映像の現在のブロックとこの映像のコーディングされた表現との間の変換を行うことを含み、このコーディングされた表現は、この現在のブロックの符号フラグが、コンテキストコーディングされたピンまたはバイパスモードのうちの1つを、残りのコンテキストコーディングされたピンの数に応じて使用してこのコーディングされた表現でコーディングされることを規定するフォーマット規則に準拠する。

10

【 0 0 1 6 】

別の例示的な態様において、別の映像処理方法が開示される。この方法は、映像の現在のブロックとこの映像のコーディングされた表現との間の変換を行うことを含み、この変換は、ブロックベースのデルタパルスコード変調 (B D P C M) モードを使用し、このコーディングされた表現は、符号フラグをコーディングするためのコンテキストインデックスオフセットがこの現在のブロックのコーディング条件の関数となるように、この B D P C M からの符号フラグをこのコーディングされた表現においてコンテキストコーディングすることを規定するフォーマット規則に準拠する。

20

【 0 0 1 7 】

別の例示的な態様において、別の映像処理方法が開示される。この方法は、映像の現在のブロックと映像のコーディングされた表現との間の変換を行うことを含み、このコーディングされた表現は、連続した復号順にあり、かつ単一のコーディングされたスライスを含むネットワーク抽象化レイヤユニットのセットに対応する、コーディングされた表現のスライスユニットが、最大でも1回、補足拡張情報 (S E I) の少なくとも部分を含むように許可されることを規定するフォーマット規則に準拠する。

【 0 0 1 8 】

別の例示的な態様において、別の映像処理方法が開示される。この方法は、映像の現在のブロックとこの映像のコーディングされた表現との間の変換を行うことを含み、このコーディングされた表現は、連続した復号順にあり、かつ単一のコーディングされたスライスを含むネットワーク抽象化レイヤユニットのセットに対応する、コーディングされた表現におけるスライスユニットが、1つ以上の映像コーディングレイヤネットワーク抽象化レイヤ (V C L N A L) ユニットを含むことを規定するフォーマット規則に準拠し、このフォーマット規則はさらに、このスライスユニットにおいて、繰り返されるように許可される第1のタイプのユニットと繰り返されるように許可されない第2のタイプのユニットとを規定する。

30

【 0 0 1 9 】

別の例示的な態様において、別の映像処理方法が開示される。この方法は、規則に基づいて、1つ以上の映像レイヤを含む映像とこの映像のコーディングされた表現との間の変換を行うことを含み、このコーディングされた表現は、1つ以上の適応パラメータセット (A P S) を含み、この規則は、1つ以上の映像レイヤの変換へのこの1つ以上の A P S のうちのいくつかの適用可能性を規定する。

40

【 0 0 2 0 】

別の例示的な態様において、別の映像処理方法が開示される。この方法は、規則に従って、1つ以上の映像レイヤを含む映像とこの映像のコーディングされた表現との間の変換を行うことを含み、このコーディングされた表現は、1つ以上のネットワーク抽象化レイヤ (N A L) ユニットに構成され、このコーディングされた表現は、この変換の特徴を制御するための1つ以上の適応パラメータセットを含む。

50

【 0 0 2 1 】

さらに別の例示的な態様において、映像エンコーダ装置が開示される。この映像エンコーダは、上述した方法を実装するように構成された処理装置を備える。

【 0 0 2 2 】

さらに別の例示的な態様において、映像デコーダ装置が開示される。この映像デコーダは、上述した方法を実装するように構成された処理装置を備える。

【 0 0 2 3 】

さらに別の例示的な態様では、コードが記憶されたコンピュータ可読媒体が開示される。このコードは、本明細書に記載の方法の1つを処理装置が実行可能なコードの形式で実施する。

10

【 0 0 2 4 】

これらのおよび他の特徴は、本明細書全体にわたって説明される。

【 図面の簡単な説明 】

【 0 0 2 5 】

【 図 1 】 映像処理システム例を示すブロック図である。

【 図 2 】 映像処理装置のブロック図である。

【 図 3 】 映像処理方法の一例を示すフローチャートである。

【 図 4 】 本発明のいくつかの実施形態による映像コーディングシステムを示すブロック図である。

【 図 5 】 本発明のいくつかの実施形態によるエンコーダを示すブロック図である。

20

【 図 6 】 本発明のいくつかの実施形態によるデコーダを示すブロック図である。

【 図 7 】 適応ループフィルタ (A L F) の形状の例を示す (彩度 : 5 × 5 菱形、輝度 : 7 × 7 菱形) 。

【 図 8 】 A L F および C C - A L F 図の例を示す。

【 図 9 】 映像処理の方法の例を示すフローチャートである。

【 図 1 0 】 映像処理の方法の例を示すフローチャートである。

【 図 1 1 】 映像処理の方法の例を示すフローチャートである。

【 図 1 2 】 映像処理の方法の例を示すフローチャートである。

【 図 1 3 】 映像処理の方法の例を示すフローチャートである。

【 図 1 4 】 映像処理の方法の例を示すフローチャートである。

30

【 図 1 5 】 映像処理の方法の例を示すフローチャートである。

【 図 1 6 】 映像処理の方法の例を示すフローチャートである。

【 図 1 7 】 映像処理の方法の例を示すフローチャートである。

【 発明を実施するための形態 】

【 0 0 2 6 】

本明細書では、理解を容易にするために章の見出しを使用しており、その技術および各章に記載された実施形態の適用可能性をその章のみに限定するものではない。さらに、H . 2 6 6 という用語は、ある説明において、理解を容易にするためだけに用いられ、開示される技術の範囲を限定するために用いられたものではない。このように、本明細書で説明される技術は、他の映像コーデックプロトコルおよび設計にも適用可能である。

40

【 0 0 2 7 】

1 導入

本明細書は、映像コーディング技術に関する。具体的には、これは、変換スキップモードにおける係数コーディング、並びにビデオコーディングにおける非VCLデータユニットの繰り返しおよび更新に関する。HEVCのような既存の映像コーディング規格に適用してもよいし、規格 (V e r s a t i l e V i d e o C o d i n g) を確定させるために適用してもよい。本発明は、将来の映像コーディング規格または映像コーデックにも適用可能である。

【 0 0 2 8 】

2 略語

50

ALF Adaptive Loop Filter (適応ループフィルタ)	
APS Adaptation Parameter Set (適応パラメータセット)	
AU Access Unit (アクセスユニット)	
AUD Access Unit Delimiter (アクセスユニットデリミター)	
AVC Advanced Video Coding (高度映像コーディング)	
CLVS Coded Layer Video Sequence (コーディングレイヤ映像シーケンス)	
CPB Coded Picture Buffer (コーディングピクチャバッファ)	
CRA Clean Random Access (クリーンランダムアクセス)	
CTU Coding Tree Unit (コーディングツリーユニット)	10
CVS Coded Video Sequence (コーディング映像シーケンス)	
DCI Decoding Capability Information (復号能力情報)	
DPB Decoded Picture Buffer (復号ピクチャバッファ)	
DU Decoding Unit (復号ユニット)	
EOB End Of Bitstream (ビットストリーム終端)	
EOS End Of Sequence (シーケンス終端)	
GDR Gradual Decoding Refresh (漸進的復号リフレッシュ)	
HEVC High Efficiency Video Coding (高効率映像コーディング)	20
HRD Hypothetical Reference Decoder (仮想参照デコーダ)	
IDR Instantaneous Decoding Refresh (瞬時復号更新)	
JEM Joint Exploration Model (共同探索モデル)	
LMCS Luma Mapping with Chroma Scaling (彩度スケールリングを伴う輝度マッピング)	
MCTS Motion-Constrained Tile Sets (動作制約タイルセット)	
NAL Network Abstraction Layer (ネットワーク抽象化レイヤ)	30
OLS Output Layer Set (出力層セット)	
PH Picture Header (ピクチャヘッダ)	
PPS Picture Parameter Set (ピクチャパラメータセット)	
PTL Profile, Tier and Level プロファイル、層およびレベル	
PU Picture Unit (ピクチャユニット)	
RADL Random Access Decodable Leading (ランダムアクセス復号可能リーディング) (ピクチャ)	
RAP Random Access Point (ランダムアクセスポイント)	
RASL Random Access Skipped Leading (ランダムアクセススキップリーディング) (ピクチャ)	40
RBSP Raw Byte Sequence Payload (生バイトシーケンスペイロード)	
RPL Reference Picture List (参照ピクチャリスト)	
SAO Sample Adaptive Offset (サンプル適応オフセット)	
SEI Supplemental Enhancement Information (補足拡張情報)	
SPS Sequence Parameter Set (シーケンスパラメータセット)	
STSA Step-wise Temporal Sublayer Access (段階的時間的サブレイヤアクセス)	
SVC Scalable Video Coding (スケーラブル映像コーディング)	50

VCL Video Coding Layer (映像コーディングレイヤ)
 VPS Video Parameter Set (映像パラメータセット)
 VTM VVC Test Model (VVC試験モデル)
 VUI Video Usability Information (映像ユーザビリティ
 情報)
 VVC Versatile Video Coding (汎用映像コーディング)

【0029】

3 映像コーディングの導入

映像コーディングする規格は、主に周知のITU-TおよびISO/IEC規格の開発
 によって発展してきた。ITU-TはH.261とH.263を作り、ISO/IECは
 MPEG-1とMPEG-4 Visualを作り、両団体はH.262/MPEG-2 V
 ideoとH.264/MPEG-4 AVC (Advanced Video Codin
 g)とH.265/HEVC規格を共同で作った。H.262以来、映像コーディング規
 格は、時間的予測と変換コーディングが利用されるハイブリッド映像コーディング構造に
 基づく。HEVCを超えた将来の映像コーディング技術を探索するため、2015年には
 、VCEGとMPEGが共同でJVET (Joint Video Exploratio
 n Team) を設立した。それ以来、多くの新しい方法がJVETによって採用され、
 JEM (Joint Exploration Model) と呼ばれる参照ソフトウェア
 に組み込まれてきた。JVETは四半期に1回開催され、新しいコーディング規格はHE
 VCに比べて50%のビットレート低減を目指している。2018年4月のJVET会議
 において、新しい映像コーディング規格を「VVC (Versatile Video C
 oding)」と正式に命名し、その時、第1版のVVCテストモデル (VTM) をリリ
 ースした。VVCの標準化に寄与する努力が続けられているので、すべてのJVET会議
 において、VVC標準に新しいコーディング技術が採用されている。毎回の会議の後、V
 VC作業草案およびテストモデルVTMを更新する。VVCプロジェクトは、現在、20
 20年7月の会合における技術完成 (FDIS) を目指している。

【0030】

10

20

30

40

50

【化 1】

3. 1 本開示に最も関連するVVCにおける概念および定義

アクセスユニット (AU) : 異なるレイヤに属し、かつDPBからの出力のために同時に関連付けられたコーディングされたピクチャを含むPUのセット。

適応ループフィルタ (ALF) : 復号処理の一部として適用され、APSにおいて伝達されるパラメータによって制御されるフィルタリング処理。

ALF APS : ALF処理を制御するAPS。

適応パラメータセット (APS) : スライスヘッダに見出される構文要素が0以上であることにより決定される、スライスが0以上である場合に適用される構文要素を含む構文構造。

関連付けられた非VCL NALユニット : VCL NALユニットが非VCL NALユニットの関連付けられたVCL NALユニットである場合、VCL NALユニットのための非VCL NALユニット (存在する場合)。

10

関連付けられたVCL NALユニット : nal_unit_typeがEOS_NUT、EOB_NUT、SUFFIX_APS_NUT、SUFFIX_SEI_NUT、FD_NUT、RSV_NVCL_27と等しい、またはUNSPEC_30.. UNSPEC_31の範囲内にある、非VCL NALユニットに対して復号順序で先行するVCL NALユニット、そうでない場合、復号順で次のVCL NALユニット。

復号ユニット (DU) : DecodingUnitHrdFlagが0に等しい場合はAU、そうでない場合、AUのサブセットで、AU内の1つ以上のVCL NALユニットと関連付けられた非VCL NALユニットで構成される。

レイヤ : すべてnuh_layer_idという特定の値を持ち、関連付けられた非VCL NALユニットを持つVCL NALユニットのセット。

20

LMCS APS : LMCS処理を制御するAPS

彩度スケーリングを伴う輝度マッピング (LMCS) : 復号処理の一部として適用され、輝度サンプルを特定の値にマッピングする処理であって、彩度サンプルの値にスケーリング演算を適用してもよい。

ピクチャユニット (PU) : 規定された分類規則に従って互いに関連付けられた1つのNALユニットのセットは、復号順に連続しており、正確に1つのコーディングされたピクチャを含む。

スケーリングリスト : 周波数インデックスごとにスケーリング処理の倍率を関連付けたリスト

スケーリングリストAPS : スケーリングリストを構築するために使用される構文要素を持つAPS。

映像コーディングレイヤ (VCL) NALユニット : コーディングされたスライスNALユニットの総称であり、NALユニットのサブセットは、nal_unit_typeの確保済み値を有し、本明細書ではこれはVCL_NALユニットに分類される。

30

【0031】

3. 2 変換スキップモードにおける係数コーディング

現在のVVC草案において、残差コーディングを変換スキップレベルの統計および信号特性に適應させるために、非TS係数コーディングに比べて、TS (変換スキップ) モードにおける係数コーディングについていくつかの修正が提案されている。

【0032】

J V E T - Q 2 0 0 1 - v E における本パートに関連する最新のテキストは、以下のとおりである。

7. 3. 1 0. 1 1 残差コーディング構文

40

【表 1】

residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {	記述子
if(sps_mts_enabled_flag && cu_sbt_flag && cIdx==0 && log2TbWidth==5 && log2TbHeight<6)	
log2ZoTbWidth = 4	
else	
log2ZoTbWidth = Min(log2TbWidth, 5)	
if(sps_mts_enabled_flag && cu_sbt_flag && cIdx==0 && log2TbWidth<6 && log2TbHeight==5)	
log2ZoTbHeight = 4	
else	
log2ZoTbHeight = Min(log2TbHeight, 5)	
if(log2TbWidth > 0)	
last_sig_coeff_x_prefix	ae(v)
if(log2TbHeight > 0)	
last_sig_coeff_y_prefix	ae(v)
if(last_sig_coeff_x_prefix > 3)	
last_sig_coeff_x_suffix	ae(v)
if(last_sig_coeff_y_prefix > 3)	
last_sig_coeff_y_suffix	ae(v)
log2TbWidth = log2ZoTbWidth	

10

20

【表 2】

log2TbHeight = log2ZoTbHeight	
remBinsPass1 = ((1 << (log2TbWidth + log2TbHeight)) * 7) >> 2	
log2SbW = (Min(log2TbWidth, log2TbHeight) < 2 ? 1 : 2)	
log2SbH = log2SbW	
if(log2TbWidth + log2TbHeight > 3)	
if(log2TbWidth < 2) {	
log2SbW = log2TbWidth	
log2SbH = 4 - log2SbW	
} else if(log2TbHeight < 2) {	
log2SbH = log2TbHeight	
log2SbW = 4 - log2SbH	
}	
numSbCoeff = 1 << (log2SbW + log2SbH)	
lastScanPos = numSbCoeff	
lastSubBlock = (1 << (log2TbWidth + log2TbHeight - (log2SbW + log2SbH))) - 1	
do {	
if(lastScanPos == 0) {	
lastScanPos = numSbCoeff	
lastSubBlock--	
}	
lastScanPos--	

30

40

50

【表 3】

xS = DiagScanOrder[log2TbWidth-log2SbW][log2TbHeight-log2SbH][lastSubBlock][0]	
yS = DiagScanOrder[log2TbWidth-log2SbW][log2TbHeight-log2SbH][lastSubBlock][1]	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][lastScanPos][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][lastScanPos][1]	
} while((xC != LastSignificantCoeffX) (yC != LastSignificantCoeffY))	10
if(lastSubBlock == 0 && log2TbWidth >= 2 && log2TbHeight >= 2 && !transform_skip_flag[x0][y0][cIdx] && lastScanPos > 0)	
LfnstDcOnly = 0	
if((lastSubBlock > 0 && log2TbWidth >= 2 && log2TbHeight >= 2) (lastScanPos > 7 && (log2TbWidth == 2 log2TbWidth == 3) && log2TbWidth == log2TbHeight))	
LfnstZeroOutSigCoeffFlag = 0	
if((lastSubBlock > 0 lastScanPos > 0) && cIdx == 0)	
MtsDcOnly = 0	
QState = 0	
for(i = lastSubBlock; i >= 0; i--) {	20

【表 4】

startQStateSb = QState	
xS = DiagScanOrder[log2TbWidth-log2SbW][log2TbHeight-log2SbH][i][0]	
yS = DiagScanOrder[log2TbWidth-log2SbW][log2TbHeight-log2SbH][i][1]	
inferSbDcSigCoeffFlag = 0	
if(i < lastSubBlock && i > 0) {	
sb_coded_flag[xS][yS]	ae(v)
inferSbDcSigCoeffFlag = 1	
}	
if(sb_coded_flag[xS][yS] && (xS > 3 yS > 3) && cIdx == 0)	
MtsZeroOutSigCoeffFlag = 0	
firstSigScanPosSb = numSbCoeff	
lastSigScanPosSb = -1	
firstPosMode0 = (i == lastSubBlock ? lastScanPos : numSbCoeff-1)	
firstPosMode1 = firstPosMode0	40
for(n = firstPosMode0; n >= 0 && remBinsPass1 >= 4; n--) {	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]	

10

20

30

40

50

【表 5】

if(sb_coded_flag[xS][yS] && (n>0 !inferSbDcSigCoeffFlag) && (xC != LastSignificantCoeffX yC != Last SignificantCoeffY)) {	
sig_coeff_flag[xC][yC]	ae(v)
remBinsPass1-	
if(sig_coeff_flag[xC][yC])	
inferSbDcSigCoeffFlag = 0	
}	
if(sig_coeff_flag[xC][yC]) {	
abs_level_gtx_flag[n][0]	ae(v)
remBinsPass1-	
if(abs_level_gtx_flag[n][0]) {	
par_level_flag[n]	ae(v)
remBinsPass1-	
abs_level_gtx_flag[n][1]	ae(v)
remBinsPass1-	
}	
if(lastSigScanPosSb == -1)	
lastSigScanPosSb = n	
firstSigScanPosSb = n	
}	

10

20

【表 6】

AbsLevelPass1[xC][yC] = sig_coeff_flag[xC][yC] + par_level_flag[n] + abs_level_gtx_flag[n][0] + 2*abs_level_gtx_flag[n][1]	
if(ph_dep_quant_enabled_flag)	
QState = QStateTransTable[QState][AbsLevelPass1[xC][yC] & 1]	
firstPosModel = n-1	
}	
for(n = firstPosMode0; n > firstPosModel; n-) {	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]	
if(abs_level_gtx_flag[n][1])	
abs_remainder[n]	ae(v)
AbsLevel[xC][yC] = AbsLevelPass1[xC][yC] +2*abs_remainder[n]	
}	
for(n = firstPosModel; n >= 0; n-) {	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]	
if(sb_coded_flag[xS][yS])	

30

40

50

【表 7】

dec_abs_level[n]	ae(v)
if(AbsLevel[xC][yC] > 0) {	
if(lastSigScanPosSb == -1)	
lastSigScanPosSb = n	
firstSigScanPosSb = n	
}	
if(ph_dep_quant_enabled_flag)	
QState =	
QStateTransTable[QState][AbsLevel[xC][yC]&1]	
}	
if(ph_dep_quant_enabled_flag	
!pic_sign_data_hiding_enabled_flag)	
signHidden = 0	
else	
signHidden = (lastSigScanPosSb - firstSigScanPosSb > 3 ? 1 : 0)	
for(n = numSbCoeff-1; n >= 0; n--) {	
xC = (xS << log2SbW) +	
DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) +	
DiagScanOrder[log2SbW][log2SbH][n][1]	
if((AbsLevel[xC][yC] > 0) &&	
(!signHidden (n != firstSigScanPosSb)))	
coeff_sign_flag[n]	ae(v)
}	

10

20

【表 8】

if(ph_dep_quant_enabled_flag) {	
QState = startQStateSb	
for(n = numSbCoeff-1; n >= 0; n--) {	
xC = (xS << log2SbW) +	
DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) +	
DiagScanOrder[log2SbW][log2SbH][n][1]	
if(AbsLevel[xC][yC] > 0)	
TransCoeffLevel[x0][y0][cIdx][xC][yC] =	
(2 * AbsLevel[xC][yC] - (QState > 1 ? 1 : 0)) *	
(1 - 2 * coeff_sign_flag[n])	
QState =	
QStateTransTable[QState][AbsLevel[xC][yC]&1]	
} else {	
sumAbsLevel = 0	
for(n = numSbCoeff-1; n >= 0; n--) {	
xC = (xS << log2SbW) +	
DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) +	
DiagScanOrder[log2SbW][log2SbH][n][1]	
if(AbsLevel[xC][yC] > 0) {	
TransCoeffLevel[x0][y0][cIdx][xC][yC] =	
AbsLevel[xC][yC] * (1 - 2 *	
coeff_sign_flag[n])	
if(signHidden) {	

30

40

50

【表 9】

sumAbsLevel += AbsLevel[xC][yC]	
if((n == firstSigScanPosSb) && (sumAbsLevel%2) == 1)	
TransCoeffLevel[x0][y0][cIdx][xC][yC] =	
-TransCoeffLevel[x0][y0][cIdx][xC][yC]	
}	
}	
}	
}	
}	
}	
}	
}	

10

【表 10】

residual_ts_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {	記述子
log2SbW=(Min(log2TbWidth, log2TbHeight) < 2 ? 1 : 2)	
log2SbH = log2SbW	
if(log2TbWidth + log2TbHeight > 3)	
if(log2TbWidth < 2) {	
log2SbW = log2TbWidth	
log2SbH = 4 - log2SbW	
} else if(log2TbHeight < 2) {	
log2SbH = log2TbHeight	
log2SbW = 4 - log2SbH	

20

【表 11】

}	
numSbCoeff = 1 << (log2SbW + log2SbH)	
lastSubBlock =	
(1 << (log2TbWidth + log2TbHeight - (log2SbW + log2SbH))) - 1	
inferSbCbf = 1	
RemCcbs = ((1 << (log2TbWidth + log2TbHeight)) * 7) >> 2	
for(i = 0; i <= lastSubBlock; i++) {	
xS =	
DiagScanOrder[log2TbWidth-log2SbW][log2TbHeight-log2SbH][i][0]	
yS =	
DiagScanOrder[log2TbWidth-log2SbW][log2TbHeight-log2SbH][i][1]	
if(i != lastSubBlock !inferSbCbf)	
sb_coded_flag[xS][yS]	ae(v)
if(sb_coded_flag[xS][yS] && i < lastSubBlock)	
inferSbCbf = 0	
/* First scan pass */	
inferSbSigCoeffFlag = 1	
lastScanPosPass1 = -1	
for(n = 0; n <= numSbCoeff-1 && RemCcbs >= 4; n++) {	
xC = (xS << log2SbW) +	
DiagScanOrder[log2SbW][log2SbH][n][0]	

30

40

50

【表 1 2】

yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]	
if(sb_coded_flag[xS][yS] && (n != numSbCoeff-1 !inferSbSigCoeffFlag)) {	
sig_coeff_flag[xC][yC]	ae(v)
RemCcbs--	
if(sig_coeff_flag[xC][yC])	
inferSbSigCoeffFlag = 0	
}	
CoeffSignLevel[xC][yC] = 0	
if(sig_coeff_flag[xC][yC] {	
coeff_sign_flag[n]	ae(v)
RemCcbs--	
CoeffSignLevel[xC][yC] = (coeff_sign_flag[n] > 0 ? -1 : 1)	
abs_level_gtx_flag[n][0]	ae(v)
RemCcbs--	
if(abs_level_gtx_flag[n][0]) {	
par_level_flag[n]	ae(v)
RemCcbs--	
}	
}	
AbsLevelPass1[xC][yC] = sig_coeff_flag[xC][yC] + par_level_flag[n] + abs_level_gtx_flag[n][0]	

10

20

【表 1 3】

lastScanPosPass1 = n	
}	
/* Greater than X scan pass (numGtXFlags=5) */	
lastScanPosPass2=-1	
for(n = 0; n <= numSbCoeff-1 && RemCcbs >= 4; n++) {	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]	
AbsLevelPass2[xC][yC] = AbsLevelPass1[xC][yC]	
for(j = 1; j < 5; j++) {	
if(abs_level_gtx_flag[n][j-1]) {	
abs_level_gtx_flag[n][j]	ae(v)
RemCcbs--	
}	
AbsLevelPass2[xC][yC]+=2*abs_level_gtx_flag[n][j]	
}	
lastScanPosPass2 = n	
}	
/* remainder scan pass */	
for(n = 0; n <= numSbCoeff-1; n++) {	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]	

30

40

50

【表 1 4】

<code>yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]</code>	
<code>if((n <= lastScanPosPass2 && AbsLevelPass2[xC][yC] >= 10) (n > lastScanPosPass2 && n <= lastScanPosPass1 && AbsLevelPass1[xC][yC] >= 2) (n > lastScanPosPass1 && sb_coded_flag[xS][yS]))</code>	
<code>abs_remainder[n]</code>	ae(v)
<code>if(n <= lastScanPosPass2)</code>	
<code> AbsLevel[xC][yC] = AbsLevelPass2[xC][yC]+2*abs_remainder[n]</code>	10
<code> else if(n <= lastScanPosPass1)</code>	
<code> AbsLevel[xC][yC] = AbsLevelPass1[xC][yC]+2*abs_remainder[n]</code>	
<code> else { /* bypass */</code>	
<code> AbsLevel[xC][yC] = abs_remainder[n]</code>	
<code> if(abs_remainder[n])</code>	
<code> coeff_sign_flag[n]</code>	ae(v)
<code> }</code>	
<code>if(BdpcmFlag[x0][y0][cIdx] == 0 && n <= lastScanPosPass1) {</code>	
<code> absLeftCoeff = xC > 0 ? AbsLevel[xC-1][yC] : 0</code>	20
<code> absAboveCoeff = yC > 0 ? AbsLevel[xC][yC-1] : 0</code>	
<code> predCoeff = Max(absLeftCoeff, absAboveCoeff)</code>	
<code> if(AbsLevel[xC][yC] == 1 && predCoeff > 0)</code>	

【表 1 5】

<code> AbsLevel[xC][yC] = predCoeff</code>	
<code> else if(AbsLevel[xC][yC] > 0 && AbsLevel[xC][yC] <= predCoeff)</code>	
<code> AbsLevel[xC][yC]—</code>	
<code> }</code>	30
<code>}</code>	
<code> TransCoeffLevel[x0][y0][cIdx][xC][yC] = (1 - 2 * coeff_sign_flag[n]) * AbsLevel[xC][yC]</code>	
<code>}</code>	
<code>}</code>	

【 0 0 3 3 】

3.2.1 符号フラグ `coeff_sign_flag` のコンテキストモデリングおよびコンテキストインデックスオフセット導出

表 5 1 - 初期化プロセスにおける各初期化タイプの `ctxIdx` と構文要素の関連付け

【表 1 6】

構文構造	構文要素	ctxTable	initType		
			0	1	2
<code>residual_coding()</code>	<code>last_sig_coeff_x_prefix</code>	表 119	0..22	23..45	46..68
	<code>last_sig_coeff_y_prefix</code>	表 120	0..22	23..45	46..68
	<code>sb_coded_flag[][]</code>	表 121	0..7	8..15	16..23
	<code>sig_coeff_flag[][]</code>	表 122	0..62	63..125	126..188
	<code>par_level_flag[]</code>	表 123	0..32	33..65	66..98
	<code>abs_level_gtx_flag[][]</code>	表 124	0..73	74..147	148..220
	<code>coeff_sign_flag[]</code>	表 125	0..5	6..11	12..17

表 1 2 5 - c o e f f _ s i g n _ f l a g の c t x I n c の i n i t V a l u e およ
び s h i f t I d x の 仕 様

【表 1 7】

初期化 変数	ctxIdx of coeff_sign_flag																	
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
initValue	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP
shiftIdx	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

10

表 1 3 1 - コンテキストコーディングされたピンを有する構文要素への c t x I n c の
割り当て

【表 1 8】

構文要素	binIdx					
	0	1	2	3	4	>= 5
coeff_sign_flag[] transform_skip_flag[x0][y0][cIdx] == 0 RemCbs == 0 slice_ts_residual_co ding_disabled_flag	bypass	na	na	na	na	na
coeff_sign_flag[] transform_skip_flag[x0][y0][cIdx] == 1 && RemCbs >= 0 && !slice_ts_residua l_coding_disabled_fl ag	0..5 (clause 9.3.4.2.10)	na	na	na	na	na

20

【 0 0 3 4 】

9 . 3 . 4 . 2 . 1 0 変換スキップモードのための構文要素 c o e f f _ s i g n _ f
l a g の c t x I n c の 導 出 プ ロ セ ス

この処理への入力は、色成分インデックス c I d x 、現在のピクチャの左上サンプルに
対して現在の変換ブロックの左上サンプルを規定する輝度位置 (x 0 , y 0) 、現在の係
数スキャン位置 (x C , y C) である。

この処理の出力は変数 c t x I n c である。

変数 l e f t S i g n および a b o v e S i g n は、以下のように導出される。

$$l e f t S i g n = (x C = = 0) ? 0 : C o e f f S i g n L e v e l [x C - 1] [y C] \quad (1 5 9 5)$$

40

$$a b o v e S i g n = (y C = = 0) ? 0 : C o e f f S i g n L e v e l [x C] [y C - 1] \quad (1 5 9 6)$$

変数 c t x I n c は、以下のように導出される。

- l e f t S i g n が 0 に等しく、 a b o v e S i g n が 0 に等しい場合、または l e
f t S i g n が - a b o v e S i g n に等しい場合、以下が適用される。

$$c t x I n c = (B d p c m F l a g [x 0] [y 0] [c I d x] = = 0 ? 0 : 3) \quad (1 5 9 7)$$

- そうでない場合、 l e f t S i g n が 0 以上かつ a b o v e S i g n が 0 以上である
場合、以下が適用される。

$$c t x I n c = (B d p c m F l a g [x 0] [y 0] [c I d x] ? 1 : 4)$$

50

(1 5 9 8)

- そうでない場合、以下が適用される。

$$c t \times I n c = (B d p c m F l a g [x 0] [y 0] [c I d x] ? 2 : 5)$$

(1 5 9 9)

【 0 0 3 5 】

3.3 ブロックベースの量子化残差ドメイン DPCM (BDPCM)

J V E T - M 0 4 1 3 において、画面コンテンツを効率的にコーディングするために、量子化残差ブロック差動パルスコード変調 (BDPCM) が提案され、VVC 草案に採用されている。

【 0 0 3 6 】

QR-BDPCM で使用される予測方向は、垂直予測モードおよび水平予測モードであり得る。イントラ予測は、イントラ予測と同様に、予測方向 (水平または垂直予測) にサンプルコピーすることで、ブロック全体で予測する。残差を量子化し、量子化された残差とその予測子 (水平または垂直) 量子化値との間のデルタはコーディングされる。これは、以下のように説明することができる。サイズ M (行) \times N (列) のブロックについて、 $r_{i,j}, 0 \leq i \leq M-1, 0 \leq j \leq N-1$ を、上または左ブロックの境界サンプルからのフィルタリングされていないサンプルを使用して、水平方向 (予測ブロックに対して左隣の画素値を1ラインずつコピーする) または垂直方向 (予測ブロックにおける各ラインに上隣のラインをコピーする) にイントラ予測を行った後の予測残差とする。 $Q(r_{i,j}), 0 \leq i \leq M-1, 0 \leq j \leq N-1$ は、残差 $r_{i,j}$ の量子化バージョンを表し、この場合、残差は、元のブロックと予測ブロック値との間の差である。次に、ブロック DPCM が量子化された残差サンプルに適用され、その結果、要素 $\tilde{r}_{i,j}$ を有する修正された $M \times N$ 個の配列 \tilde{R} が得られる。垂直 BDPCM が信号通知されると、以下ようになる。

【 0 0 3 7 】

【数 1】

$$\tilde{r}_{i,j} = \begin{cases} Q(r_{i,j}), & i=0, 0 \leq j \leq (N-1) \\ Q(r_{i,j}) - Q(r_{(i-1),j}), & 1 \leq i \leq (M-1), 0 \leq j \leq (N-1) \end{cases}$$

【 0 0 3 8 】

水平予測の場合、類似した規則が適用され、残差量子化サンプルは、以下の式によって得られる。

【 0 0 3 9 】

【数 2】

$$\tilde{r}_{i,j} = \begin{cases} Q(r_{i,j}), & 0 \leq i \leq (M-1), j=0 \\ Q(r_{i,j}) - Q(r_{i,(j-1)}), & 0 \leq i \leq (M-1), 1 \leq j \leq (N-1) \end{cases}$$

【 0 0 4 0 】

残差量子化サンプル $\tilde{r}_{i,j}$ はデコーダに送られる。

【 0 0 4 1 】

デコーダ側では、上記の計算を逆にして、 $Q(r_{i,j}), 0 \leq i \leq M-1, 0 \leq j \leq N-1$ を生成する。垂直予測の場合、

【 0 0 4 2 】

【数 3】

$$Q(r_{i,j}) = \sum_{k=0}^i \tilde{r}_{k,j}, \quad 0 \leq i \leq (M-1), 0 \leq j \leq (N-1).$$

【 0 0 4 3 】

水平方向の場合、

【数 4】

$$Q(r_{i,j}) = \sum_{k=0}^j \tilde{r}_{i,k}, \quad 0 \leq i \leq (M-1), \quad 0 \leq j \leq (N-1).$$

【0044】

逆量子化された残差 $Q^{-1}(Q(r_{i,j}))$ をイントラブロック予測値に加算し、再構成されたサンプル値を生成する。

【0045】

このスキームの主な利点は、逆方向の DPCM を、係数の構文解析中に急いで行うことができ、係数の構文解析中に予測子を追加するだけで済むこと、または、構文解析後に行うことができることである。

【0046】

3.4 全般および VVC におけるスケーラブル映像コーディング (SVC)

スケーラブル映像コーディング (SVC、時には、映像コーディングにおけるスケーラビリティとも呼ばれる) は、ベースレイヤ (BL) (時には、参照レイヤ (RL) と呼ばれる) 映像コーディングと呼ばれることもあり、1つ以上のスケーラブル拡張レイヤ (EL) が使用される。SVC において、ベースレイヤは、基本品質レベルの映像データを担持することができる。1つ以上の拡張レイヤは、例えば、より高い空間的、時間的、および/または信号対雑音 (SNR) レベルをサポートするように、追加の映像データを担持することができる。拡張レイヤは、前の、符号化されたレイヤに対して定義されてもよい。例えば、下層が BL として機能し、上層が EL として機能することができる。中間レイヤは、EL または RL のいずれか、またはその両方として機能することができる。例えば、中間レイヤ (例えば、最下レイヤでも最上レイヤでもないレイヤ) は、中間レイヤの下層のレイヤ、例えば、ベースレイヤまたは任意の介在する拡張レイヤのための EL であってもよく、同時に、中間レイヤの上層の1つ以上の拡張レイヤのための RL としての役割を果たす。同様に、HEVC 規格のマルチビューまたは 3D 拡張では、複数のビューが存在してもよく、1つのビューの情報を利用して別のビューの情報をコーディング (例えば、符号化または復号) することができる (例えば、動き推定、動きベクトル予測および/または他の冗長性)。

【0047】

SVC において、エンコーダまたはデコーダで使用されるパラメータは、それらを利用することができるコーディングレベル (例えば、映像レベル、シーケンスレベル、ピクチャレベル、スライスレベル等) に基づいてパラメータセットにグループ分けされる。例えば、ビットストリームにおける異なるレイヤのコーディング映像シーケンスによって利用できるパラメータは、映像パラメータセット (VPS) に含まれてもよく、コーディング映像シーケンスにおける1つ以上のピクチャによって利用されるパラメータは、シーケンスパラメータセット (SPS) に含まれてもよい。同様に、1つのピクチャの1つ以上のスライスで利用されるパラメータは、ピクチャパラメータセット (PPS) に含まれてもよく、1つのスライスに固有の他のパラメータは、スライスヘッダに含まれてもよい。同様に、特定のレイヤが所与の時間にどのパラメータセット (複数可) を使用しているかの指示は、様々なコーディングレベルで提供されてもよい。

【0048】

VVC における参照ピクチャリサンプリング (RPR) のサポートにより、空間的スケーラビリティサポートに必要なアップサンプリングは RPR アップサンプリングフィルタを使用するだけでよいので、追加の信号処理レベルのコーディングツールを必要とせずに、複数のレイヤ、例えば、VVC における SD および HD 解像度の2つのレイヤを含むビットストリームをサポートするように設計することができる。それにもかかわらず、スケーラビリティサポートのためには、高レベルの構文変更 (スケーラビリティをサポートし

10

20

30

40

50

ない場合と比較して)が必要である。スケーラビリティサポートは、VVCバージョン1に規定されている。AVCおよびHEVCの拡張を含む、任意の以前の映像コーディング規格におけるスケーラビリティサポートとは異なり、VVCのスケーラビリティの設計は、単層デコーダの設計にできるだけ適したものにされてきた。多層ビットストリームの復号能力は、ビットストリームに1つのレイヤしかなかったかの如く規定される。例えば、DPBサイズのような復号能力は、復号されるビットストリームのレイヤの数に依存しないよう規定される。基本的に、単層ビットストリームのために設計されたデコーダは、多層ビットストリームを復号することができるようにするために、多くの変更を必要としない。AVCおよびHEVCの多層拡張の設計と比較して、HLSの様子は、ある程度の柔軟性を犠牲にして大幅に簡略化されてきた。例えば、IRAP AUは、CVSに存在する各レイヤの画像を含むことが必要である。

10

【0049】

VVCビットストリームは、1つまたは複数個の出力レイヤセット(OLS)で構成されてもよい。OLSは、1つまたは複数のレイヤが出力レイヤとして指定されるレイヤの集合である。出力レイヤは、復号された後に出力されるレイヤである。

【0050】

3.5 パラメータセット

AVC、HEVC、VVCはパラメータ集合を指定する。パラメータセットのタイプは、SPS、PPS、APS、VPS等である。SPS、PPSは、AVC、HEVC、VVCのすべてでサポートされている。VPSは、HEVCから導入されたものであり、HEVCおよびVVCの両方に含まれる。APSは、AVCまたはHEVCに含まれていなかったが、最近のVVC草案のテキストに含まれている。

20

【0051】

SPSは、シーケンスレベルのヘッダ情報を伝送するように設計され、PPSは、頻繁に変化しないピクチャレベルのヘッダ情報を担持するように設計された。SPSおよびPPSを用いると、シーケンスまたはピクチャごとに頻繁に変化する情報を繰り返す必要がないので、この情報の冗長な信号通知を回避することができる。さらに、SPSおよびPPSを使用することは、重要なヘッダ情報の帯域外伝送を有効化し、それにより、冗長な伝送の必要性を回避するだけでなく、誤り耐性を改善する。

【0052】

VPSは、マルチレイヤのビットストリームのすべてのレイヤに共通であるシーケンスレベルのヘッダ情報を伝えるために導入された。

30

【0053】

APSは、コーディングするためかなりのビットを必要とし、複数のピクチャによって共有され得る、このようなピクチャレベルまたはスライスレベルの情報を担持するために導入された。そして、シーケンスにおいて、非常に多くの異なる変形例が存在し得る。

【0054】

VVCにおいて、APSは、ALF、LMCS、およびスケーリングリストパラメータのパラメータを搬送するために用いられる。

【0055】

3.6 VVCにおけるNALユニットタイプとNALユニットヘッダ構文および意味論

最近のVVCテキスト(JVET-Q2001-vE/v15)において、NALユニットヘッダ構文および意味論は以下のとおりである。

40

7.3.1.2 NALユニットヘッダ構文

50

【表 19】

nal_unit_header() {	記述子
forbidden_zero_bit	f(1)
nuh_reserved_zero_bit	u(1)
nuh_layer_id	u(6)
nal_unit_type	u(5)
nuh_temporal_id_plus1	u(3)
}	

7.4.2.2 NALユニットヘッダの意味論

10

【0056】

【化2】

`forbidden_zero_bit`は、0に等しいものとする。

`nuh_reserved_zero_bit`は、0に等しいものとする。`nuh_reserved_zero_bit`の値1は、将来、ITU-T | ISO/IECにより規定してもよい。デコーダは、`nuh_reserved_zero_bit`が1に等しいNALユニットを無視する（すなわち、ビットストリームから削除し、廃棄する）。

`nuh_layer_id`は、VCL NALユニットが属するレイヤの識別子、または非VCL NALユニットが適用されるレイヤの識別子を規定する。`nuh_layer_id`の値は、0から55までの範囲内にあるものとする。`nuh_layer_id`の他の値は、ITU-T | ISO/IECで将来使用されるよう、確保されている。

20

`nuh_layer_id`の値は、1つのコーディングされたピクチャのすべてのVCL NALユニットに対して同じであるものとする。コーディングされたピクチャまたはPUの`nuh_layer_id`の値は、コーディングされたピクチャまたはPUのVCL NALユニットの`nuh_layer_id`の値である。

AUD、PH、EOS、FD NALユニットの`nuh_layer_id`の値は、以下のように制約される。

- `nal_unit_type`がAUD_NUTに等しい場合、`nuh_layer_id`は`vps_layer_id[0]`に等しいものとする。
- あるいは、`nal_unit_type`がPH_NUT、EOS_NUT、FD_NUTに等しい場合、`nuh_layer_id`は関連付けられたVCL NALユニットの`nuh_layer_id`に等しいものとする。

30

注1 - DCI、VPS、およびEOB NALユニットの`nuh_layer_id`の値は制約されていない。

【化3】

`nal_unit_type`の値は、CVSS AUのすべてのピクチャについて同じものとする。

`nal_unit_type`は、表5で規定されているように、NALユニットタイプ、すなわちNALユニットに含まれるRbspデータ構造のタイプを規定する。

40

UNSPEC_28...UNSPEC_31の範囲内にあり、意味論が規定されていない`nal_unit_type`を有するNALユニットは、本明細書で規定される復号処理に影響を及ぼさないものとする。

注2 - UNSPEC_28...UNSPEC_31の範囲内にあるNALユニットタイプ、アプリケーションにより決定されたとおりに使用されてもよい。本明細書では、`nal_unit_type`のこれらの値の復号処理は規定されていない。異なるアプリケーションはこれらのNALユニットタイプを異なる目的で使用してもよいので、これらの`nal_unit_type`値を有するNALユニットを生成するエンコーダの設計、およびこれらの`nal_unit_type`値を有するNALユニットのコンテンツを解釈するデコーダの設計にあたり、特に注意しなければならない。本明細書は、これらの値

50

の管理を定義していない。これらの `nal_unit_type` 値は、使用の「衝突」（すなわち、同じ `nal_unit_type` 値に対するNALユニットのコンテンツの意味の異なる定義）が重要でない、または可能でない、または管理された状況、例えば、制御アプリケーションまたはトランスポート仕様において、またはビットストリームが分散される環境を制御することによって定義または管理されるコンテキストでの使用にのみ適している場合がある。

（附属書Cに規定されるように）ビットストリームのDUにおけるデータの数を決定すること以外の目的のために、デコーダは、`nal_unit_type`の予約済みの値を使用するすべてのNALユニットのコンテンツを無視する（ビットストリームから取り除き、廃棄する）ものとする。

注3 - この要件は、本明細書に適合する拡張を将来的に定義することを可能にする。

表5 - NALユニットタイプコードおよびNALユニットタイプクラス

【0057】

【表20】

<code>nal_unit_type</code>	<code>nal_unit_type</code> の名称	NALユニットのコンテンツとRbsp構文構造	NALユニットタイプクラス
0	TRAIL_NUT	末尾のピクチャのコーディングされたスライス <code>slice_layer_rbsp()</code>	VCL
1	STSA_NUT	STSAピクチャのコーディングされたスライス <code>slice_layer_rbsp()</code>	VCL
2	RADL_NUT	RADLピクチャのコーディングされたスライス <code>slice_layer_rbsp()</code>	VCL
3	RASL_NUT	RASLピクチャのコーディングされたスライス <code>slice_layer_rbsp()</code>	VCL
4..6	RSV_VCL_4.. RSV_VCL_6	確保済み非IRAP VCL NALユニットタイプ	VCL
7	IDR_W_RADL	IDRピクチャのコーディングされたスライス <code>slice_layer_rbsp()</code>	VCL
8	IDR_N_LP		
9	CRA_NUT	CRAピクチャのコーディングされたスライス <code>silce_layer_rbsp()</code>	VCL
10	GDR_NUT	GDRピクチャのコーディングされたスライス <code>slice_layer_rbsp()</code>	VCL
11	RSV_IRAP_11	確保済みIRAP VCL NALユニットタイプ	VCL
12	RSV_IRAP_12		
13	DCI_NUT	復号能力情報 <code>decoding_capability_information_rbsp()</code>	non-VCL
14	VPS_NUT	映像パラメータセット <code>video_parameter_set_rbsp()</code>	non-VCL
15	SPS_NUT	シーケンスパラメータセット <code>seq_parameter_set_rbsp()</code>	non-VCL
16	PPS_NUT	ピクチャパラメータセット <code>pic_parameter_set_rbsp()</code>	non-VCL
17	PREFIX_APS_NUT	適応パラメータセット	non-VCL
18	SUFFIX_APS_NUT	<code>adaptation_parameter_set_rbsp()</code>	

10

20

30

40

50

【表 2 1】

	SUFFIX_APS_NUT		
19	PH_NUT	ピクチャヘッダ picture_header_rbsp()	non-VCL
20	AUD_NUT	AU区切り文字 access_unit_delimiter_rbsp()	non-VCL
21	EOS_NUT	シーケンス終端 end_of_seq_rbsp()	non-VCL
22	EOB_NUT	ビットストリーム終端 end_of_bitstream_rbsp()	non-VCL
23	PREFIX_SEI_NUT	補足拡張情報	non-VCL
24	SUFFIX_SEI_NUT	sei_rbsp()	
25	FD_NUT	フィラーデータ filler_data_rbsp()	non-VCL
26	RSV_NVCL_26	確保済み非VCL NALユニットタイプ	non-VCL
27	RSV_NVCL_27		
28..31	UNSPEC_28.. UNSPEC_31	規定されていない非VCL NALユニットタイプ	non-VCL

10

注 4 - CRA (Clean Random Access) ピクチャは、ビットストリームに存在する関連付けられた RASL または RADL ピクチャを有してもよい。

20

注 5 - IDR_N_LP に等しい nal_unit_type を有する IDR (Instantaneous Decoding Refresh) ピクチャは、ビットストリームに存在する関連付けられた先頭ピクチャを有さない。IDR_W_RADL に等しい nal_unit_type を有する IDR ピクチャは、ビットストリームに存在する関連付けられた RASL ピクチャ有していないが、ビットストリームに関連付けられた RADL ピクチャを有していてもよい。

nal_unit_type の値は、1 つのサブピクチャのすべての VCL NAL ユニットについて同じものとする。1 つのサブピクチャは、このサブピクチャの VCL NAL ユニットと同じ NAL ユニットタイプを有すると見なされている。

【0058】

30

任意の特定のピクチャの VCL NAL ユニットに対して、以下が適用される。

- mixed_nalu_types_in_pic_flag が 0 に等しい場合、nal_unit_type の値は、ピクチャの全ての VCL NAL ユニットについて同一であるものとし、ピクチャまたは PU は、このピクチャまたは PU のコーディングされたスライス NAL ユニットと同一の NAL ユニットタイプを有するとみなされる。

- そうでない場合 (mixed_nalu_types_in_pic_flag が 1 に等しい)、このピクチャは、少なくとも 2 つのサブピクチャを有し、このピクチャの VCL NAL ユニットは、以下のような正確に 2 つの異なる nal_unit_type 値を有するものとする。このピクチャの少なくとも 1 つのサブピクチャの VCL NAL ユニットは、全て STSA_NUT、RADL_NUT、RASL_NUT、IDR_W_RADL、IDR_N_LP、または CRA_NUT に等しい特定の nal_unit_type を有する一方、ピクチャ内の他のサブピクチャの VCL NAL ユニットは、全て、異なる特定値として TRAIL_NUT、RADL_NUT、または RASL_NUT に等しい nal_unit_type を有するものとする。

40

【0059】

単レイヤビットストリームの場合、以下の制約が適用される。

- 各ピクチャは、復号順においてビットストリームの第 1 のピクチャを除き、復号順において前の IRAP ピクチャに関連付けられていると考えられる。

- ピクチャが IRAP ピクチャの先端ピクチャである場合、RADL または RASL ピクチャとする。

50

- ピクチャが I R A P ピクチャの後端ピクチャである場合、R A D L または R A S L ピクチャではないものとする。

- R A S L ピクチャは、I D R ピクチャに関連付けられたビットストリームに含まれていないものとする。

- R A D L ピクチャは、I D R _ N _ L P に等しい `nal_unit_type` を有する I D R ピクチャに関連付けられたビットストリームに含まれていないものとする。

注 6 - 各パラメータセットが参照されるときに、(ビットストリームにおいて、または本明細書で規定されていない外部手段によって) 利用可能である限り、I R A P P U の前のすべての P U を破棄することにより、I R A P P U の位置でランダムアクセスを実行する(かつ、復号順において I R A P ピクチャおよびすべての後続の非 R A S L ピクチャを正しく復号する)ことが可能である。

- 復号順において、I R A P ピクチャに先行するピクチャは、出力順で I R A P ピクチャに先行し、出力順において、I R A P ピクチャに関連付けられた R A D L ピクチャに先行するものとする。

- C R A ピクチャに関連付けられた R A S L ピクチャは、出力順において C R A ピクチャに関連付けられた R A D L ピクチャに先行するものとする。

- C R A ピクチャに関連付けられた R A S L ピクチャは、復号順で C R A ピクチャに先行する I R A P ピクチャの出力順に従うものとする。

- `field_seq_flag` が 0 に等しく、現在のピクチャが I R A P ピクチャに関連付けられた先頭ピクチャに等しい場合、それは、復号順において、同じ I R A P ピクチャに関連付けられたすべての非先頭ピクチャに先行するものとする。そうでない場合、`picA` および `picB` を、それぞれ、I R A P ピクチャに関連付けられた、復号順において、最初のおよび最後の先頭ピクチャとすると、復号順において `picA` に先行する最大で 1 つの非先頭ピクチャが存在し、復号順において `picA` および `picB` の間に非先頭ピクチャはないものとする。

【0060】

【化4】

`nuh_temporal_id_plus1-1` は、NAL ユニットの時間的識別子を規定する。

`nuh_temporal_id_plus1` の値は 0 に等しくないものとする。

変数 `TemporalId` は、以下のように導出される。

$$\text{TemporalId} = \text{nuh_temporal_id_plus1} - 1 \quad (36)$$

【0061】

`nal_unit_type` が I D R _ W _ R A D L ~ R S V _ I R A P _ 1 2 の範囲内にある場合、`TemporalId` は 0 に等しいものとする。

`nal_unit_type` が S T S A _ N U T に等しく、`vps_independent_layer_flag[GeneralLayerIdx[nuh_layer_id]]` が 1 に等しい場合、`TemporalId` は 0 に等しくないものとする。

【0062】

`TemporalId` の値は、A U のすべての V C L N A L ユニットに対して同じであるものとする。コーディングされたピクチャ、P U、または A U の `TemporalId` の値は、コーディングされたピクチャ、P U、または A U の V C L N A L ユニットの `TemporalId` の値である。サブレイヤ表現の `TemporalId` の値は、サブレイヤ表現におけるすべての V C L N A L ユニットの `TemporalId` の最大値である。

【0063】

非 V C L N A L ユニットの `TemporalId` の値は、以下のように制約される。

- `nal_unit_type` が D C I _ N U T、V P S _ N U T、V P S _ N U T、または S P S _ N U T に等しい場合、`TemporalId` は 0 に等しく、N A L ユニ

10

20

30

40

50

ットを含むAUのTemporalIdは0に等しいものとする。

- そうでない場合、nal_unit_typeがPH_NUTに等しい場合、TemporalIdは、NALユニットを含むPUのTemporalIdであるものとする。

- そうでない場合、nal_unit_typeがEOS_NUTまたはEOB_NUTに等しい場合、TemporalIdは0に等しいものとする。

- そうでない場合、nal_unit_typeがAUD_NUT、FD_NUT、PREFIX_SEI_NUT、またはSUFFIX_SEI_NUTに等しい場合、TemporalIdはNALユニットを含むAUのTemporalIdであるものとする。

- そうでない場合、nal_unit_typeがPPS_NUT、PREFIX_APS_NUT、またはSUFFIX_APS_NUTに等しい場合、TemporalIdはNALユニットを含むPUのTemporalId以上であるものとする。

注7 - NALユニットが非VCL NALユニットである場合、TemporalIdの値は、非VCL NALユニットが適用されるすべてのAUのTemporalId値の最小値に等しい。nal_unit_typeがPPS_NUT、PREFIX_APS_NUT、またはSUFFIX_APS_NUTに等しい場合、TemporalIdは、AUを含むTemporalId以上であってもよく、すべてのPPSおよびAPSがビットストリームの始まりに含まれてもよく（例えば、それらが帯域外に輸送されている場合、受信機はそれらをビットストリームの先端に配置する）、第1のコーディングされたピクチャは、0に等しいTemporalIdを有する。

【0064】

3.7. 適応ループフィルタ (ALF)

2つの菱形フィルタ形状（図7に示す）がブロックベースのALFにおいて使用される。輝度成分に対し7×7の菱形が適用され、彩度成分には5×5の菱形が適用される。局所的な勾配の方向および活性度に基づいて、4×4ブロックごとに最大25個までのフィルタのうち1つを選択する。ピクチャにおける各4×4ブロックは、方向性および活性度に基づいて分類される。各4×4ブロックをフィルタリングする前に、そのブロックに対して計算された勾配値に基づいて、回転または対角および垂直フリップ等の簡単な幾何学的変換をフィルタ係数に適用することができる。これは、これらの変換をフィルタサポート領域内のサンプルに適用することに等しい。その考えは、ALFが適用される異なるブロックを、それらの方向性を揃えることによって、より類似させることである。ブロックベースの分類は彩度成分には適用されない。

【0065】

ALFフィルタパラメータは、適応パラメータセット (APS) において信号通知される。1つのAPSにおいて、最大25組の輝度フィルタ係数およびクリッピング値インデックス、並びに最大8組の彩度フィルタ係数およびクリッピング値インデックスを信号通知することができる。ビットオーバーヘッドを低減するために、輝度成分の異なる分類のフィルタ係数をマージすることができる。ピクチャまたはスライスヘッダにおいて、現在のピクチャまたはスライスに対して使用される輝度フィルタセットを特定するように、最大7個のAPSのIDを通知することができる。フィルタリング処理はCTBレベルでさらに制御される。1つの輝度CTBは、16個の固定フィルタセットと複数のAPSにおいて信号通知されるフィルタセットから、1つのフィルタセットを選択することができる。彩度成分の場合、現在のピクチャまたはスライスに使用されている彩度フィルタセットを示すように、ピクチャまたはスライスヘッダにAPS IDを信号通知する。CTBレベルにおいて、APSにおいて2以上の彩度フィルタセットが存在する場合、各彩度CTBに対しフィルタインデックスを信号通知する。CTBに対してALFが有効化されるとき、そのCTB内の各サンプルに対して、信号伝達された重みを有する菱形フィルタを実行し、クリッピング演算を適用して、隣接するサンプルと現在のサンプルとの間の差をクリップする。クリッピング演算は、現在のサンプル値とあまりにも異なる隣接サンプル値

の影響を低減することによって、ALFをより効率的にするように、非線形性を導入する。

【0066】

クロスコンポーネント適応ループフィルタ(CC-ALF)は、前述のALFの上部の各彩度成分をさらに拡張することができる。CC-ALFの目的は、輝度サンプル値を使用して各彩度成分を洗練することである。これは、菱形のハイパス線形フィルタを適用し、このフィルタリング演算の出力を彩度微調整に用いることによって実現される。図8は、他のループフィルタに対するCC-ALF処理のシステムレベルダイアグラムを提供する。図8に示すように、CC-ALFは、全ループフィルタ処理の追加の工程を避けるために、輝度ALFと同じ入力を使用する。

【0067】

3.8. 彩度スケールリングを伴う輝度マッピング(LMCS)

【化5】

他のインループフィルタ(すなわち、デブロッキング、SAO、およびALF)とは異なり、一般的には、空間的隣接のサンプルの情報を使用して現在のサンプルにフィルタリング処理を適用し、コーディングアーチファクトを低減する。これに対し、彩度スケールリング(LMCS)を用いた輝度マッピングは、コード名をダイナミックレンジ全体に再配布することで、符号化前の入力信号を修正して、圧縮効率を改善する。LMCSは、2つの主要なコンポーネント、即ち、(a)適応区分的線形モデルに基づく輝度成分のインループマッピング、および(b)彩度成分のための輝度依存彩度残差スケールリング、を有する。輝度マッピングは、順方向マッピング関数 $FwdMap$ と、対応する逆方向マッピング関数 $InvMap$ を利用する。 $FwdMap$ 関数は、16個の等しい区分を持つ区分的線形モデルを使用して信号通知される。 $InvMap$ 関数は信号通知する必要はなく、代わりに $FwdMap$ 関数から派生する。APSにおいて、輝度マッピングモデルを信号通知する。コーディング映像シーケンスにおいて、最大4つのLMCS APSを使用できる。1つのピクチャに対してLMCSが有効化されると、輝度マッピングパラメータを搬送するAPSを識別するように、ピクチャヘッダにAPS IDが信号通知される。スライスに対してLMCSが有効化されるとき、 $InvMap$ 関数をすべての再構成された輝度ブロックに適用し、サンプルを元のドメインに戻す。インターコーディングブロックの場合、追加のマッピング処理が必要であり、このマッピング処理は、 $FwdMap$ 関数を適用して、通常の補償処理の後、元のドメインの輝度予測ブロックをマッピングされたドメインにマッピングする。彩度残差スケールリングは、輝度信号とそれに対応する彩度信号間の相互作用を補正するように設計されている。輝度マッピングが有効化されるとき、輝度依存彩度残差スケールリングが有効化されるかどうかを示すために、追加のフラグが信号通知される。彩度残差スケールリング係数は、現在のCUの上側および/または左側の再構成された隣接の輝度サンプルの平均値に依存する。スケールリング係数が決定されると、符号化段階において、イントラ予測残差およびインター予測残差の両方に順方向スケールリングを適用し、再構成された残差に逆方向スケールリングを適用する。

【0068】

4 開示される技術的解決策が解決しようとする技術課題の例

最近のVVCテキスト(JVET-Q2001-vE/v15)における既存の設計は、以下の問題を有する。

1) JVET-N0280における係数コーディングは、画面内容のコーディングにおいてコーディングの利点を実現することができるが、係数コーディングおよびTSモードは、依然としていくつかの欠点を有する可能性がある。

a) 例えば、BDPCMがシーケンスに対して無効になっている場合、コーディングフラグをコーディングするために使用されるコンテキスト(3.2.1章に記載)は非連続であってもよい。このような設計では、非連続的なコンテキストインデックスに関連付けられたメモリアドレスも非連続的であるため、コンテキストの切り替えが困難になる。

b) コーディングフラグにバイパスコーディングを使用するか、コンテキストコーディングを使用するかは、このケースでは不明である。

- 残りの許可されたコンテキストコーディングされたピンの数($RemCcb_s$ で表される)は、0に等しい。

- 現在のブロックはTSモードでコーディングされる。

- `slice_ts_residual_coding_disabled_fl`

10

20

30

40

50

`ag` は `false` である。

2) `SEI` メッセージの大部分の繰り返しは、`PU` 内で最大 4 回に制限され、復号ユニット情報 `SEI` メッセージの繰り返しは、`DU` 内で最大 4 回に制限される。しかしながら、2 つの `VCL NAL` ユニット間で同じ `SEI` メッセージを複数回繰り返すことも可能であり、それは無意味である。

3) 2 つの `VCL NAL` ユニットの間に `SEI NAL` ユニット以外の非 `VCL NAL` ユニットの繰り返しは認められるが、これも意味がない。

4) `APS NAL` ユニットは複数のレイヤにわたって共有され得る。しかしながら、`ALF` の場合、フィルタは再構成されたサンプルに大きく依存し、異なるレイヤの場合は、同じ映像コンテンツであっても、`QP` が異なって使用される可能性があり、したがって、ほとんどの場合、異なるレイヤのピクチャに対して生成されたものからの `ALF` フィルタ継承は、無視し得るコーディングゲインベネフィットを生じない。したがって、`ALF APS` 動作をよりシンプルにするために、複数のレイヤにわたる `APS NAL` ユニットの共有は禁止されるべきである。他のタイプの `APS` (すなわち、`LMCS APS` および `スケーリングリスト APS`) についても同様である。

【化 6】

5) 以下の制約が存在する：

`adaptation__parameter__set__id` の特定の値を有し、かつ `PU` 内の `aps__params__type` の特定の値を有するすべての `APS NAL` ユニットは、それらがプレフィックスであるかサフィックス `APS NAL` ユニットであるかどうかに関わらず、同じコンテンツを有するものとする。

この制約は、`PU` 内の `APS NAL` ユニットの内容の更新を許可しない。しかし、ピクチャを複数のスライスに分割する場合、エンコーダが各スライスごとに `ALF` パラメータを個別に把握するのが可能にすることは理にかなっている。スライスの数が大きい場合、例えば、360 度映像応用において、`ALF APS NAL` ユニットに対するアップデート (即ち、`NAL` ユニットのタイプ、`APS ID`、および `APS` タイプフィールド以外の `APS NAL` ユニットの構文要素の値の変更) が必要となる場合がある。

6) 特定の値の `aps__params__type` を有する全ての `APS NAL` ユニットは、`nuh__layer__id` の値に関わらず、`adaptation__parameter__set__id` に対して同じ値空間を共有することが規定されている。ただし、その一方で、`PU` 内で `adaptation__parameter__set__id` の特定の値および `aps__params__type` の特定の値を有する全ての `APS NAL` ユニットは、これらがプレフィックス `APS NAL` ユニット、サフィックス `APS NAL` ユニットであるかに関わらず、同じ内容でなければならないことが規定されている。したがって、異なる `NAL` ユニットタイプを有する同じタイプの `APS` が、`adaptation__parameter__set__id` に対して同じ値空間を共有することも理にかなっている。

7) `APS NAL` ユニット (`nal__unit__type` の特定の値、`adaptation__parameter__set__id` の特定の値、および `aps__params__type` の特定の値を有する) は、`PU` 間および `CLVS` 間でも共有されることが認められている。しかし、特定の `APS NAL` ユニットの `CLVS` にわたって共有できるような良好なコーディングゲインを有することは期待されない。

【0069】

5 技術的解決策および実施例の一覧

上述した課題等を解決するために、以下に示す方法が開示されている。これらの項目は、一般的な概念を説明するための例であり、狭義に解釈されるべきではない。さらに、これらの項目は、個々に適用されてもよく、または任意の方法で組み合わせられてもよい。

10

20

30

40

50

【化7】

問題1に対処するための解決策:

$BdpcmFlag$ を現在のBDPCMフラグとし、 $leftSign$ を左隣りの係数の符号とし、および $aboveSign$ を上隣りの係数の符号とする。

条件Mを、 $leftSign$ が0に等しく、 $aboveSign$ が0に等しい、または $leftSign$ が $-aboveSign$ に等しいとする。

1. 符号フラグをコーディングするために許容されるコンテキストインデックスオフセットは、現在のブロックがモードXでコーディングされる場合、第1の範囲 $[N0, N1]$ 内にあり、それ以外の場合は $[N2, N3]$ であることが提案される。

a. 一例において、モードXは、輝度および/または彩度BDPCMモードであってもよい。

b. 一例において、 $N0, N1, N2, N3$ は、それぞれ0、2、3、5であることができる。

c. 一例において、上記例は、条件Mが偽である場合に適用されてもよい。

d. 一例として、上記の例は、ブロックに対して変換スキップ($transformation\ skip$)を適用する場合に適用されてもよい。

e. $N0, N1, N2, N3$ は、以下の1つ以上に基づいて決定されてもよい。

i. SPS/VPS/PPS/ピクチャヘッダ/スライスヘッダ/タイルグループヘッダ/LCU行/LCU群/LCU/CUにおいて信号通知された指示

ii. 現在のブロックおよび/またはその隣接のブロックのブロック寸法

iii. 現在のブロックおよび/またはその隣接のブロックのブロック形状

iv. カラーフォーマットの表示(例えば、4:2:0、4:4:4)

v. 別個または二重のコーディングツリー構造が使用されているかどうか

vi. スライスのタイプおよび/またはピクチャのタイプ

vii. 色成分の数

2. ($RemCcbS$ で表される)残りの許可されたコンテキストコーディングされたピンの数がNよりも小さい場合、符号フラグをバイパスモードでコーディングすることが提案される。

a. 一例において、符号フラグは、 $RemCcbS < N$ である場合、バイパスモードでコーディングされる。

i. あるいは、一例において、 $RemCcbS = N$ である場合、符号フラグはコンテキストモードでコーディングされる。

b. 一例において、 $RemCcbS$ がNに等しい場合、符号フラグはバイパスモードでコーディングされる。

i. あるいは、一例において、 $RemCcbS > N$ である場合、符号フラグはバイパスモードでコーディングされる。

c. 一例において、Nは4に等しく設定されてもよい。

i. あるいは、一例において、Nは0に等しく設定されてもよい。

d. 一例において、Nは整数であり、以下に基づいてもよい。

i. SPS/VPS/PPS/ピクチャヘッダ/スライスヘッダ/タイルグループヘッダ/LCU行/LCU群/LCU/CUにおいて信号通知された指示

ii. 現在のブロックおよび/またはその隣接のブロックのブロック寸法

iii. 現在のブロックおよび/またはその隣接のブロックのブロック形状

iv. カラーフォーマットの表示(例えば、4:2:0、4:4:4)

v. 別個または二重のコーディングツリー構造が使用されているかどうか

vi. スライスのタイプおよび/またはピクチャのタイプ

vii. 色成分の数

e. 上記の例は、BDPCM符号化ブロックを含む、または含まない変換ブロックおよび/または変換スキップブロックに適用されてもよい。

10

20

30

40

50

3. 符号フラグをコーディングするためのコンテキストインデックスオフセットは、以下のうちの1つ以上に基づいて決定されてもよい。

- a. S P S / V P S / P P S / ピクチャヘッダ / スライスヘッダ / タイルグループヘッダ / L C U 行 / L C U 群 / L C U / C U において信号通知された指示
- b. 現在のブロックおよび / またはその隣接のブロックのブロック寸法
- c. 現在のブロックおよび / またはその隣接のブロックのブロック形状
 - i. 一例において、符号フラグのコンテキスト値は、異なるブロック形状に対して別個であってもよい。
- d. 現在のブロックの隣接ブロックの予測モード (イントラ / インター)
 - i. 一例において、符号フラグのコンテキスト値は、イントラブロックとインターブロックとで別個であってもよい。
- e. 現在のブロックの隣接ブロックの B D P C M モードの表示
- f. カラーフォーマットの表示 (例えば、4 : 2 : 0、4 : 4 : 4)
- g. 別個または二重のコーディングツリー構造が使用されているかどうか
- h. スライスのタイプおよび / またはピクチャのタイプ
- i. 色成分の数
 - i. 1つの例において、標識フラグのコンテキスト値は、輝度色成分と彩度色成分とで別個であってもよい。

10

【化 8】

課題に対処するための解決策 2~5 :

20

用語「スライスユニット」は、以下のように定義される。

【化 9】

スライスユニット (S U) : 復号順序が連続し、正確に1つのコーディングされたスライスとそれに関連する全ての非 V C L N A L ユニットの含む N A L ユニットのセット。

4. 問題 2 を解決するために、全てのタイプの S E I メッセージの1つ以上を、S U 内で繰り返されるように許可されなくてもよい。

【化 10】

- a. 一例において、特定の内容を有する特定のペイロードタイプ値の任意の S E I メッセージを、S U 内で繰り返すように許可されず、この制約は、S U 内で *payloadType* の任意の値を有する同一の *sei_payload()* 構文構造の数は1以下でなければならない、と規定される。

30

- b. あるいは、特定のペイロードタイプ値の任意の S E I メッセージを (内容にかかわらず)、S U 内で繰り返されるように許可されない。

【化 11】

- i. 一例において、この制約は、以下のように規定される。S U 内で *payloadType* の任意の特定の値を有する *sei_payload()* 構文構造の数は、1以下でなくてはならない。

40

- c. あるいは、特定のペイロードタイプを有する任意の S E I メッセージを、S U 内で更新するように許可されない。

【化 12】

- i. 一例において、この制約は、以下のように規定される。S U 内で *payloadType* が特定の値を有する全ての *sei_payload()* 構文構造は、同じ内容を有するものとする。

5. 問題 3 を解決するために、P R E F I X _ S E I _ N U T および S U F F I X _ S E I _ N U T 以外の非 V C L N A L タイプの1つ以上について、S U 内の非 V C L N

50

ALユニットの繰り返しを許可しなくてもよい。

a. 1つの例において、以下の制約の1つ以上が指定される：

【化13】

i. 一例において、この制約は、以下のように規定される：1つのSU内でDCI NALユニットの数は、1以下でなくてはならない。

ii. 一例において、この制約は、以下のように規定される：1つのSU内でvps__video__parameter__set__idの特定の値を有するVPS NALユニットの数は、1以下でなくてはならない。

iii. 一例において、この制約は、以下のように規定される：1つのSUにおいてsps__seq__parameter__set__idの特定の値を有するSPS NALユニットの数は、1以下でなくてはならない。

iv. 一例において、この制約は、以下のように規定される：1つのSU内でpps__pic__parameter__set__idの特定の値を有するPPS NALユニットの数は、1以下でなくてはならない。

v. 一例において、この制約は、以下のように規定される：1つのSU内で、adaptation__parameter__set__idの特定の値を有し、かつ、aps__params__typeの特定の値を有するAPS NALユニットの数は、1以下でなくてはならない。

vi. 一例において、この制約は、以下のように規定される：1つのSU内で、nal__unit__typeの特定の値、adaptation__parameter__set__idの特定の値、およびaps__params__typeの特定の値を有するAPS NALユニットの数は、1以下でなくてはならない。

b. 代替的にまたは追加的に、DCI NALユニットを、CLVSまたはCVS内で繰り返すように許可されない。

c. 代替的にまたは追加的に、vps__video__parameter__set__idの特定の値を有するVPS NALユニットは、CLVSまたはCVS内で繰り返されるように許可されない。

d. 代替的にまたは追加的に、sps__seq__parameter__set__idの特定の値を有するSPS NALユニットは、CLVS内で繰り返されるように許可されない。

6. 問題4を解決するために、全てのタイプのAPSをレイヤにわたって共有してはならないことを規定してもよい。

a. あるいは、ALF APSをレイヤにわたって共有しないことを規定してもよい。

7. 問題5を解決するために、PU内でALF APSを更新することができることを規定してもよい。

a. あるいは、ピクチャユニット内で適応パラメータセットを更新することを許可しないことを規定してもよい。

8. 問題6を解決するために、nuh__layer__id値に関わらず、かつそれらがプレフィックスAPS NALユニットかサフィックスAPS NALユニットであるかどうかに関わらず、aps__params__typeの特定の値を有する全てのAPS NALユニットは、adaptation__parameter__set__idに対して同じ値空間を共有することが規定されてもよい。

a. あるいは、LMCS APS NALユニットは、nuh__layer__id値に関わらず、かつそれらがプレフィックスAPS NALユニットかサフィックスAPS NALユニットであるかどうかに関わらず、adaptation__parameter__set__idに対して同じ値空間を共有することが規定されてもよい。

b. あるいは、ALF APS NALユニットは、nuh__layer__id値に関わらず、かつそれらがプレフィックスAPS NALユニットであるかサフィックスAPS NALユニットであるかどうかに関わらず、adaptation__parameter__set__idに対して同じ値空間を共有することが規定されてもよい。

9. 課題6を解決するために、APS NALユニット(nal__unit__typeの特定の値、adaptation__parameter__set__idの特定の値、お

10

20

30

40

50

よび `aps_params_type` の特定の値) は CLVS 間で共有してはならないことが規定されてもよい。

a. 一例において、VCL NAL ユニット `vclNalUnitA` が参照する APS NAL ユニットは、`vclNalUnitA` を含む CLVS とは異なる CLVS 内の VCL NAL ユニットの関連付けられた非 VCL NAL ユニットであってはならないことが規定される。

b. あるいは、VCL NAL ユニット `vclNalUnitA` が参照する APS NAL ユニットは、`vclNalUnitA` が関連付けられた IRAP ピクチャと異なる IRAP ピクチャに関連付けられた VCL NAL ユニットの関連付けられた非 VCL NAL ユニットであってはならないことが規定される。

c. あるいは、VCL NAL ユニット `vclNalUnitA` によって参照される APS NAL ユニットは、`vclNalUnitA` が関連付けられた IRAP または GDR ピクチャとは異なる IRAP または GDR ピクチャに関連付けられた VCL NAL ユニットの関連付けられた非 VCL NAL ユニットであってはならないことが規定される。

d. あるいは、APS は CVS 間で共有されないように指定されてもよい。

i. あるいは、特定の APS タイプ (例えば、ALF、LMCS、SCALING) を有する APS は、CVS 間で共有されないように指定されてもよい。

【0070】

6 実施形態

【化14】

本実施形態は J V E T - Q 2 0 0 1 - v E に基づく。削除されたテキストには、開閉二重括弧 (例えば、`[[[]]]`) が付けられ、二重括弧の間には削除されたテキストが表示される。そして、追加されたテキストには**太字のイタリック体**の印が付けられる。

6. 1. 実施形態 # 1

transform skip mode についての構文要素 `coeff_sign_flag` のための `ctxInc` の導出処理

この処理への入力は、色成分インデックス `cIdx`、現在のピクチャの左上サンプルに対して現在の変換ブロックの左上サンプルを規定する輝度位置 (`x0`, `y0`)、現在の係数スキャン位置 (`xC`, `yC`) である。

このプロセスの出力は変数 `ctxInc` である。

変数 `leftSign` および `aboveSign` は、以下のように導出される。

$leftSign = (xC == 0) ? 0 : CoeffSignLevel[xC - 1][yC] \quad (1595)$

$aboveSign = (yC == 0) ? 0 : CoeffSignLevel[xC][yC - 1] \quad (1596)$

変数 `ctxInc` は、以下のように導出される。

- `leftSign` が 0 に等しく、`aboveSign` が 0 に等しい場合、または `leftSign` が `-aboveSign` に等しい場合、以下が適用される。

$ctxInc = (BdpcmFlag[x0][y0][cIdx] == 0 ? 0 : 3) \quad (1597)$

- `leftSign` が 0 以上かつ `aboveSign` が 0 以上である場合、以下が適用される。

$ctxInc = (BdpcmFlag[x0][y0][cIdx] == 0 ? 1 : 4) \quad (1598)$

- そうでない場合、以下が適用される。

$ctxInc = (BdpcmFlag[x0][y0][cIdx] == 0 ? 2 : 5) \quad (1599)$

【0071】

10

20

30

40

50

6.2 実施形態 2

【化15】

*transform skip mode*についての構文要素*coeff_sign_flag*のための*ctxInc*の導出処理

この処理への入力は、色成分インデックス*cIdx*、現在のピクチャの左上サンプルに対して現在の変換ブロックの左上サンプルを規定する輝度位置(*x0*, *y0*)、現在の係数スキャン位置(*xC*, *yC*)である。

このプロセスの出力は変数*ctxInc*である。

変数*leftSign*および*aboveSign*は、以下のように導出される。

$leftSign = (xC == 0) ? 0 : CoeffSignLevel[xC - 1][yC] \quad (1595)$

$aboveSign = (yC == 0) ? 0 : CoeffSignLevel[xC][yC - 1] \quad (1596)$

変数*ctxInc*は、以下のように導出される。

- *leftSign*が0に等しく、*aboveSign*が0に等しい場合、または*leftSign*が*-aboveSign*に等しい場合、以下が適用される。

$ctxInc = [[(BdpcmFlag[x0][y0][cIdx] == 0 ?)]] 0 [[: 3)]] \quad (1597)$

- *leftSign*が0以上かつ*aboveSign*が0以上である場合、以下が適用される。

$ctxInc = [[(BdpcmFlag[x0][y0][cIdx] ?)]] 1 [[: 4)]] \quad (1598)$

- そうでない場合、以下が適用される。

$ctxInc = [[(BdpcmFlag[x0][y0][cIdx] ?)]] 2 [[: 5)]] \quad (1599)$

- *BdpcmFlag[x0][y0][cIdx]*が1に等しい場合、以下が適用される。

$ctxInc = ctxInc + 3$

【0072】

6.3 実施形態#3

【化16】

*sb_coded_flag[xS][yS]*は、現在の変換ブロック内の位置(*xS*, *yS*)のサブブロックについて次のように規定し、サブブロックは、変換係数レベルの配列である。

- *sb_coded_flag[xS][yS]*が0に等しい場合、位置(*xS*, *yS*)におけるサブブロックの16個の変換係数レベルは、0に等しいと推論される。

- そうでない場合(*sb_coded_flag[xS][yS]*が1に等しい)、以下が適用される。

- (*xS*, *yS*)が(0, 0)と等しく、かつ(*LastSignificantCoeffX*, *LastSignificantCoeffY*)が(0, 0)と等しくない場合、(*xS*, *yS*)位置のサブブロックに対して[[16]] *numSbCoeff_sig_coeff_flag*の少なくとも1つの構文要素が存在する。

- あるいは、位置(*xS*, *yS*)のサブブロックの[[16]] *numSbCoeff*変換係数レベルの少なくとも1つが非ゼロ値を有する。

*sb_coded_flag[xS][yS]*が存在しない場合、1に等しいと推測される。

【0073】

6.4 実施形態#4

本実施形態は5~7項に対するものである。

7.4.3.5 適応パラメータセット意味論

各*APS* *RBS* *P*は、それが参照される前に復号処理で利用できるか、それを参照す

10

20

30

40

50

るコーディングされたスライスNALユニットのTemporalId以下のTemporalIdを有するつ少なくとも一つのAU内に含まれるか、外部手段を通じて提供されるものとする。

【化17】

adaptation_parameter_set_idの特定の値を有し、かつPU内のLMCS_APSまたはSCALING_APSに等しいaps_params_typeの特定の値を有するすべてのAPS NALユニットは、それらがプレフィックスであるかサフィックスAPS NALユニットであるかどうかに関わらず、同じコンテンツを有するものとする。

adaptation_parameter_set_idは、他の構文要素が参照するAPSの識別子を提供する。

10

aps_params_typeがALF_APSまたはSCALING_APSに等しい場合、adaptation_parameter_set_idの値は0~7の範囲に含まれるものとする。

aps_params_typeがLMCS_APSに等しい場合、adaptation_parameter_set_idの値は0~3の範囲にあるものとする。

apsLayerIdを特定のAPS NALユニットのnuh_layer_idの値とし、vclLayerIdを特定のVCL NALユニットのnuh_layer_idの値とする。特定のVCL NALユニットは、apsLayerIdがvclLayerIdに[[以下]]であり、[[nuh_layer_idがapsLayerIdであるレイヤが、vclLayerIdであるnuh_layer_idを有するレイヤを含む少なくとも一つのOLSに含まれていない限り、特定のAPS NALユニットを参照しないものとする。]]

20

【化18】

aps_params_typeは、表6に示されるように、APSにおいて実行されるAPSパラメータのタイプを規定する。

1. 表6 - APSパラメータのタイプコードおよびAPSパラメータのタイプ

【表22】

aps_params_type	aps_params_typeの名称	APSパラメータの種類
0	ALF_APS	ALFパラメータ
1	LMCS_APS	LMCSパラメータ
2	SCALING_APS	スケーリングリストパラメータ
3..7	確保済み	確保済み

30

【化19】

aps_params_typeの特定の値を有する全てのAPS NALユニットは、nuh_layer_id値に関わらず、かつそれらがプレフィックスAPS NALユニットかサフィックスAPS NALユニットであるかどうかに関わらず、adaptation_parameter_set_idに対して同じ値空間を共有する。aps_params_typeの値が異なるAPS NALユニットは、adaptation_parameter_set_idに別個の値空間を使用する。

40

注1-APS NALユニット(nal_unit_typeの特定の値およびaps_params_typeの特定の値を有する)は、ピクチャ間で共有されてもよく、ピクチャ内の異なるスライスは、異なるALF_APSを参照することができる。

注2-特定のVCL NALユニットに関連付けられたサフィックスAPS NALユニット(このVCL NALユニットは、復号順でサフィックスAPS NALユニットに先行する)は、特定のVCL NALユニットによって使用されるものではなく、サフィックスAPS NALユニットに続くVCL NALユニットによって復号順で使用されるものである。

50

【化 2 0】

0に等しいaps_extension_flagは、APS RBSP構文構造にaps_extension_data_flag構文要素が含まれていないことを規定する。1に等しいaps_extension_data_flagは、APS RBSP構文構造にaps_extension_data_flag構文要素が存在することを規定する。

aps_extension_data_flagは任意の値を有することができる。その存在および値は、本明細書バージョンで特定された特徴に対するのデコーダの適合性に影響を与えない。本明細書バージョンに準拠するデコーダは、すべてのaps_extension_data_flag構文要素を無視しなければならない。

【0074】

10

6.5. 符号フラグをコーディングするための実施例

一実施例において、RemCcb sが0より小さい場合、バイパスコーディングが用いられる。

表 1 3 1 - コンテキスト符号化ピンを有する構文要素へのctxIncの割り当て

【表 2 3】

構文要素	binIdx					
	0	1	2	3	4	>= 5
coeff_sign_flag[] transform_skip_flag[x0][y0][cIdx] == 0 RemCcb s < 0 [== 0] slice_ts_residual_coding_disabled_flag	bypass	na	na	na	na	na
coeff_sign_flag[] transform_skip_flag[x0][y0][cIdx] == 1 && RemCcb s >= 0 && !slice_ts_residual_coding_disabled_flag	0.5 (9. 3. 4. 2. 10項)	na	na	na	na	na

20

30

【化 2 1】

あるいは、一実施例において、現在のブロックがトランスフォームスキップモードでコーディングされ、RemCcb sが0より大きく、および!slice_ts_residual_coding_disabled_flagのとき、コンテキストコーディングが用いられる。

表 1 3 1 - コンテキスト符号化ピンを有する構文要素へのctxIncの割り当て

40

50

【表 2 4】

構文要素	binIdx					
	0	1	2	3	4	>= 5
coeff_sign_flag[] transform_skip_flag[x0][y0][cIdx] == 0 RemCcbcs == 0 slice_ts_residual_co ding_disabled_flag	bypass	na	na	na	na	na
coeff_sign_flag[] transform_skip_flag[x0][y0][cIdx] == 1 && RemCcbcs >[[=]] 0 && !slice_ts_residua l_coding_disabled_fl ag	0..5 (9. 3. 4. 2. 10項)	na	na	na	na	na

10

あるいは、一実施例において、RemCcbcsが4より小さい場合、バイパスコーディングが用いられる。

表 1 3 1 - コンテキスト符号化ピンを有する構文要素へのctxIncの割り当て

【表 2 5】

20

構文要素	binIdx					
	0	1	2	3	4	>= 5
coeff_sign_flag[] transform_skip_flag[x0][y0][cIdx] == 0 RemCcbcs < 4 [[== 0]] slice_ts_residual_co ding_disabled_flag	bypass	na	na	na	na	na
coeff_sign_flag[] transform_skip_flag[x0][y0][cIdx] == 1 && RemCcbcs >=4 [[0]] && !slice_ts_residua l_coding_disabled_fl ag	0..5 (9. 3. 4. 2. 10項)	na	na	na	na	na

30

【0075】

図1は、本明細書で開示される様々な技術が実装され得る例示的な映像処理システム1900を示すブロック図である。様々な実装形態は、システム1900のモジュールの一部または全部を含んでもよい。システム1900は、映像コンテンツを受信するための入力ユニット1902を含んでもよい。映像コンテンツは、未加工または非圧縮フォーマット、例えば、8または10ビットのマルチコンポーネント画素値で受信されてもよく、または圧縮または符号化フォーマットで受信されてもよい。入力ユニット1902は、ネットワークインターフェース、周辺バスインターフェース、または記憶インターフェースを表してもよい。ネットワークインターフェースの例は、イーサネット（登録商標）、パッシブ光ネットワーク（PON）等の有線インターフェース、およびWi-Fiまたはセルラーインターフェース等の無線インターフェースを含む。

40

【0076】

システム1900は、本明細書に記載される様々なコーディングまたは符号化方法を実装することができるコーディングモジュール1904を含んでもよい。コーディングコン

50

ポーネント 1904 は、入力 1902 からの映像の平均ビットレートをコーディングコンポーネント 1904 の出力に低減し、映像のコーディング表現を生成してもよい。従って、このコーディング技術は、映像圧縮または映像トランスコーディング技術と呼ばれることがある。コーディングコンポーネント 1904 の出力は、コンポーネント 1906 によって表されるように、記憶されてもよいし、接続された通信を介して送信されてもよい。入力 1902 において受信された、記憶されたまたは通信された映像のビットストリーム（またはコーディングされた）表現は、コンポーネント 1908 によって使用されて、表示インターフェース 1910 に送信される画素値または表示可能な映像を生成してもよい。ビットストリーム表現からユーザが見ることができる映像を生成する処理は、映像展開と呼ばれることがある。さらに、特定の映像処理動作を「コーディング」動作またはツールと呼ぶが、コーディングツールまたは動作は、エンコーダで使用され、対応する復号ツールまたは動作でありコーディングの結果を逆にするものは、デコーダによって行われることが理解されよう。

10

【0077】

周辺バスインターフェースまたは表示インターフェースの例は、ユニバーサルシリアルバス（USB）または高精細マルチメディアインターフェース（HDMI）またはディスプレイポート等を含んでもよい。ストレージインターフェースの例は、SATA（Serial Advanced Technology Attachment）、PCI、IDE インターフェース等を含む。本明細書に記載される技術は、携帯電話、ノートパソコン、スマートフォン、またはデジタルデータ処理および/または映像表示を実施可能な他のデバイス等の様々な電子デバイスに実施されてもよい。

20

【0078】

図 2 は、映像処理装置 3600 のブロック図である。装置 3600 は、本明細書に記載の方法の 1 つ以上を実装するために使用されてもよい。装置 3600 は、スマートフォン、タブレット、コンピュータ、モノのインターネット（IoT）受信機等にも実施されてもよい。装置 3600 は、1 つ以上の処理装置 3602 と、1 つ以上のメモリ 3604 と、映像処理ハードウェア 3606 と、を含んでもよい。1 つまたは複数のプロセッサ 3602 は、本明細書に記載される 1 つ以上の方法を実装するように構成されてもよい。メモリ（複数可）3604 は、本明細書で説明される方法および技術を実装するために使用されるデータおよびコードを記憶するために使用してもよい。映像処理ハードウェア 3606

30

【0079】

図 4 は、本開示の技法を利用し得る例示的な映像符号化システム 100 を示すブロック図である。

【0080】

図 4 に示すように、映像コーディングシステム 100 は、送信元デバイス 110 と、送信先デバイス 120 と、を備えてもよい。送信元デバイス 110 は、映像コーディング機器とも称され得る符号化された映像データを生成する。送信先装置 120 は、送信元装置 110 によって生成された、コーディング映像データを復号してよく、映像復号デバイスと呼ばれ得る。

40

【0081】

送信元デバイス 110 は、映像ソース 112 と、映像エンコーダ 114 と、入出力（I/O）インターフェース 116 と、を含んでもよい。

【0082】

映像ソース 712 は、映像キャプチャデバイスなどのソース、映像コンテンツプロバイダからの映像データを受信するためのインターフェース、および/または映像データを生成するためのコンピュータグラフィックスシステム、またはこれらのソースの組み合わせたものを含んでもよい。映像データは、1 または複数のピクチャを含んでもよい。映像エンコーダ 114 は、映像ソース 112 からの映像データを符号化し、ビットストリームを生成する。ビットストリームは、映像データのコーディング表現を形成するビットのシー

50

ケンスを含んでもよい。ビットストリームは、コーディングされたピクチャおよび関連付けられたデータを含んでもよい。コーディングされたピクチャは、ピクチャのコーディング表現である。関連付けられたデータは、シーケンスパラメータセット、ピクチャパラメータセット、および他の構文構造を含んでもよい。I/Oインターフェース116は、変復調器(モデム)および/または送信機を含んでもよい。符号化された映像データは、ネットワーク130aを介して、I/Oインターフェース116を介して送信先デバイス120に直接送信されてよい。符号化された映像データは、送信先デバイス120によるアクセスのために、記録媒体/サーバ130b上に記憶されてもよい。

【0083】

送信先デバイス120は、I/Oインターフェース126、映像デコーダ124、および表示装置122を含んでもよい。

10

【0084】

I/Oインターフェース126は、受信機および/またはモデムを含んでもよい。I/Oインターフェース126は、送信元デバイス110または記憶媒体/サーバ130bから符号化された映像データを取得してもよい。映像デコーダ124は、符号化された映像データを復号してもよい。表示装置122は、復号された映像データをユーザに表示してもよい。表示装置122は、送信先デバイス120と一体化されてもよく、または外部表示装置とインターフェースするように構成される送信先デバイス120の外部にあってもよい。

【0085】

映像エンコーダ114および映像デコーダ124は、高効率映像符号化(HEVC)規格、汎用映像符号化(VVVM)規格、および他の現在のおよび/または更なる規格等の映像圧縮規格に従って動作してもよい。

20

【0086】

図5は、映像エンコーダ200の一例を示すブロック図であり、図4に示されるシステム100における映像エンコーダ114であってもよい。

【0087】

映像エンコーダ200は、本開示の技術のいずれかまたは全部を実行するように構成されてもよい。図5の例において、映像エンコーダ200は、複数の機能コンポーネントを含む。本開示で説明される技法は、映像エンコーダ200の様々なモジュール間で共有されてもよい。いくつかの例では、プロセッサは、本開示で説明される技術のいずれかまたはすべてを行うように構成してもよい。

30

【0088】

映像エンコーダ200の機能コンポーネントは、分割ユニット201、予測ユニット202を含んでもよく、予測ユニット202は、モード選択ユニット203、動き推定ユニット204、動き補償ユニット205、およびイントラ予測ユニット206、残差生成ユニット207、変換ユニット208、量子化ユニット209、逆量子化ユニット210、逆変換ユニット211、再構成ユニット212、バッファ213、およびエントロピーエンコーディングユニット214を含んでもよい。

【0089】

他の例において、映像エンコーダ200は、より多く、より少ない、または異なる機能コンポーネントを含んでもよい。一例において、予測ユニット202は、イントラブロックコピー(IBC)ユニットを含んでもよい。IBCユニットは、少なくとも1つの参照ピクチャが現在の映像ブロックが位置するピクチャであるIBCモードにおいて予測を行うことができる。

40

【0090】

さらに、動き推定部204および動き補償部205などのいくつかのモジュールは、高度に統合されてもよいが、説明のために、図5の例においては別個に表現されている。

【0091】

分割部201は、ピクチャを1または複数の映像ブロックに分割してもよい。映像エン

50

コーダ 200 および映像デコーダ 300 は、様々な映像ブロックサイズをサポートしてもよい。

【0092】

モード選択ユニット 203 は、例えば、エラー結果に基づいて、イントラまたはインターのいずれかのコーディングモードの 1 つを選択し、得られたイントラまたはインターコーディングブロックを、残差生成ユニット 207 に供給して残差ブロックデータを生成し、再構成ユニット 212 に供給して参照ピクチャとしての使用のために符号化ブロックを再構成してもよい。いくつかの例において、モード選択ユニット 203 は、インター予測信号およびイントラ予測信号に基づいて予測を行う C I I P (C o m b i n a t i o n o f I n t r a a n d I n t e r P r e d i c t i o n) モードを選択してもよい。また、モード選択ユニット 203 は、インター予測の場合、ブロックの動きベクトルの解像度 (例えば、サブピクセルまたは整数画素精度) を選択してもよい。

10

【0093】

現在の映像ブロックに対してインター予測を行うために、動き推定ユニット 204 は、バッファ 213 からの 1 または複数の参照フレームと現在の映像ブロックとを比較することで、現在の映像ブロックのための動き情報を生成してもよい。動き補償部 205 は、現在の映像ブロックに関連付けられたピクチャ以外のバッファ 213 からのピクチャの動き情報および復号されたサンプルに基づいて、現在の映像ブロックのための予測映像ブロックを判定してもよい。

【0094】

動き推定部 204 および動き補償部 205 は、現在の映像ブロックが I スライス、P スライス、または B スライスであるかに基づいて、例えば、現在の映像ブロックに対して異なる動作を行ってもよい。

20

【0095】

いくつかの例では、動き推定ユニット 204 は、現在の映像ブロックに対して単一方向予測を行い、動き推定ユニット 204 は、現在の映像ブロックに対して、参照映像ブロック用のリスト 0 またはリスト 1 の参照ピクチャを検索してもよい。そして、動き推定ユニット 204 は、参照映像ブロックと、現在の映像ブロックと参照映像ブロックとの間の空間的変位を示す動きベクトルとを含む、リスト 0 またはリスト 1 における参照ピクチャを示す参照インデックスを生成してもよい。動き推定ユニット 204 は、参照インデックス、予測方向インジケータ、および動きベクトルを、現在の映像ブロックの動き情報として出力してもよい。動き補償ユニット 205 は、現在の映像ブロックの動き情報が示す参照映像ブロックに基づいて、現在のブロックの予測映像ブロックを生成してもよい。

30

【0096】

他の例において、動き推定部 204 は、現在の映像ブロックに双方向予測を実行してもよく、動き推定部 204 は、リスト 0 における参照ピクチャの中から現在の映像ブロックを求めるための参照映像ブロックを検索してもよく、また、リスト 1 における参照ピクチャの中から現在の映像ブロックを求めるための別の参照映像ブロックを検索してもよい。そして、動き推定ユニット 204 は、参照映像ブロックを含むリスト 0 およびリスト 1 における参照ピクチャを示す参照インデックスと、参照映像ブロックと現在の映像ブロックとの間の空間的変位を示す動きベクトルとを生成してもよい。動き推定ユニット 204 は、現在の映像ブロックの参照インデックスおよび動きベクトルを、現在の映像ブロックの動き情報として出力してもよい。動き補償部 205 は、現在の映像ブロックの動き情報が示す参照映像ブロックに基づいて、現在の映像ブロックの予測映像ブロックを生成してもよい。

40

【0097】

いくつかの例では、動き推定部 204 は、デコーダの復号処理のために、動き情報のフルセットを出力してもよい。

【0098】

いくつかの例では、動き推定ユニット 204 は、現在の映像のための動き情報のフルセ

50

ットを出力しなくてもよい。むしろ、動き推定ユニット204は、別の映像ブロックの動き情報を参照して、現在の映像ブロックの動き情報を信号通知してもよい。例えば、動き推定ユニット204は、現在の映像ブロックの動き情報が隣接の映像ブロックの動き情報に十分に類似していることを判定してもよい。

【0099】

一例において、動き推定部204は、現在の映像ブロックに関連付けられた構文構造において、現在の映像ブロックが他の映像ブロックと同じ動き情報を有することを映像デコーダ300に示す値を示してもよい。

【0100】

他の例において、動き推定部204は、現在の映像ブロックに関連付けられた構文構造において、別の映像ブロックと、動きベクトル差分(MVD; Motion Vector Difference)とを識別してもよい。動きベクトル差分は、現在の映像ブロックの動きベクトルと、示された映像ブロックの動きベクトルとの差分を示す。映像デコーダ300は、指示された映像ブロックの動きベクトルおよび動きベクトル差分を使用して、現在の映像ブロックの動きベクトルを決定してもよい。

10

【0101】

上述したように、映像エンコーダ200は、動きベクトルを予測的に信号通知してもよい。映像エンコーダ200によって実装され得る予測信号通知技法の2つの例は、高度動きベクトル予測(AMVP)およびマージモード信号通知を含む。

【0102】

イントラ予測ユニット206は、現在の映像ブロックに対してイントラ予測を行ってもよい。イントラ予測ユニット206が現在の映像ブロックをイントラ予測する場合、イントラ予測ユニット206は、同じピクチャにおける他の映像ブロックの復号されたサンプルに基づいて、現在の映像ブロックのための予測データを生成してもよい。現在の映像ブロックのための予測データは、予測された映像ブロックおよび様々な構文要素を含んでもよい。

20

【0103】

残差生成ユニット207は、現在の映像ブロックから現在の映像ブロックの予測された映像ブロックを減算することによって(例えば、マイナス符号によって示されている)、現在の映像ブロックのための残差データを生成してもよい。現在の映像ブロックの残差データは、現在の映像ブロックにおけるサンプルの異なるサンプル成分に対応する残差映像ブロックを含んでもよい。

30

【0104】

他の例において、例えば、スキップモードにおいて、現在の映像ブロックのための残差データがなくてもよく、残差生成ユニット207は、減算演算を行わなくてもよい。

【0105】

変換処理ユニット208は、現在の映像ブロックに関連付けられた残差映像ブロックに1または複数の変換を適用することによって、現在の映像ブロックのための1または複数の変換係数映像ブロックを生成してもよい。

【0106】

変換処理ユニット208が現在の映像ブロックに関連付けられた変換係数映像ブロックを生成した後、量子化ユニット209は、現在の映像ブロックに関連付けられた1または複数の量子化パラメータ(QP: Quantization Parameter)値に基づいて、現在の映像ブロックに関連付けられた変換係数映像ブロックを量子化してもよい。

40

【0107】

逆量子化ユニット210および逆変換ユニット211は、変換係数映像ブロックに逆量子化および逆変換をそれぞれ適用し、変換係数映像ブロックから残差映像ブロックを再構成してもよい。再構成部212は、予測部202によって生成された1または複数の予測映像ブロックから対応するサンプルに再構成された残差映像ブロックを追加して、バッフ

50

ア 2 1 3 に格納するための現在のブロックに関連付けられた再構成された映像ブロックを生成してもよい。

【 0 1 0 8 】

再構成ユニット 2 1 2 が映像ブロックを再構成した後、映像ブロックにおける映像ブロックアーチファクトを縮小するために、ループフィルタリング動作が行われてもよい。

【 0 1 0 9 】

エントロピーエンコーディングユニット 2 1 4 は、映像エンコーダ 2 0 0 の他の機能コンポーネントからデータを受信してもよい。エントロピー符号化部 2 1 4 がデータを受信した場合、エントロピー符号化部 2 1 4 は、1 または複数のエントロピー符号化動作を行い、エントロピー符号化されたデータを生成し、エントロピー符号化されたデータを含むビットストリームを出力してもよい。

10

【 0 1 1 0 】

開示される技術のいくつかの実施形態は、映像処理ツールまたはモードを有効化するように決定または判定することを含む。一例において、映像処理ツールまたはモードが有効化される場合、エンコーダは、1 つの映像ブロックを処理する際にツールまたはモードを使用するまたは実装するが、ツールまたはモードの使用に基づいて、結果として得られるビットストリームを必ずしも修正しなくてもよい。すなわち、映像のブロックから映像のビットストリーム（またはビットストリーム表現）への変換は、決定または判定に基づいて映像処理ツールまたはモードが有効化される場合に、映像処理ツールまたはモードを使用する。別の例において、映像処理ツールまたはモードが有効化される場合、デコーダは、ビットストリームが映像処理ツールまたはモードに基づいて修正されたことを知って、ビットストリームを処理する。すなわち、決定または判定に基づいて有効化された映像処理ツールまたはモードを使用して、映像のビットストリームから映像のブロックへの変換を行う。

20

【 0 1 1 1 】

図 6 は、映像デコーダ 3 0 0 の一例を示すブロック図であり、映像デコーダ 3 0 0 は、図 4 に示すシステム 1 0 0 における映像デコーダ 1 1 4 であってもよい。

【 0 1 1 2 】

映像デコーダ 3 0 0 は、本開示の技術のいずれかまたは全部を実行するように構成されてもよい。図 6 の例において、映像デコーダ 3 0 0 は、複数の機能コンポーネントを含む。本開示で説明される技法は、映像デコーダ 3 0 0 の様々なコンポーネント間で共有されてもよい。いくつかの例では、プロセッサは、本開示で説明される技術のいずれかまたはすべてを行うように構成してもよい。

30

【 0 1 1 3 】

図 6 の実施例において、映像デコーダ 3 0 0 は、エントロピー復号部 3 0 1、動き補償部 3 0 2、イントラ予測部 3 0 3、逆量子化部 3 0 4、逆変換部 3 0 5、および再構成部 3 0 6、並びにバッファ 3 0 7 を含む。映像デコーダ 3 0 0 は、いくつかの例では、映像エンコーダ 2 0 0（図 5）に関して説明したエンコーディングパスとほぼ逆のデコーディングパスを行ってもよい。

【 0 1 1 4 】

エントロピー復号部 3 0 1 は、符号化されたビットストリームを取り出す。符号化されたビットストリームは、エントロピー符号化された映像データ（例えば、映像データの符号化されたブロック）を含んでもよい。エントロピー復号ユニット 3 0 1 は、エントロピー符号化された映像データを復号し、エントロピー復号された映像データから、動き補償ユニット 3 0 2 は、動きベクトル、動きベクトル精度、参照ピクチャリストインデックス、および他の動き情報を含む動き情報を決定してもよい。動き補償ユニット 3 0 2 は、例えば、AMVP およびマージモードを実行することで、このような情報を判定してもよい。

40

【 0 1 1 5 】

動き補償ユニット 3 0 2 は、動き補償されたブロックを生成してもよく、場合によっては、補間フィルタに基づいて補間を実行する。構文要素には、サブピクセルの精度で使用

50

される補間フィルタのための識別子が含まれてもよい。

【0116】

動き補償部302は、映像ブロックの符号化中に映像エンコーダ200によって使用されるような補間フィルタを使用して、参照ブロックのサブ整数画素のための補間値を計算してもよい。動き補償ユニット302は、受信した構文情報に基づいて、映像エンコーダ200が使用する補間フィルタを決定し、補間フィルタを使用して予測ブロックを生成してもよい。

【0117】

動き補償部302は、構文情報の一部を用いて、符号化された映像シーケンスのフレームおよび/またはスライスを符号化するために使用されるブロックのサイズ、符号化された映像シーケンスのピクチャの各マクロブロックがどのように分割されるかを記述する分割情報、各分割がどのように符号化されるかを示すモード、各インター符号化されたブロックに対する1または複数の参照フレーム（および参照フレームリスト）、および符号化された映像シーケンスを復号するための他の情報を決定してもよい。

10

【0118】

イントラ予測ユニット303は、例えば、ビットストリームにおいて受信したイントラ予測モードを使用して、空間的に隣接するブロックから予測ブロックを形成してもよい。逆量子化ユニット303は、ビットストリームに提供され、エントロピー復号ユニット301によって復号された量子化された映像ブロック係数を逆量子化（すなわち、逆量子化）する。逆変換ユニット303は、逆変換を適用する。

20

【0119】

再構成部306は、残差ブロックと、動き補償部202またはイントラ予測部303によって生成された対応する予測ブロックとを合計し、復号されたブロックを形成してもよい。所望であれば、ブロックアーチファクトを除去するために、復号されたブロックをフィルタリングするためにデブロックングフィルタを適用してもよい。復号された映像ブロックは、バッファ307に記憶され、バッファ307は、後続の動き補償/イントラ予測のために参照ブロックを提供し、表示装置に表示するために復号された映像を生成する。

【0120】

次に、いくつかの実施形態において好適な解決策を列挙する。

【0121】

以下の解決策は、前章（例えば、項目1）で論じた技術の例示的な実施形態を示す。

30

【0122】

1. 本発明の映像加工方法（例えば、図3に示される方法3000）は、映像の現在のブロックとこの映像のコーディングされた表現との変換（3002）を行うことを含み、このコーディングされた表現は、コーディングされた表現におけるこの現在のブロックの符号フラグをコーディングするために使用されるコンテキストインデックスオフセットが、ある範囲内にあることを規定するフォーマット規則に準拠し、この範囲はこのコーディングされた表現におけるこの現在のブロックを表現するために用いられるコーディングモードの範囲関数である。

【0123】

2. 前記規則は、コーディングモードが特定のモードである場合、前記範囲が $[N0, N1]$ であり、そうでない場合、 $[N2, N3]$ であることを規定する、解決策1に記載の方法。

40

【0124】

3. 前記特定のモードは、彩度または輝度ブロック差動パルス符号変調モードである、解決策2に記載の方法。

【0125】

以下の解決策は、前章（例えば、項目2）で論じた技術の例示的な実施形態を示す。

【0126】

4. 映像の現在のブロックとこの映像のコーディングされた表現との変換を行うことを

50

含み、このコーディングされた表現は、この現在のブロックの符号フラグが、コンテキストコーディングされたピンまたはバイパスモードのうちの1つを、残りのコンテキストコーディングされたピンの数に応じて使用してコーディングされた表現でコーディングされることを規定するフォーマット規則に準拠する、映像処理方法

【0127】

5. 前記バイパスモードは、残りのコンテキストコーディングされたピンの数がNよりも小さく、Nが正の整数である場合、かつその場合に限りコーディングに使用される、解決策4に記載の方法。

【0128】

6. 前記バイパスモードは、残りのコンテキストコーディングされたピンの数がNに等しく、Nが正の整数である場合、かつその場合に限りコーディングに使用される、解決策5に記載の方法。

10

【0129】

以下の解決策は、前章（例えば、項目2）で論じた技術の例示的な実施形態を示す。

【0130】

7. 映像の現在のブロックと前記映像のコーディングされた表現との変換を行うことを含み、前記変換は、ブロックベースのデルタパルスコード変調（BDPCM）モードを使用し、前記コーディングされた表現は、前記符号フラグをコーディングするためのコンテキストインデックスオフセットが前記現在のブロックのコーディング条件の関数となるように、前記BDPCMからの符号フラグを前記コーディングされた表現においてコンテキストコーディングすることを規定するフォーマット規則に準拠する、映像処理方法。

20

【0131】

8. 前記コーディング条件は、シーケンスパラメータセットまたは映像パラメータセットまたはピクチャパラメータセットまたはピクチャヘッダまたはスライスヘッダまたはタイル群ヘッダまたは論理コーディングユニットレベルまたはLCU群またはコーディングユニットレベルにおける前記コーディングされた表現に含まれる構文要素に対応する、解決策7に記載の方法。

【0132】

9. 前記コーディング条件は、前記現在のブロックおよび/または隣接ブロックのブロック寸法に対応する、解決策7に記載の方法。

30

【0133】

以下の解決策は、前章（例えば、項目4）で論じた技術の例示的な実施形態を示す。

【0134】

10. 映像の現在のブロックと前記映像のコーディングされた表現との変換を行うことを含み、前記コーディングされた表現は、連続した復号順にあり、かつ単一のコーディングされたスライスを含むネットワーク抽象化レイヤユニットのセットに対応する、前記コーディングされた表現におけるスライスユニットが、補足拡張情報（SEI）の少なくとも部分を最大でも1回含むように許可されることを規定するフォーマット規則に準拠する、映像処理方法。

【0135】

11. SEIの前記部分がSEI全体に対応する、解決策10に記載の方法。

40

【0136】

12. SEIの前記部分は、特定のタイプのSEI情報フィールドに対応する、解決策10に記載の方法。

【0137】

以下の解決策は、前章（例えば、項目5）で論じた技術の例示的な実施形態を示す。

【0138】

13. 映像の現在のブロックと前記映像のコーディングされた表現との変換を行うことを含み、前記コーディングされた表現は、連続した復号順にあり、かつ単一のコーディングされたスライスを含むネットワーク抽象化レイヤユニットのセットに対応する、前記コ

50

ーディングされた表現におけるスライスユニットが、1つ以上の映像コーディングレイヤネットワーク抽象化レイヤ(VCLNAL)ユニットを含むことを規定するフォーマット規則に準拠し、前記フォーマット規則はさらに、前記スライスユニットにおいて、繰り返されるように許可される第1のタイプのユニットと繰り返されるように許可されない第2のタイプのユニットとを規定する、映像処理方法。

【0139】

14. 前記第2のタイプのユニットが、ある範囲のタイプを有するユニットを含む、解決策13に記載の方法。

【0140】

以下の解決策は、前章(例えば、項目6、7)で論じた技術の例示的な実施形態を示す。

10

【0141】

15. 規則に基づいて、1つ以上の映像レイヤを含む映像と前記映像のコーディングされた表現との変換を行うことを含み、前記コーディングされた表現は、1つ以上の適応パラメータセット(APS)を含み、前記規則は、前記1つ以上のAPSのうちいくつかの前記1つ以上の映像レイヤの前記変換への適用可能性を規定する、映像処理方法。

【0142】

16. 前記規則は、前記1つ以上のAPSの全てが前記1つ以上の映像レイヤにわたって共有されることは許可されないことを規定する、解決策15に記載の方法。

【0143】

以下の解決策は、前章(例えば、項目8、9)で論じた技術の例示的な実施形態を示す。

20

【0144】

17. 規則に従って、1つ以上の映像レイヤを含む映像と前記映像のコーディングされた表現との変換を行うことを含み、前記コーディングされた表現は、1つ以上のネットワーク抽象化レイヤ(NAL)ユニット中に構成され、前記コーディングされた表現は、前記変換の特徴を制御するための1つ以上の適応パラメータセットを含む、映像処理方法。

【0145】

18. 前記規則は、特定のタイプを有する全ての適応パラメータセットもまた、それぞれの識別子値に対して同じ空間値を共有することを規定する、解決策17に記載の方法。

【0146】

19. 前記規則は、前記1つ以上の映像レイヤが特定の適応パラメータセットネットワーク抽象化レイヤユニットを共有するように許可されないことを規定する、解決策17に記載の方法。

30

【0147】

20. 変換は、映像をコーディング表現に符号化することを含む、解決策1~19のいずれかに記載の方法。

【0148】

21. 変換は、映像の画素値を生成するためにコーディング表現を復号することを含む、解決策1~19のいずれかに記載の方法。

【0149】

22. 解決策1~21の1つ以上に記載の方法を実装するように構成された処理装置を備える、映像復号装置。

40

【0150】

23. 解決策1~21の1つ以上に記載の方法を実装するように構成された処理装置を備える、映像符号化装置。

【0151】

24. コンピュータコードが記憶されたコンピュータプログラム製品であって、コードは、プロセッサにより実行されると、プロセッサに、解決策1~21のいずれかに記載の方法を実装させるコンピュータプログラム製品。

【0152】

25. 本明細書に記載の方法、装置またはシステム。

50

【 0 1 5 3 】

図 9 は、映像処理の方法 9 0 0 の一例を示すフローチャートである。動作 9 0 2 は、映像の現在のブロックと前記映像のビットストリームとの変換を行うことを含み、前記ビットストリームは、前記ビットストリームに第 1 の係数の第 1 の符号フラグを含めるためにコンテキストインデックスオフセットを使用することを規定する規則に準拠し、前記規則は、前記コンテキストインデックスオフセットの値が、前記ビットストリームにおける前記現在のブロックに第 1 のコーディングモードが適用されるかどうかに基づくことを規定する。

【 0 1 5 4 】

方法 9 0 0 のいくつかの実施例において、前記規則は、前記第 1 のコーディングモードが適用されない場合、前記コンテキストインデックスオフセットの前記値が $N_0 \sim N_1$ の第 1 の範囲内にあることを規定し、前記第 1 のコーディングモードが適用される場合、前記規則は、前記コンテキストインデックスオフセットの前記値が $N_2 \sim N_3$ の第 2 の範囲内にあることを規定する。方法 9 0 0 のいくつかの実施例において、前記第 1 のコーディングモードにおいて、パルスコーディング変調表現を用いて、前記ビットストリームにおいて、量子化残差と前記量子化残差の予測との差を表現する。方法 9 0 0 のいくつかの実施例において、 N_0 、 N_1 、 N_2 、 N_3 はそれぞれ、0、2、3、5 である。方法 9 0 0 のいくつかの実施例において、第 1 の隣接符号値および第 2 の隣接符号値が第 1 の条件を満たすことに応じて、前記コンテキストインデックスオフセットの前記値は 0 または 3 に等しく、前記第 1 の条件は、(1) 前記第 1 の隣接符号値は 0 に等しく、第 2 の隣接符号値が 0 に等しい、または (2) 前記第 1 の隣接符号値は、前記第 2 の隣接符号値の負数に等しい、であり、ここで、前記第 1 の隣接符号値は前記第 1 の係数の水平座標が 0 であることに応じて 0 に等しく、前記第 1 の係数の前記水平座標がゼロでないことに応じて左隣接係数の符号値に等しく、前記第 2 の隣接符号値は前記第 1 の係数の垂直座標が 0 であることに応じて 0 に等しく、前記第 1 の係数の前記垂直座標が 0 でないことに応じて、上側の隣接係数の符号値に等しい。

【 0 1 5 5 】

方法 9 0 0 のいくつかの実施例において、前記第 1 のコーディングモードが前記現在のブロックに適用されないことに応じて、前記コンテキストインデックスオフセットの値は 0 に等しく、前記第 1 のコーディングモードが前記現在のブロックに適用されることに応じて、前記コンテキストインデックスオフセットの値は 3 に等しい。方法 9 0 0 のいくつかの実施例において、前記第 1 の隣接符号値および前記第 2 の隣接符号値が前記第 1 の条件を満たさないことに応じて、かつ前記第 1 の隣接符号値および前記第 2 の隣接符号値が第 2 の条件を満たすことに応じて、前記コンテキストインデックスオフセットの値は 1 または 4 に等しく、前記第 2 の条件は、(1) 前記第 1 の隣接符号値は 0 以上で、前記第 2 の隣接符号値は 0 以上、である。方法 9 0 0 のいくつかの実施例において、前記第 1 のコーディングモードが前記現在のブロックに適用されないことに応じて、前記コンテキストインデックスオフセットの前記値は 1 に等しく、前記第 1 のコーディングモードが前記現在のブロックに適用されることに応じて、前記コンテキストインデックスオフセットの前記値は 4 に等しい。方法 9 0 0 のいくつかの実施例において、前記第 1 の隣接符号値および前記第 2 の隣接符号値が前記第 1 の条件および前記第 2 の条件を満たさないことに応じて、前記コンテキストインデックスオフセットの前記値は 2 または 5 に等しい。

【 0 1 5 6 】

方法 9 0 0 のいくつかの実施例において、前記第 1 のコーディングモードが前記現在のブロックに適用されないことに応じて、前記コンテキストインデックスオフセットの前記値は 2 に等しく、前記第 1 のコーディングモードが前記現在のブロックに適用されることに応じて、前記コンテキストインデックスオフセットの前記値は 5 に等しい。方法 9 0 0 のいくつかの実施例において、前記現在のブロックに対して変換スキップ処理を適用する。方法 9 0 0 のいくつかの実施例において、前記規則は、 N_0 、 N_1 、 N_3 、および N_3 が、(1) シーケンスパラメータセット、映像パラメータセット、ピクチャパラメータセ

10

20

30

40

50

ット、ピクチャヘッダ、スライスヘッダ、タイル群ヘッダ、最大コーディングユニット行、最大コーディングユニット群、最大コーディングユニットまたはコーディングユニットに含まれる指示、(2)前記現在のブロックのブロック寸法および/または前記ブロックの隣接ブロックのブロック寸法、(3)前記ブロックのブロック形状および/または前記現在のブロックの隣接ブロックのブロック形状、(4)前記映像のカラーフォーマットの指示、(5)前記現在のブロックに別個または二重のコーディングツリー構造を適用するかどうか、(6)前記現在のブロックが属するスライスタイプおよび/またはピクチャタイプ、および(7)前記現在のブロックの色成分の数、のうちのいずれか1つ以上に基づいて決定されることを規定する。

【0157】

図10は、映像処理の方法1000の一例を示すフローチャートである。動作1002は、映像の現在のブロックと前記映像のビットストリームとの変換を行うことを含み、前記ビットストリームは、残りのコンテキストコーディングされたピンの数に基づいてコンテキストモードまたはバイパスモードのいずれか1つを用いて、前記現在のブロックの符号フラグが前記ビットストリームに含まれることを規定する規則に準拠する。

【0158】

方法1000のいくつかの実施例において、前記規則は、残りのコンテキストコーディングされたピンの前記数が、Nが整数である場合に、N未満であることに応じて、前記ビットストリームに前記符号フラグを含めるために前記バイパスモードを使用することを規定する。方法1000のいくつかの実施例において、前記規則は、残りのコンテキストコーディングされたピンの前記数が、Nが整数である場合に、N以上であることに応じて、前記符号フラグを前記ビットストリームに含めるために前記コンテキストモードを使用することを規定する。方法1000のいくつかの実施例において、前記規則は、残りのコンテキストコーディングされたピンの前記数が、Nが整数である場合に、Nに等しいことに応じて、前記ビットストリームに前記符号フラグを含めるために前記バイパスモードを使用することを規定する。方法1000のいくつかの実施例において、前記規則は、残りのコンテキストコーディングされたピンの前記数が、Nが整数である場合に、Nよりも大きいことに応じて、前記ビットストリームに前記符号フラグを含めるために前記バイパスモードを使用することを規定する。方法1000のいくつかの実施形態において、Nは4に等しい。方法1000のいくつかの実施形態において、Nは0に等しい。

【0159】

方法1000のいくつかの実施例において、Nは、(1)シーケンスパラメータセット、映像パラメータセット、ピクチャパラメータセット、ピクチャヘッダ、スライスヘッダ、タイル群ヘッダ、最大コーディングユニット行、最大コーディングユニット群、最大コーディングユニットまたはコーディングユニットに含まれる指示、(2)前記現在のブロックのブロック寸法および/または前記ブロックの隣接ブロックのブロック寸法、(3)前記ブロックのブロック形状および/または前記現在のブロックの隣接ブロックのブロック形状、(4)前記映像のカラーフォーマットの指示、(5)前記現在のブロックに別個または二重のコーディングツリー構造を適用するかどうか、(6)前記現在のブロックが属するスライスタイプおよび/またはピクチャタイプ、および(7)前記現在のブロックの色成分の数、のうちのいずれか1つ以上に基づいて決定されることを規定する。方法1000のいくつかの実施例において、前記規則は、前記現在のブロックが変換ブロックまたは変換スキップブロックであることを規定する。方法1000のいくつかの実施例において、前記規則は、ブロック差動パルスコーディングされた変調モードを前記現在のブロックに適用することを規定する。方法1000のいくつかの実施例において、前記規則は、ブロック差動パルスコーディングされた変調モードが前記現在のブロックに適用されないことを規定する。

【0160】

図11は、映像処理の方法1100の一例を示すフローチャートである。動作1102は、映像の現在のブロックと前記映像のビットストリームとの変換を行うことを含み、前

10

20

30

40

50

記ビットストリームは、前記ビットストリームに前記現在のブロックの符号フラグを含めるためにコンテキストインデックスオフセットを使用することを規定する規則に準拠し、前記規則は、前記コンテキストインデックスオフセットを前記現在のブロックの情報に基づいて決定することを規定する。

【0161】

方法1100のいくつかの実施例において、前記情報は、シーケンスパラメータセット、映像パラメータセット、ピクチャパラメータセット、ピクチャヘッダ、スライスヘッダ、タイル群ヘッダ、最大コーディングユニット行、最大コーディングユニット群、最大コーディングユニット、またはコーディングユニットにおける指示を含む。方法1100のいくつかの実施例において、前記情報は、前記現在のブロックのブロック寸法および/または前記現在のブロックの隣接ブロックのブロック寸法を含む。方法1100の一部の実施例において、前記情報は、前記現在のブロックのブロック形状および/または前記現在のブロックの隣接ブロックのブロック形状を含む。方法1100の一部の実施例において、前記符号フラグのコンテキスト値は、異なるブロック形状に対して別個である。方法1100のいくつかの実施例において、前記情報は、前記現在のブロックの1つ以上の隣接ブロックの1つ以上の予測モードを含む。方法1100のいくつかの実施例において、前記1つ以上の予測モードは、イントラ予測モードを含む。方法1100のいくつかの実施例において、前記1つ以上の予測モードは、インター予測モードを含む。

10

【0162】

方法1100のいくつかの実施例において、前記符号フラグのコンテキスト値は、1つ以上のインターコーディングされたブロックと、1つ以上のイントラコーディングされたブロックに対して別個である。方法1100のいくつかの実施例において、前記情報は、前記現在のブロックの隣接ブロックのブロックベースのデルタパルスコード変調モードの指示を含む。方法1100のいくつかの実施例において、前記情報は、前記映像のカラーフォーマットの指示を含む。方法1100のいくつかの実施例において、前記情報は、前記現在のブロックに対して別個または二重のコーディングツリー構造を適用するかを含む。方法1100のいくつかの実施例において、前記情報は、前記現在のブロックが属するスライスタイプおよび/またはピクチャタイプを含む。方法1100のいくつかの実施例において、前記情報は、前記現在のブロックの色成分の数を含む。方法1100のいくつかの実施例において、前記符号フラグのコンテキスト値は、前記現在のブロックの輝度色成分と前記現在のブロックの彩度色成分に対して別個である。方法(群)900~1100のいくつかの実施例において、前記現在のブロックは、バイナリデルタパルスコード変調モードを用いて前記ビットストリームにおいて表現される。

20

30

【0163】

図12は、映像処理の方法1200の一例を示すフローチャートである。動作1202は、規則に従って、1つ以上の映像レイヤを含む映像と前記映像のビットストリームとの変換を行うことを含み、前記規則は、前記映像に複数の適応パラメータセットネットワーク抽象化レイヤユニットを使用することを規定し、各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する適応パラメータタイプ値を有し、各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する映像レイヤ識別子に関連付けられ、各適応パラメータセットネットワーク抽象化レイヤユニットは、プレフィックスユニットまたはサフィックスユニットであり、前記規則は、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットが同じ適応パラメータタイプ値を共有することに応じて、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットの適応パラメータセット識別子値が同じ識別子空間に属することを規定する。

40

【0164】

方法1200のいくつかの実施例において、前記規則は、複数の適応パラメータセットネットワーク抽象化レイヤユニットが、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットの複数の映像レイヤ識別子とは独立した前記同じ適応パラメータタイプ値を共有することを規定する。方法1200のいくつかの実施例において、前記規則は

50

、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットがプレフィックスユニットであるかサフィックスユニットであるかに関わらず、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットが前記同じ適応パラメータタイプ値を共有することを規定する。方法 1 2 0 0 のいくつかの実施例において、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットは、複数の第 1 のモード適応パラメータセットネットワーク抽象化レイヤユニットであり、第 1 のモードにおいて、線形モデルに基づいてマッピングされた輝度予測サンプルを導出し、さらに輝度再構成サンプルを導出するために用いられ、輝度再構成サンプルに基づいて彩度残差値をスケールリングする。方法 1 2 0 0 のいくつかの実施例において、前記複数の適応パラメータセットネットワーク抽象化レイヤユニットは、複数の第 2 のモード適応パラメータセットネットワーク抽象化レイヤユニットであり、第 2 のモードにおいて、異なる方向の輝度再構成サンプル間の差に基づいてフィルタインデックスを生成する分類演算を用いて、輝度再構成サンプルをフィルタリングし、前記分類演算を行わずに、彩度再構成サンプルをフィルタリングする。

10

【 0 1 6 5 】

図 1 3 は、例示的な映像処理の方法 (1 3 0 0) のフローチャートである。動作 1 3 0 2 は、映像の現在のブロックと前記映像のビットストリームとの変換を行うことを含み、前記ビットストリームは、特定の特徴を有する第 1 の補足拡張情報メッセージが、(1) この特定の特徴を有する第 2 の補足拡張情報メッセージがスライスユニットに含まれていることに応じて、このビットストリームのスライスユニット内で繰り返されること、または (2) この第 1 の補足拡張メッセージがこのビットストリームのこのスライスユニット内にあることに応じて更新されること、は許可されないことを規定する規則に準拠し、このスライスユニットは、復号順に連続したネットワーク抽象化レイヤユニットのセットを含み、このネットワーク抽象化レイヤのセットは、単一のコーディングされたスライスと、この単一のコーディングされたスライスに関連付けられた 1 つ以上の非映像コーディングレイヤネットワーク抽象化レイヤユニットとを含む。

20

【 0 1 6 6 】

方法 1 3 0 0 のいくつかの実施例において、前記規則は、前記第 1 の補足拡張情報メッセージが前記スライスユニットに含まれる第 2 の補足拡張情報メッセージに対するものと同じ特定のペイロードタイプの値を有することに応じて、前記第 1 の補足拡張情報メッセージが前記スライスユニット内で繰り返されることは許可されないことを規定する。方法 1 3 0 0 のいくつかの実施例において、前記規則は、前記スライスユニット内の前記特定のペイロードタイプについての同一の構文構造の数が 1 以下であることを規定する。方法 1 3 0 0 のいくつかの実施例において、前記規則は、前記第 1 の補足拡張情報メッセージが特定のペイロードタイプを有することに応じて、前記第 1 の補足拡張情報メッセージが前記スライスユニット内で繰り返されることは許可されないことを規定する。方法 1 3 0 0 のいくつかの実施例において、前記規則は、前記スライスユニット内の前記特定のペイロードタイプの構文構造の数が 1 以下であることを規定する。

30

【 0 1 6 7 】

方法 1 3 0 0 のいくつかの実施例において、前記規則は、前記第 1 の補足拡張情報メッセージが特定のペイロードタイプを有することに応じて、前記第 1 の補足拡張情報メッセージが前記スライスユニット内で更新されることは許可されないことを規定する。方法 1 3 0 0 のいくつかの実施例において、前記規則は、前記スライスユニット内の前記特定のペイロードタイプの 1 つ以上の構文構造が同じ内容を有することを規定する。

40

【 0 1 6 8 】

図 1 4 は、映像処理の方法 1 4 0 0 の一例を示すフローチャートである。動作 1 4 0 2 は、映像の現在のブロックと前記映像のビットストリームとの変換を行うことを含み、前記ビットストリームは、前記ビットストリームにおけるスライスユニットが、(1) このスライスユニットは、第 1 の非映像コーディングレイヤネットワーク抽象化レイヤユニットと同じ特徴を有する第 2 の非映像コーディングレイヤネットワーク抽象化レイヤユニットを含むこと、および (2) この第 1 の非映像コーディングレイヤネットワーク抽象化レ

50

イヤユニットはプレフィックス補足拡張情報またはサフィックス補足拡張情報以外のネットワーク抽象化レイヤユニットタイプを有すること、に応じて、第1の非映像コーディングレイヤネットワーク抽象化レイヤユニットの繰り返しを許可しないことを規定する規則に準拠する。

【0169】

方法1400のいくつかの実施例において、前記第2の非映像コーディングレイヤネットワーク抽象化レイヤユニットは、復号能力情報ネットワーク抽象化レイヤユニットであり、前記規則は、前記スライスユニット内の復号能力情報ネットワーク抽象化レイヤユニットの数が1以下であることを規定する。方法1400のいくつかの実施例において、前記第2の非映像コーディングレイヤネットワーク抽象化レイヤユニットは、特定の識別子を有する映像パラメータセットネットワーク抽象化レイヤユニットであり、前記規則は、前記スライスユニット内の前記特定の識別子を有する映像パラメータセットネットワーク抽象化レイヤユニットの数が1以下であることを規定する。方法1400のいくつかの実施例において、前記第2の非映像コーディングレイヤネットワーク抽象化レイヤユニットは、特定の識別子を有するシーケンスパラメータセットネットワーク抽象化レイヤユニットであり、前記規則は、前記スライスユニット内の前記特定の識別子を有するシーケンスパラメータセットネットワーク抽象化レイヤユニットの数が1以下であることを規定する。方法1400のいくつかの実施例において、前記第2の非映像コーディングレイヤネットワーク抽象化レイヤユニットは、特定の識別子を有するピクチャパラメータセットネットワーク抽象化レイヤユニットであり、前記規則は、前記スライスユニット内の前記特定の識別子を有するピクチャパラメータセットネットワーク抽象化レイヤユニットの数は1を越えてはならないことを規定する。

10

20

【0170】

方法1400のいくつかの実施例において、前記第2の非映像コーディングレイヤネットワーク抽象化レイヤユニットは、特定の識別子および特定のパラメータタイプを有する適応パラメータセットネットワーク抽象化レイヤユニットであり、前記規則は、特定の識別子および前記スライスユニット内の前記特定のパラメータタイプを有する適応パラメータセットネットワーク抽象化レイヤユニットの数が1以下であることを規定する。方法1400のいくつかの実施例において、前記第2の非映像コーディングレイヤネットワーク抽象化レイヤユニットは、特定のタイプのネットワーク抽象化レイヤユニット、特定の識別子および特定のパラメータタイプを有する適応パラメータセットネットワーク抽象化レイヤユニットであり、前記規則は、前記特定のタイプのネットワーク抽象化レイヤユニット、前記特定の識別子、および前記スライスユニット内の前記特定のパラメータタイプを有する適応パラメータセットネットワーク抽象化レイヤユニットの数が1以下であることを規定する。方法1400のいくつかの実施例において、前記第2の非映像コーディングレイヤネットワーク抽象化レイヤユニットは、復号能力情報ネットワーク抽象化レイヤユニットであり、前記規則は、前記第1の非映像コーディングレイヤネットワーク抽象化レイヤユニットが前記復号能力情報ネットワーク抽象化レイヤユニットであることに応じて、コーディングされたレイヤ映像シーケンスまたはコーディングされた映像シーケンス内で前記第1の非映像コーディングレイヤネットワーク抽象化レイヤユニットが繰り返されることは許可されないことを規定する。

30

40

【0171】

方法1400のいくつかの実施例において、前記第2の非映像コーディングレイヤネットワーク抽象化レイヤユニットは、特定の識別子を有する映像パラメータセットネットワーク抽象化レイヤユニットであり、前記規則は、前記第1の非映像コーディングレイヤネットワーク抽象化レイヤユニットが特定の識別子を有する映像パラメータセットネットワーク抽象化レイヤユニットであることに応じて、コーディングされたレイヤ映像シーケンスまたはコーディングされた映像シーケンス内で前記第1の非映像コーディングレイヤネットワーク抽象化レイヤユニットが繰り返されることは許可されないことを規定する。方法1400のいくつかの実施例において、前記第2の非映像コーディングレイヤネットワ

50

ーク抽象化レイヤユニットは、特定の識別子を有するシーケンスパラメータセットネットワーク抽象化レイヤユニットであり、前記規則は、前記第1の非映像コーディングレイヤネットワーク抽象化レイヤユニットが特定の識別子を有するシーケンスパラメータセットネットワーク抽象化レイヤユニットであることに応じて、コーディングされたレイヤ映像シーケンスまたはコーディングされた映像シーケンス内で前記第1の非映像コーディングレイヤネットワーク抽象化レイヤユニットが繰り返されることは許可されないことを規定する。

【0172】

図15は、映像処理の方法1500の一例を示すフローチャートである。動作1502は、規則に従って、複数の映像レイヤを含む映像と前記映像のビットストリームとの変換を行うことを含み、前記規則は、複数の適応パラメータセットのうちのどれが前記複数の映像レイヤにわたって共有されることは許可されないかを規定する。

10

【0173】

方法1500のいくつかの実施例において、前記規則は、前記複数の適応パラメータセットの全てが共有されることは許可されないことを規定する。方法1500のいくつかの実施例において、前記規則は、前記複数の適応パラメータセットの適応ループフィルタタイプが共有されることは許可されないことを規定する。

【0174】

図16は、映像処理の方法1600の一例を示すフローチャートである。動作1602は、規則に従って、1つ以上の映像レイヤを含む映像と前記映像のビットストリームとの変換を行うことを含み、前記ビットストリームは、1つ以上の適応ループフィルタ適応パラメータセットを含み、この規則は、この1つ以上の適応ループフィルタ適応パラメータセットをピクチャユニット内で更新するように許可されるかどうかを規定する。

20

【0175】

方法1600のいくつかの実施例において、前記規則は、前記1つ以上の適応ループフィルタ適応パラメータセットを前記ピクチャユニット内で更新するように許可されることを規定する。方法1600のいくつかの実施例において、前記規則は、適応パラメータセットを前記ピクチャユニット内で更新するように許可されないことを規定する。

【0176】

図17は、映像処理の方法1700の一例を示すフローチャートである。動作1702は、規則に従って、1つ以上のコーディングされたレイヤ映像シーケンスを含む映像と前記映像のビットストリームとの変換を行うことを含み、前記ビットストリームは適応ループフィルタ適応パラメータセットを含み、前記規則は、前記適応ループフィルタ適応パラメータセットが1つ以上の特定の特徴を有することに応じて、前記適応ループフィルタ適応パラメータセットが前記1つ以上のコーディングされたレイヤ映像シーケンスにわたって共有されることは許可されないことを規定する。

30

【0177】

方法1700のいくつかの実施例において、前記特定の特徴は、特定のネットワーク抽象化レイヤユニットタイプを有する前記適応ループフィルタ適応パラメータセットを含む。方法1700のいくつかの実施例において、前記1つ以上の特定の特徴は、特定の識別子を有する前記適応ループフィルタ適応パラメータセットを含む。方法1700のいくつかの実施例において、前記1つ以上の特定の特徴は、特定のパラメータタイプを有する前記適応ループフィルタ適応パラメータセットを含む。方法1700のいくつかの実施例において、前記規則は、第1のコーディングされたレイヤ映像シーケンスにおける映像コーディングレイヤネットワーク抽象化レイヤユニットによって参照される前記適応ループフィルタ適応パラメータセットが、第2のコーディングされたレイヤ映像シーケンスにおける非映像コーディングレイヤネットワーク抽象化レイヤユニットに関連付けられず、前記第1のコーディングされたレイヤ映像シーケンスが前記第2のコーディングされたレイヤ映像シーケンスと異なることを規定する。方法1700のいくつかの実施例において、前記規則は、第1のイントラランダムアクセスピクチャに関連付けられた映像コーディング

40

50

レイヤネットワーク抽象化レイヤユニットによって参照される前記適応ループフィルタ適応パラメータセットが、第2のイントラランダムアクセスピクチャに関連付けられた非映像コーディングレイヤネットワーク抽象化レイヤユニットに関連付けられず、前記第1のイントラランダムアクセスピクチャが前記第2のイントラランダムアクセスピクチャと異なることを規定する。

【0178】

方法1700のいくつかの実施例において、前記規則は、第1のイントラランダムアクセスピクチャまたは第1の漸次復号更新ピクチャに関連付けられた映像コーディングレイヤネットワーク抽象化レイヤユニットによって参照される前記適応ループフィルタ適応パラメータセットが、第2のイントラランダムアクセスピクチャまたは第2の漸次復号更新ピクチャに関連する非映像コーディングレイヤネットワーク抽象化レイヤユニットに関連付けられず、前記第1のイントラランダムアクセスピクチャまたは前記第1の漸次復号更新ピクチャが前記第2のイントラランダムアクセスピクチャまたは前記第2の漸次復号更新ピクチャと異なることを規定する。方法1700のいくつかの実施例において、前記規則は、前記適応ループフィルタ適応パラメータセットが、前記映像の1つ以上のコーディングされた映像シーケンスにわたって共有されない前記映像の1つ以上の適応パラメータセットに含まれることを規定する。方法1700のいくつかの実施例において、前記規則は、特定のタイプに属する前記1つ以上の適応パラメータセットが、前記1つ以上のコーディングされた映像シーケンスにわたって共有されることは許可されないことを規定する。方法1700のいくつかの実施例において、前記特定のタイプは、適応ループフィルタ適応パラメータセットを含む。方法1700のいくつかの実施例において、前記特定のタイプは、彩度スケール適応パラメータセットを用いた輝度マッピングを含む。方法1700のいくつかの実施例において、前記特定のタイプは、スケール適応パラメータセットを含む。

【0179】

方法900~1700のいくつかの実施形態において、変換を実行することは、映像をビットストリームに符号化することを含む。方法900~1700のいくつかの実施形態において、変換を実行することは、映像からビットストリームを生成することを含み、方法は、ビットストリームを非一時的なコンピュータ可読記録媒体に記憶することをさらに含む。方法900~1700のいくつかの実施形態において、変換を実行することは、ビットストリームから映像を復号することを含む。いくつかの実施形態において、映像復号装置は、方法900~1700の動作を実装するように構成された処理装置を含む。

【0180】

いくつかの実施形態において、映像符号化装置は、方法900~1700の動作を実装するように構成された処理装置を含む。いくつかの実施形態において、コンピュータ命令が記憶されたコンピュータプログラム製品は、処理装置により実行されることにより、処理装置に方法900~1700に説明された動作を実装させる。いくつかの実施形態において、方法900~1700に説明された動作に従って生成されたビットストリームを記憶する非一時的なコンピュータ可読記憶媒体。いくつかの実施形態において、処理装置に方法900~1700に説明された動作を実装させる命令を記憶する非一時的なコンピュータ可読記憶媒体。いくつかの実施形態において、方法(s)900~1700に説明された動作に従って映像のビットストリームを生成し、このビットストリームをコンピュータ可読プログラム媒体に記憶するステップを含む、ビットストリーム生成方法。いくつかの実施形態において、本願明細書に開示される方法またはシステムに従って、方法、装置、生成されたビットストリームを提供する。

【0181】

本明細書では、「映像処理」という用語は、映像符号化、映像復号、映像圧縮、または映像展開を指してよい。例えば、映像圧縮アルゴリズムは、映像の画素表現から対応するビットストリーム表現への変換、またはその逆の変換中に適用されてもよい。現在の映像ブロックのビットストリーム表現は、例えば、構文によって規定されるように、ビットス

10

20

30

40

50

トリーム内の同じ場所または異なる場所に拡散されるビットに対応していてもよい。例えば、1つのマクロブロックは、変換およびコーディングされた誤り残差値の観点から、かつビットストリームにおけるヘッダおよび他のフィールドにおけるビットを使用して符号化されてもよい。さらに、変換中、デコーダは、上記解決策で説明されているように、判定に基づいて、いくつかのフィールドが存在しても存在しなくてもよいという知識を持って、ビットストリームを構文解析してもよい。同様に、エンコーダは、特定の構文フィールドが含まれるべきであるか、または含まれないべきであるかを判定し、構文フィールドをコーディング表現に含めるか、またはコーディング表現から除外することによって、それに応じてコーディング表現を生成してもよい。

【0182】

本明細書に記載された開示された、およびその他の解決策、実施例、実施形態、モジュール、および機能動作の実装形態は、本明細書に開示された構造およびその構造的等価物を含め、デジタル電子回路、またはコンピュータソフトウェア、ファームウェア、若しくはハードウェアで実施されてもよく、またはそれらの1つ以上の組み合わせで実施してもよい。開示された、およびその他の実施形態は、1または複数のコンピュータプログラム製品、すなわち、データ処理装置によって実装されるため、またはデータ処理装置の動作を制御するために、コンピュータ可読媒体上に符号化されたコンピュータプログラム命令の1または複数のモジュールとして実施することができる。このコンピュータ可読媒体は、機械可読記憶デバイス、機械可読記憶基板、メモリデバイス、機械可読伝播信号をもたらず物質の組成物、またはこれらの1または複数の組み合わせであってもよい。「データ処理装置」という用語は、例えば、プログラマブルプロセッサ、コンピュータ、または複数のプロセッサ、若しくはコンピュータを含む、データを処理するためのすべての装置、デバイス、および機械を含む。この装置は、ハードウェアの他に、当該コンピュータプログラムの実行環境を作るコード、例えば、プロセッサファームウェア、プロトコルスタック、データベース管理システム、オペレーティングシステム、またはこれらの1または複数の組み合わせを構成するコードを含むことができる。伝播信号は、人工的に生成した信号、例えば、機械で生成した電気、光、または電磁信号であり、適切な受信装置に送信するための情報を符号化するために生成される。

【0183】

コンピュータプログラム（プログラム、ソフトウェア、ソフトウェアアプリケーション、スクリプト、またはコードとも呼ばれる）は、コンパイルされた言語または解釈された言語を含む任意の形式のプログラミング言語で記述することができ、また、それは、スタンドアロンプログラムとして、またはコンピューティング環境で使用するのに適したモジュール、成分、サブルーチン、または他のユニットとして含む任意の形式で展開することができる。コンピュータプログラムは、必ずしもファイルシステムにおけるファイルに対応するとは限らない。プログラムは、他のプログラムまたはデータを保持するファイルの一部（例えば、マークアップ言語文書に格納された1または複数のスクリプト）に記録されていてもよいし、当該プログラム専用の単一のファイルに記憶されていてもよいし、複数の調整ファイル（例えば、1または複数のモジュール、サブプログラム、またはコードの一部を格納するファイル）に記憶されていてもよい。1つのコンピュータプログラムを、1つのサイトに位置する1つのコンピュータ、または複数のサイトに分散され通信ネットワークによって相互接続される複数のコンピュータで実行させるように展開することも可能である。

【0184】

本明細書に記載された処理およびロジックフローは、入力データ上で動作し、出力を生成することによって機能を実行するための1つ以上のコンピュータプログラムを実行する1つ以上のプログラマブル処理装置によって行うことができる。処理およびロジックフローはまた、特定用途のロジック回路、例えば、FPGA（フィールドプログラマブルゲートアレイ）またはASIC（特定用途向け集積回路）によって行うことができ、装置はまた、特別目的のロジック回路として実装することができる。

10

20

30

40

50

【 0 1 8 5 】

コンピュータプログラムの実行に適した処理装置は、例えば、汎用および専用マイクロ処理装置の両方、並びに任意の種類デジタルコンピュータの任意の1または複数の処理装置を含む。一般的に、プロセッサは、リードオンリーメモリまたはランダムアクセスメモリまたはその両方から命令およびデータを受信する。コンピュータの本質的な要素は、命令を実行するためのプロセッサと、命令およびデータを記憶するための1つ以上の記憶装置とである。一般的に、コンピュータは、データを記憶するための1または複数の大容量記憶デバイス、例えば、磁気、光磁気ディスク、または光ディスクを含んでもよく、またはこれらの大容量記憶デバイスからデータを受信するか、またはこれらにデータを転送するように動作可能に結合されてもよい。しかしながら、コンピュータは、このようなデバイス

を有する必要はない。コンピュータプログラム命令およびデータを記憶するのに適したコンピュータ可読媒体は、あらゆる形式の不揮発性メモリ、媒体、およびメモリデバイスを含み、例えば、EPROM、EEPROM、フラッシュ記憶装置、磁気ディスク、例えば内部ハードディスクまたはリムーバブルディスク、光磁気ディスク、およびCD-ROMおよびDVD-ROMディスク等の半導体記憶装置を含む。処理装置およびメモリは、特定用途のロジック回路によって補完されてもよく、または特定用途のロジック回路に組み込まれてもよい。

10

【 0 1 8 6 】

本特許明細書は多くの詳細を含むが、これらは、任意の主題の範囲または特許請求の範囲を限定するものと解釈されるべきではなく、むしろ、特定の技術の特定の実施形態に特有であり得る特徴の説明と解釈されるべきである。本特許文献において別個の実施形態のコンテキストで説明されている特定の機能は、1つの例において組み合わせて実装してもよい。逆に、1つの例のコンテキストで説明された様々な機能は、複数の実施形態において別個にまたは任意の適切なサブコンビネーションで実装してもよい。さらに、特徴は、特定の組み合わせで作用するものとして上記に記載され、最初にそのように主張されていてもよいが、主張された組み合わせからの1つ以上の機能は、場合によっては、組み合わせから抜粋されることができ、主張された組み合わせは、サブコンビネーションまたはサブコンビネーションのバリエーションに向けられてもよい。

20

【 0 1 8 7 】

同様に、動作は図面において特定の順番で示されているが、これは、所望の結果を達成するために、このような動作が示された特定の順番でまたは連続した順番で行われること、または示された全ての動作が行われることを必要とするものと理解されるべきではない。また、本特許明細書に記載されている例における様々なシステムの構成要素の分離は、全ての実施形態においてこのような分離を必要とするものと理解されるべきではない。

30

【 0 1 8 8 】

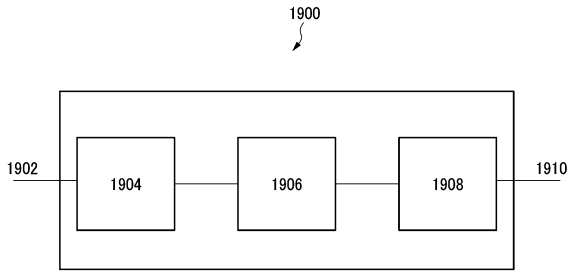
いくつかの実装形態および例のみが記載されており、この特許文献に記載され図示されているコンテンツに基づいて、他の実施形態、拡張および変形が可能である。

40

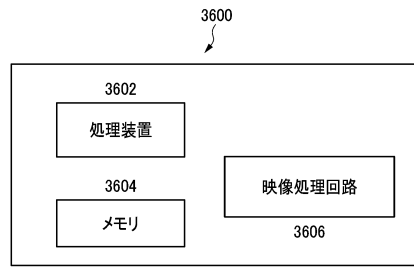
50

【図面】

【図 1】

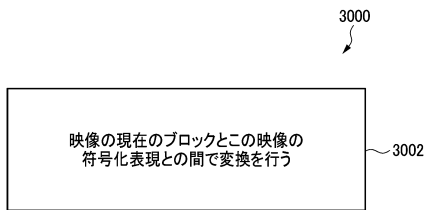


【図 2】

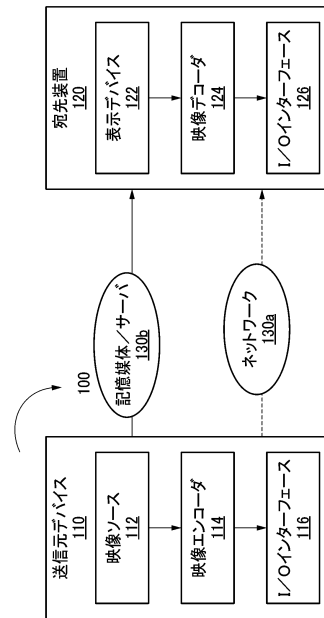


10

【図 3】



【図 4】



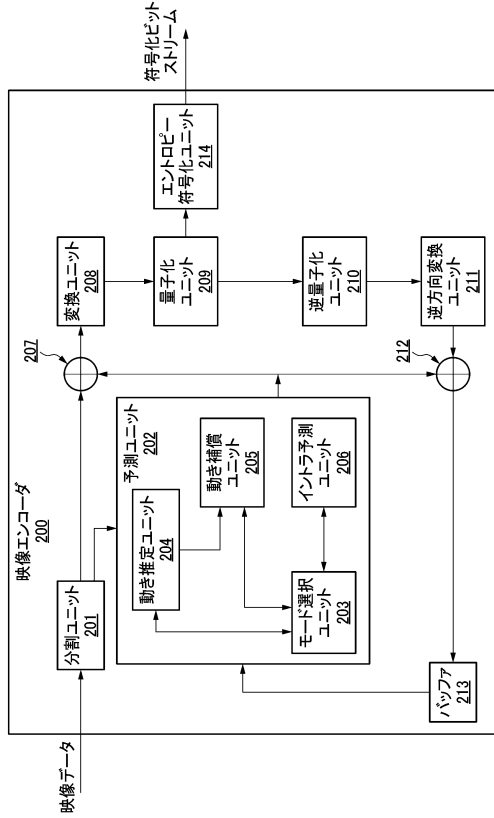
20

30

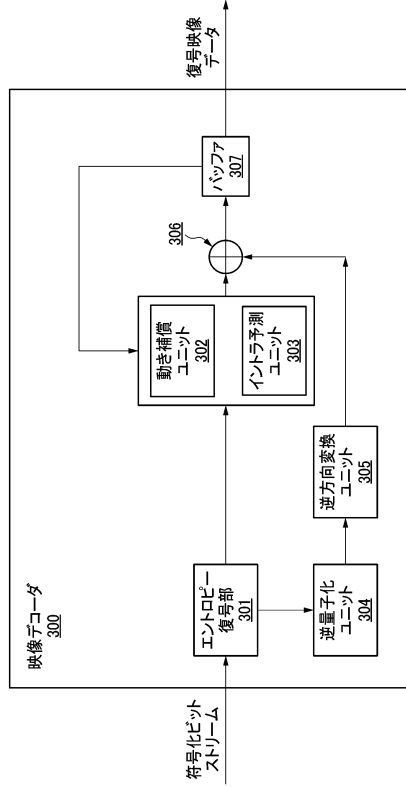
40

50

【 図 5 】



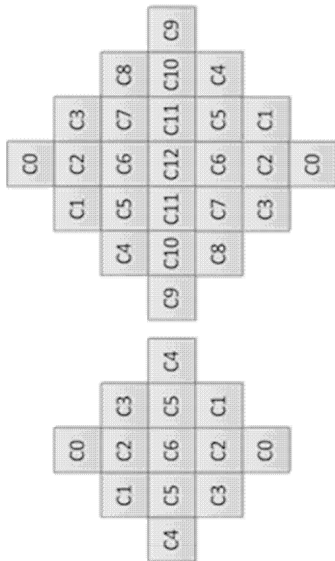
【 図 6 】



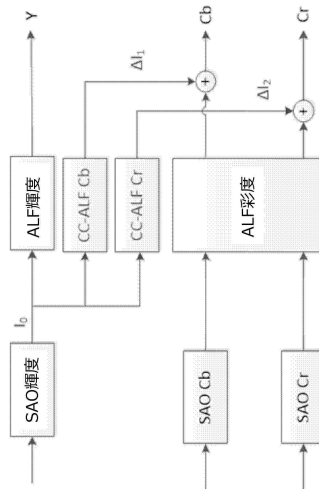
10

20

【 図 7 】



【 図 8 】



30

40

50

【 図 9 】

900

映像の現在のブロックと映像のビットストリームとの変換を行うことであって、このビットストリームは、ビットストリームにおいて第1の係数の第1の符号フラグを含めるためにコンテキストインデックスオフセットを使用することを規定する規則に準拠し、この規則は、コンテキストインデックスオフセットの値が、ビットストリームにおける現在のブロックに第1のコーディングモードが適用されるかどうかに基づきことを規定する

902

【 図 10 】

1000

映像の現在のブロックとこの映像のビットストリームとの変換を行うことであって、このビットストリームは、残りのコンテキストコーディングされたピンの数に基づいて、コンテキストモードおよびバイパスモードのうちいずれか1つを使用して、現在のブロックの符号フラグがビットストリームに含まれることを規定する規則に準拠する

1002

10

【 図 11 】

1100

映像の現在のブロックとこの映像のビットストリームとの変換を行うことであって、このビットストリームは、ビットストリームに現在のブロックの符号フラグを含めるためにコンテキストインデックスオフセットを使用することを規定する規則に準拠し、この規則は、コンテキストインデックスオフセットを現在のブロックの情報に基づいて決定することを規定する

1102

【 図 12 】

1200

規則に従って、1つ以上の映像レイヤを含む映像とこの映像のビットストリームとの変換を行うことであって、この規則は、この映像のために複数の適応パラメータセットネットワーク抽象化レイヤユニットを使用することを規定し、各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する適応パラメータタイプ値を有し、各適応パラメータセットネットワーク抽象化レイヤユニットは、対応する映像レイヤ識別子に関連付けられ、各適応パラメータセットネットワーク抽象化レイヤユニットは、プレフィックスユニットまたはサフィックスユニットであり、この規則は、複数の適応パラメータセットネットワーク抽象化レイヤユニットが同じ適応パラメータタイプ値を共有することに対応して、複数の適応パラメータセットネットワーク抽象化レイヤユニットの適応パラメータセット識別子値が同じ識別子空間に属することを規定する

1202

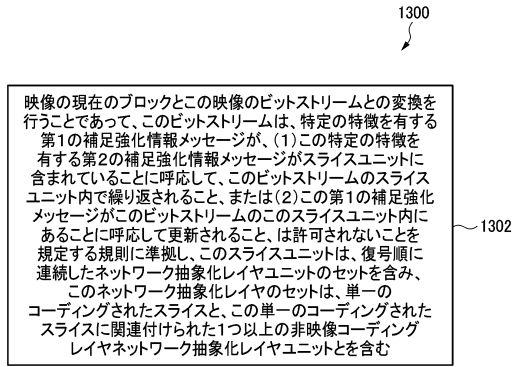
20

30

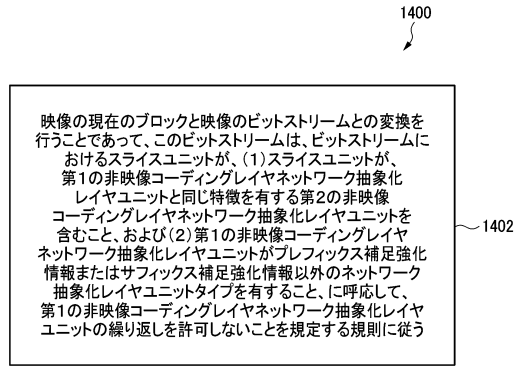
40

50

【 図 1 3 】

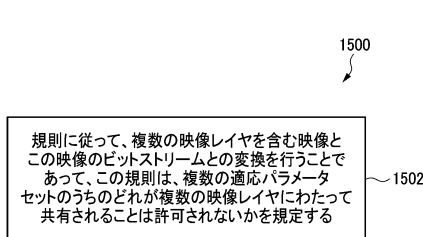


【 図 1 4 】

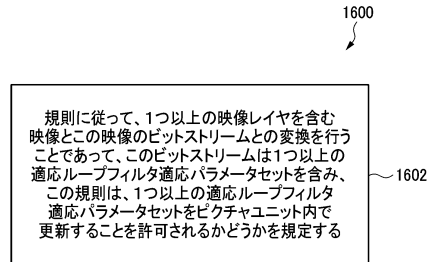


10

【 図 1 5 】



【 図 1 6 】



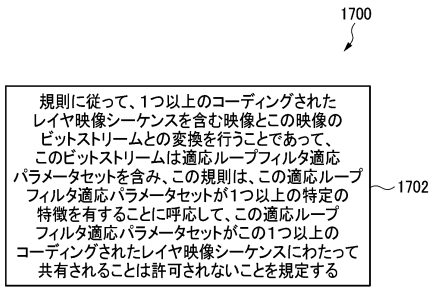
20

30

40

50

【 図 17 】



10

20

30

40

50

フロントページの続き

BYTEDANCE INC .

アメリカ合衆国 カリフォルニア州 90066, ロサンゼルス, ウェスト ジェファーソン ブールヴァード 12655, シックス フロア, 스위트 ナンバー・137

12655 West Jefferson Boulevard, Sixth Floor, Suite No. 137 Los Angeles, California 90066 United States of America

(74)代理人 110002000

弁理士法人栄光事務所

(72)発明者 ワン イェクイ

アメリカ合衆国 90066 カリフォルニア州, ロサンゼルス, ウェスト ジェファーソン ブールバード 12655, シックス フロア, 스위트 ナンバー137

(72)発明者 ジュー ウェイジャ

アメリカ合衆国 90066 カリフォルニア州, ロサンゼルス, ウェスト ジェファーソン ブールバード 12655, シックス フロア, 스위트 ナンバー137

(72)発明者 ジャン リー

アメリカ合衆国 90066 カリフォルニア州, ロサンゼルス, ウェスト ジェファーソン ブールバード 12655, シックス フロア, 스위트 ナンバー137

(72)発明者 ドン ジピン

中華人民共和国 100080 ベイジン, ハイディアン ディストリクト, チーチュン ロード, チャイナ サテライト コミュニケーションズ タワー, ナンバー63, ジンリトウティアオ ポストオフィス

審査官 久保 光宏

(56)参考文献

Benjamin Bross, et al., "Versatile Video Coding (Draft 8)", Document: JVET-Q2001-v9, [online], JVET-Q2001 (version 9), Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 2020年01月22日, Pages 13,115,155,169,420, [令和6年4月1日検索], インターネット, URL: https://jvet-experts.org/doc_end_user/current_document.php?id=9675 and URL: https://jvet-experts.org/doc_end_user/documents/17_Brussels/wg11/JVET-Q2001-v9.zip ., (See document file "JVET-Q2001-v9.docx" in the zip file "JVET-Q2001-v9.zip".)

Cheung Auyeung, et al., "CE7-related: Modifications to transform skip coefficient sign flag coding", Document: JVET-P0437_r1, [online], JVET-P0437 (version 2), Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 2019年10月04日, Pages 1-9, [令和6年4月1日検索], インターネット, URL: https://jvet-experts.org/doc_end_user/current_document.php?id=8226 and URL: https://jvet-experts.org/doc_end_user/documents/16_Geneva/wg11/JVET-P0437-v2.zip ., (See document file "JVET-P0437_r1.docx" in the zip file "JVET-P0437-v2.zip".)

Benjamin Bross, et al., "Versatile Video Coding (Draft 7)", Document: JVET-P2001-v3, [online], JVET-P2001 (version 3), 2019年10月13日, Pages 13,138,153,394, [令和6年4月1日検索], インターネット, URL: https://jvet-experts.org/doc_end_user/current_document.php?id=8857 and URL: https://jvet-experts.org/doc_end_user/documents/16_Geneva/wg11/JVET-P2001-v3.zip ., (See document file "JVET-P2001-v3.docx" in the zip file "JVET-P2001-v3.zip".)

Ye-Kui Wang, et al., "AHG9: Cleanups on parameter sets", Document: JVET-Q0117-v1, [online], JVET-Q0117 (version 1), Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 2019年12月30日, Pages 1-9, [令和6年9月9日検索], インターネット, URL: https://jvet-experts.org/doc_end_user/current_document.php?id=8942 and URL: https://jvet-experts.org/doc_end_user/documents/17_Brussels/wg11/JVET-Q0117-v1.zip ., (See document file "JVET-Q0117-v1.docx" in the zip file "JVET-Q0117-v1.zip".)

Ye-Kui Wang, et al. , "AHG9: On repetition and update of non-VCL data units" , Document: JVET-R0070-v1, [online] , JVET-R0070 (version 1) , Joint Video Experts Team (JVET) of IT U-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 , 2020年04月02日 , Pages 1-4 , [令和6年9月9日検索], インターネット, URL: https://jvet-experts.org/doc_end_user/current_document.php?id=9714 and URL: https://jvet-experts.org/doc_end_user/documents/18_Alpbach/wg11/JVET-R0070-v1.zip . , (See document file "JVET-R0070-v1.docx" in the zip file "JVET-R0070-v1.zip".)

(58)調査した分野 (Int.Cl. , D B 名)

H 0 4 N 1 9 / 0 0 - 1 9 / 9 8

C S D B (日本国特許庁)

学術文献等データベース (日本国特許庁)

I E E E X p l o r e (I E E E)