

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7631455号
(P7631455)

(45)発行日 令和7年2月18日(2025.2.18)

(24)登録日 令和7年2月7日(2025.2.7)

(51)国際特許分類	F I
H 0 4 N 19/126 (2014.01)	H 0 4 N 19/126
H 0 4 N 19/157 (2014.01)	H 0 4 N 19/157
H 0 4 N 19/70 (2014.01)	H 0 4 N 19/70

請求項の数 3 (全80頁)

(21)出願番号	特願2023-146837(P2023-146837)	(73)特許権者	502032105
(22)出願日	令和5年9月11日(2023.9.11)		エルジー エレクトロニクス インコーポ レイティド
(62)分割の表示	特願2022-520214(P2022-520214)の分割		L G E L E C T R O N I C S I N C .
原出願日	令和2年10月5日(2020.10.5)		大韓民国,ソウル,ヨンドゥンポ-ク, ヨイ-デロ,128
(65)公開番号	特開2023-158204(P2023-158204 A)		128, Yeoui-daero, Y eongdeungpo-gu, 07 336 Seoul, Republic of Korea
(43)公開日	令和5年10月26日(2023.10.26)	(74)代理人	100099759
審査請求日	令和5年9月11日(2023.9.11)		弁理士 青木 篤
(31)優先権主張番号	62/911,221	(74)代理人	100123582
(32)優先日	令和1年10月5日(2019.10.5)		弁理士 三橋 真二
(33)優先権主張国・地域又は機関	米国(US)	(74)代理人	100165191

最終頁に続く

(54)【発明の名称】 変換スキップ及びパレットコーディング関連情報のシグナリングベースの映像又はビデオコーディング

(57)【特許請求の範囲】

【請求項1】

デコーディング装置により行われる映像デコーディング方法において、
ビットストリームから予測モード情報、レジデュアル情報及び変換スキップ可用情報を含む映像情報を取得するステップと、
前記変換スキップ可用情報の値に基づいて変換スキップフラグを取得するステップと、
前記予測モード情報に基づいて現在ブロックに対する予測サンプルを導出するステップと、
前記変換スキップフラグ及び前記レジデュアル情報に基づいて前記現在ブロックに対するレジデュアルサンプルを導出するステップと、
前記予測サンプル及び前記レジデュアルサンプルに基づいて復元サンプルを生成するステップと、を含み、
前記映像情報は、パレットコーディング可用情報を含み、
前記変換スキップ可用情報及び前記パレットコーディング可用情報のうち少なくとも一つに基づいて、前記映像情報は、変換スキップモードに対する最小許容量子化パラメータに関連する最小量子化パラメータ情報をさらに含み、
前記変換スキップフラグの値が1に等しいことは、変換が前記現在ブロックに適用されないことを表す前記変換スキップモードに関連し、
前記変換スキップフラグの前記値が1に等しいことに基づき、
前記最小量子化パラメータ情報に基づいて前記現在ブロックに対する量子化パラメータ

が導出され、前記量子化パラメータに基づいて前記レジデュアルサンプルが導出される、映像デコーディング方法。

【請求項 2】

エンコーディング装置により行われる映像エンコーディング方法において、
 現在ブロックに対する予測モードを導出するステップと、
 前記予測モードに関する情報を生成するステップと、
 変換スキップ可用情報に基づいて変換スキップフラグを生成するステップと、
前記変換スキップフラグに基づいてレジデュアル情報を生成するステップと、
 前記予測モードに関する前記情報、前記変換スキップフラグ、前記変換スキップ可用情報及び前記レジデュアル情報を含む映像情報をエンコーディングするステップと、
 前記映像情報は、パレットコーディング可用情報を含み、
 前記変換スキップ可用情報及び前記パレットコーディング可用情報のうちの少なくとも1つに基づいて、前記映像情報は、変換スキップモードに対する最小許容量子化パラメータに関連する最小量子化パラメータ情報をさらに含み、
前記変換スキップフラグの値が1に等しいことは、変換が前記現在ブロックに適用されないことを表す前記変換スキップモードに関連し、
 前記変換スキップフラグの前記値が1に等しいことに基づいて、前記レジデュアル情報は、前記変換が適用されない前記現在ブロックに対して生成される、映像エンコーディング方法。

10

【請求項 3】

映像に対するデータの送信方法であって、
 ビットストリームを取得するステップであって、前記ビットストリームは、
 現在ブロックに対する予測モードを導出するステップと、
 前記予測モードに関する情報を生成するステップと、
 変換スキップ可用情報に基づいて変換スキップフラグを生成するステップと、
前記変換スキップフラグに基づいてレジデュアル情報を生成するステップと、
 前記予測モードに関する前記情報、前記変換スキップフラグ、前記変換スキップ可用情報及び前記レジデュアル情報を含む映像情報をエンコーディングするステップと、
に基づいて生成される、ステップと、
 前記ビットストリームを含む前記データを送信するステップと、
 前記映像情報は、パレットコーディング可用情報を含み、
 前記変換スキップ可用情報及び前記パレットコーディング可用情報のうち少なくとも1つに基づいて、前記映像情報は、変換スキップモードに対する最小許容量子化パラメータに関連する最小量子化パラメータ情報をさらに含み、
前記変換スキップフラグの値が1に等しいことは、変換が前記現在ブロックに適用されないことを表す前記変換スキップモードに関連し、
 前記変換スキップフラグの前記値が1に等しいことに基づいて、前記レジデュアル情報は、前記変換が適用されない前記現在ブロックに対して生成される、送信方法。

20

30

【発明の詳細な説明】

【技術分野】

40

【0001】

本技術は、ビデオ又は映像コーディングに関し、例えば、変換スキップ及びパレットコーディング関連情報のシグナリングベースの映像又はビデオコーディング技術に関する。

【背景技術】

【0002】

近年、4Kまたは8K以上のUHD(Ultra High Definition)画像/ビデオのような高解像度、高品質の画像/ビデオに対する需要が様々な分野で増加している。画像/ビデオデータが高解像度、高品質になるほど、既存の画像/ビデオデータに比べて相対的に送信される情報量またはビット量が増加するので、既存の有無線広帯域回線のような媒体を利用して画像データを送信するか、既存の格納媒体を利用して画像/

50

ビデオデータを格納する場合、送信費用と格納費用が増加される。

【0003】

また、近年、VR (Virtual Reality)、AR (Artificial Reality) コンテンツやホログラムなどの実感メディア (Immersive Media) に対する関心及び需要が増加しており、画像のように、現実画像と異なる画像特性を有する画像/ビデオに対する放送が増加している。

【0004】

これにより、前記のような様々な特性を有する高解像度・高品質の画像/ビデオの情報を効果的に圧縮して送信するか、格納し、再生するために高効率の画像/ビデオ圧縮技術が求められる。

10

【0005】

また、変換スキップ及びパレットコーディングの実行において、必須であるか又は補助的に使われる情報の従属性及び非終属性に応じて関連情報のコーディングを行うか否かを効率的に区分することにより全般的な映像/ビデオコーディングの効率を向上させる方案が必要である。

【発明の概要】

【発明が解決しようとする課題】

【0006】

本文書の技術的課題はビデオ/映像コーディング効率を高める方法及び装置を提供することにある。

20

【0007】

本文書の他の技術的課題は、変換スキップ及び/又はパレットコーディング関連情報を効率的にパーシング/シグナリングする方法及び装置を提供することにある。

【0008】

本文書のまた他の技術的課題は、変換スキップ及び/又はパレットコーディング時に使用される情報の従属性及び/又は非終属性に応じて効率的にコーディングの可否を決定する方法及び装置を提供することにある。

【0009】

本文書のまた他の技術的課題は、変換スキップ及び/又はパレットコーディングに関するハイレベルシンタックス要素 (high level syntax element) に対して従属性のあるシンタックス要素 (syntax element) を効果的にパーシングするための従属条件を定義し、前記従属条件に基づいてパーシングを行うか否かを決定する方法及び装置を提供することにある。

30

【課題を解決するための手段】

【0010】

本文書の実施例によると、SPS (Sequence Parameter Set) を介して変換スキップ可用情報及びパレット可用情報をシグナリングし、前記変換スキップ可用情報及び前記パレット可用情報のうち少なくとも1つに基づいて変換スキップモードに対する最小許容量子化パラメータ (minimum allowed quantization parameter) に関する最小量子化パラメータ情報をパーシング/シグナリングするか否かを決定することができる。例えば、前記変換スキップ可用情報の値が1又は前記パレットコーディング可用情報の値が1である条件に基づいて前記SPSにおいて前記最小量子化パラメータ情報をパーシング/シグナリングすることができる。

40

【0011】

本文書の一実施例によると、デコーディング装置により行われるビデオ/映像デコーディング方法を提供する。前記ビデオ/映像デコーディング方法は、本文書の実施例において開示された方法を含む。

【0012】

本文書の一実施例によると、ビデオ/映像デコーディングを行うデコーディング装置を提供する。前記デコーディング装置は、本文書の実施例において開示された方法を行う。

50

【 0 0 1 3 】

本文書の一実施例によると、エンコーディング装置により行われるビデオ/映像エンコーディング方法を提供する。前記ビデオ/映像エンコーディング方法は、本文書の実施例において開示された方法を含む。

【 0 0 1 4 】

本文書の一実施例によると、ビデオ/映像エンコーディングを行うエンコーディング装置を提供する。前記エンコーディング装置は、本文書の実施例において開示された方法を行う。

【 0 0 1 5 】

本文書の一実施例によると、本文書の実施例のうち少なくとも一つに開示されたビデオ/映像エンコーディング方法によって生成されたエンコーディングされたビデオ/映像情報が格納されたコンピュータ読み取り可能なデジタル格納媒体を提供する。

10

【 0 0 1 6 】

本文書の一実施例によると、デコーディング装置により本文書の実施例のうち少なくとも一つに開示されたビデオ/映像デコーディング方法を実行するようにするエンコーディングされた情報またはエンコーディングされたビデオ/映像情報が格納されたコンピュータ読み取り可能なデジタル格納媒体を提供する。

【 発明の効果 】

【 0 0 1 7 】

本文書は様々な効果を有することができる。例えば、本文書の一実施例によると、一般的な映像/ビデオ圧縮効率を高めることができる。また、本文書の一実施例によると、変換スキップ及び/又はパレットコーディング関連情報を効率的にパーシング/シグナリングすることができる。また、本文書の一実施例によると、変換スキップ及び/又はパレットコーディング時に使用される情報の従属性及び/又は非終属性に応じてコーディングするか否かを効果的に決定することができる。また、本文書の一実施例によると、変換スキップ及び/又はパレットコーディング関連ハイレベルシンタックス要素 (`high level syntax element`) に対して従属性のあるシンタックス要素 (`syntax element`) を効果的にパーシングするための従属条件を定義し、前記従属条件に応じてパーシングするか否かを決定することにより効率的なコーディングが可能である。また、本文書の一実施例によると、変換スキップ及び/又はパレットコーディング関連ハイレベルシンタックス要素 (`high level syntax element`) に対して従属条件に応じてパーシングするか否かを決定させることにより、送信されるビットを節約する効果が得られる。

20

30

【 0 0 1 8 】

本文書の具体的な一例を介して得ることができる効果は、以上で羅列された効果に制限されない。例えば、関連した技術分野の通常の知識を有する者が (`a person having ordinary skill in the related art`) が本文書から理解し、または誘導できる多様な技術的效果が存在できる。それによって、本文書の具体的な効果は、本文書に明示的に記載されたものに制限されずに、本文書の技術的特徴から理解され、または誘導されることができる多様な効果を含むことができる。

40

【 図面の簡単な説明 】

【 0 0 1 9 】

【 図 1 】 本文書の実施例に適用できるビデオ/映像コーディングシステムの例を概略的に示す。

【 図 2 】 本文書の実施例が適用できるビデオ/映像エンコーディング装置の構成を概略的に説明する図である。

【 図 3 】 本文書の実施例が適用できるビデオ/映像デコーディング装置の構成を概略的に説明する図である。

【 図 4 】 本文書の実施例が適用可能な概略的なビデオ/映像エンコーディング方法の一例を示す。

50

【図5】本文書の実施例が適用可能な概略的なビデオ/映像デコーディング方法の一例を示す。

【図6】本文書の実施例が適用可能なエントロピーエンコーディング方法の一例を概略的に示す。

【図7】エンコーディング装置内のエントロピーエンコーディング部を概略的に示す。

【図8】本文書の実施例が適用可能なエントロピーデコーディング方法の一例を概略的に示す。

【図9】デコーディング装置内のエントロピーデコーディング部を概略的に示す。

【図10】本文書の実施例(ら)によるビデオ/映像エンコーディング方法及び関連コンポーネントの一例を概略的に示す。

10

【図11】本文書の実施例(ら)によるビデオ/映像エンコーディング方法及び関連コンポーネントの一例を概略的に示す。

【図12】本文書の実施例(ら)によるビデオ/映像デコーディングの方法及び関連コンポーネントの一例を概略的に示す。

【図13】本文書の実施例(ら)によるビデオ/映像デコーディングの方法及び関連コンポーネントの一例を概略的に示す。

【図14】本文書において開示された実施例が適用できるコンテンツストリーミングシステムの例を示す。

【発明を実施するための形態】

【0020】

20

本文書は、様々な変更を加えることができ、種々の実施例を有することができ、特定実施例を図面に例示し、詳細に説明しようとする。しかしながら、これは、本文書を特定実施例に限定しようとするものではない。本明細書で常用する用語は、単に特定の実施例を説明するために使用されたものであって、本文書の技術的思想を限定しようとする意図で使用されるものではない。単数の表現は、文脈上明白に異なるように意味しない限り、複数の表現を含む。本明細書において「含む」または「有する」などの用語は、明細書上に記載された特徴、数字、ステップ、動作、構成要素、部品、またはこれらを組み合わせたものが存在することを指定しようとするものであり、一つまたはそれ以上の他の特徴や数字、ステップ、動作、構成要素、部品、またはこれらを組み合わせたものなどの存在または付加可能性を予め排除しないことと理解されるべきである。

30

【0021】

一方、本文書で説明される図面上の各構成は、互いに異なる特徴的な機能に関する説明の都合上、独立に図示されたものであって、各構成が互いに別個のハードウェアや別個のソフトウェアで具現されるということの意味するものではない。例えば、各構成のうち、二つ以上の構成が結合されて一つの構成をなすこともでき、一つの構成を複数の構成に分けることもできる。各構成が統合及び/または分離された実施例も本文書の本質から外れない限り、本文書の権利範囲に含まれる。

【0022】

本文書において「A又はB(A or B)」は「ただA」、「ただB」又は「AとBの両方とも」を意味することができる。言い換えれば、本文書において「A又はB(A or B)」は「A及び/又はB(A and/or B)」と解釈されることができる。例えば、本明細書において「A、B又はC(A、B or C)」は「ただA」、「ただB」、「ただC」、又は「A、B及びCの任意の全ての組み合わせ(any combination of A、B and C)」を意味することができる。

40

【0023】

本明細書で使われるスラッシュ(/)や読点(comma)は「及び/または(and/or)」を意味することができる。例えば、「A/B」は「A及び/又はB」を意味することができる。それによって、「A/B」は「ただA」、「ただB」、又は「AとBの両方とも」を意味することができる。例えば、「A、B、C」は「A、B又はC」を意味することができる。

50

【0024】

本明細書において「少なくとも1つのA及びB (at least one of A and B)」は、「ただA」、「ただB」又は「AとBの両方とも」を意味することができる。また、本明細書において「少なくとも1つのA又はB (at least one of A or B)」や「少なくとも1つのA及び/又はB (at least one of A and/or B)」という表現は「少なくとも1つのA及びB (at least one of A and B)」と同じく解釈されることができる。

【0025】

また、本明細書において「少なくとも1つのA、B及びC (at least one of A, B and C)」は、「ただA」、「ただB」、「ただC」、又は「A、B及びCの任意の全ての組み合わせ (any combination of A, B and C)」を意味することができる。また、「少なくとも1つのA、BまたはC (at least one of A, B or C)」や「少なくとも1つのA、B及び/又はC (at least one of A, B and/or C)」は「少なくとも1つのA、B及びC (at least one of A, B and C)」を意味することができる。

10

【0026】

また、本明細書で使われる括弧は「例えば (for example)」を意味することができる。具体的に、「予測 (イントラ予測)」で表示された場合、「予測」の一例として「イントラ予測」が提案されたものであり得る。言い換えると、本明細書の「予測」は「イントラ予測」に制限 (limit) されずに、「イントラ予測」が「予測」の一例として提案されたものであり得る。また、「予測 (すなわち、イントラ予測)」で表示された場合も、「予測」の一例として「イントラ予測」が提案されたものであり得る。

20

【0027】

本文書は、ビデオ/映像コーディングに関する。例えば、本文書において開示された方法/実施例はVVC (versatile video coding) 標準に開示される方法に適用されることができる。また、本文書において開示された方法/実施例はEVC (essential video coding) 標準、AV1 (AOMedia Video 1) 標準、AVS2 (2nd generation of audio video coding standard) 又は次世代ビデオ/映像コーディング標準 (例えば、H.267又はH.268など) に開示される方法に適用されることができる。

30

【0028】

この文書ではビデオ/映像コーディングに関する多様な実施例を提示し、他の言及がない限り、前記実施例は、互いに組み合わせて実行されることもできる。

【0029】

この文書においてビデオ (video) は、時間の流れによる一連の映像 (image) の集合を意味し得る。ピクチャ (picture) は、一般に特定の時間帯の1つの映像を示す単位を意味し、スライス (slice) / タイル (tile) はコーディングにおいてピクチャの一部を構成する単位である。スライス/タイルは、1つ以上のCTU (coding tree unit) を含んでもよい。1つのピクチャは1つ以上のスライス/タイルで構成されてもよい。1つのピクチャは1つ以上のタイルグループで構成されてもよい。タイルはピクチャ内の特定タイル列及び特定タイル列以内のCTUの四角領域である (A tile is a rectangular region of CTUs within a particular tile column and a particular tile row in a picture)。前記タイル列はCTUの四角領域であり、前記四角領域は前記ピクチャの高さと同一の高さを有し、幅はピクチャパラメータセット内のシンタックス要素により明示される (The tile column is a rectangular region of CTUs having a height equal to the height of the picture and a width specified by syntax elements in the picture parameter set)。前記タイル行はCTUの四角領域であり、前記四角領域はピクチャパラメータセット内のシンタックス要素により明示される幅を有し、高さは前記ピクチャの高さと同一であり得る (The tile row is a recta

40

50

ngular region of CTUs having a height specified by syntax elements in the picture parameter set and a width equal to the width of the picture)。タイルスキューンはピクチャをパーティショニングするCTUの特定の順次オーダリングを示し、前記CTUはタイル内のCTUラスタスキャンで連続整列され、ピクチャ内のタイルは前記ピクチャの前記タイルのラスタスキャンで連続整列される (A tile scan is a specific sequential ordering of CTUs partitioning a picture in which the CTUs are ordered consecutively in CTU raster scan in a tile whereas tiles in a picture are ordered consecutively in a raster scan of the tiles of the picture)。スライスは単一NALユニットに排他的に含まれ得る、整数個の完全なタイル又はピクチャのタイル内の整数個の連続的な完全なCTU行を含む (A slice includes an integer number of complete tiles or an integer number of consecutive complete CTU rows within a tile of a picture that may be exclusively contained in a single NAL unit)。

10

【0030】

一方、1つのピクチャは2つ以上のサブピクチャに区分される。サブピクチャはピクチャ内の1つ以上のスライスの四角領域である (an rectangular region of one or more slices within a picture)。

【0031】

ピクセル (pixel) またはペル (pel) は、一つのピクチャ (または、映像) を構成する最小の単位を意味することができる。また、ピクセルに対応する用語として「サンプル (sample)」が使用されることができる。サンプルは、一般的にピクセルまたはピクセルの値を示すことができ、ルマ (luma) 成分のピクセル / ピクセル値のみを示すこともでき、クロマ (chroma) 成分のピクセル / ピクセル値のみを示すこともできる。または、サンプルは、空間ドメインでのピクセル値を意味することもでき、このようなピクセル値が周波数ドメインに変換されると、周波数ドメインでの変換係数を意味することもできる。

20

【0032】

ユニット (unit) は、映像処理の基本単位を示すことができる。ユニットは、ピクチャの特定領域及び当該領域に関連した情報のうち少なくとも一つを含むことができる。一つのユニットは、一つのルマブロック及び二つのクロマ (例えば、cb、cr) ブロックを含むことができる。ユニットは、場合によって、ブロック (block) または領域 (area) などの用語と混用して使用されることができる。一般的な場合、M×Nブロックは、M個の列とN個の行からなるサンプル (または、サンプルアレイ)、または変換係数 (transform coefficient) の集合 (または、アレイ) を含むことができる。

30

【0033】

本文書において、量子化 / 逆量子化及び / 又は変換 / 逆変換の少なくとも一つは省略されてもよい。前記量子化 / 逆量子化が省略される場合、量子化された変換係数は変換係数と呼ばれてもよい。変換 / 逆変換が省略される場合、変換係数は係数又はレジデュアル係数と呼ばれてもよく、又は表現の統一性のために変換係数と依然として呼ばれてもよい。

【0034】

本文書において、量子化された変換係数及び変換係数は、それぞれ変換係数及びスケールリングされた (scaled) 変換係数と呼ばれてもよい。この場合、レジデュアル情報は、変換係数 (ら) に関する情報を含み、変換係数 (ら) に関する情報は、レジデュアルコーディングシンタックスを介してシグナリングされることができる。レジデュアル情報 (又は、前記変換係数 (ら) に関する情報) に基づいて変換係数が導出され、変換係数に対する逆変換 (スケールリング) によりスケールリングされた変換係数が導出される。スケールリングされた変換係数に対する逆変換 (変換) に基づいてレジデュアルサンプルが導出される。これは、本文書の他の部分においても同様に適用 / 表現できる。

40

【0035】

本文書において1つの図面内で個別的に説明される技術的特徴は、個別的に実現されて

50

もよく、同時に実現されてもよい。

【 0 0 3 6 】

以下、添付の図面を参照して、本文書の好ましい実施例をより詳細に説明する。以下、図面上の同一の構成要素に対しては同一の参照符号を使用し、同一の構成要素に関する重複した説明は省略される。

【 0 0 3 7 】

図 1 は、本文書の実施例に適用できるビデオ / 映像コーディングシステムの例を概略的に示す。

【 0 0 3 8 】

図 1 に示すように、ビデオ / 映像コーディングシステムは第 1 装置 (ソースデバイス) 及び第 2 装置 (受信デバイス) を含む。ソースデバイスはエンコーディングされたビデオ (v i d e o) / 映像 (i m a g e) 情報又はデータをファイル又はストリーミング形態でデジタル格納媒体又はネットワークを介して受信デバイスに伝達することができる。

【 0 0 3 9 】

前記ソースデバイスは、ビデオソース、エンコーディング装置、送信部を含むことができる。前記受信デバイスは、受信部、デコーディング装置、及びレンダラを含むことができる。前記エンコーディング装置は、ビデオ / 映像エンコーディング装置と呼ばれることができ、前記デコーディング装置は、ビデオ / 映像デコーディング装置と呼ばれることができる。送信機は、エンコーディング装置に含まれることができる。受信機は、デコーディング装置に含まれることができる。レンダラは、ディスプレイ部を含むこともでき、ディスプレイ部は、別個のデバイスまたは外部コンポーネントで構成されることもできる。

【 0 0 4 0 】

ビデオソースは、ビデオ / 映像のキャプチャ、合成または生成過程などを介してビデオ / 映像を取得することができる。ビデオソースは、ビデオ / 映像キャプチャデバイス及び / またはビデオ / 映像生成デバイスを含むことができる。ビデオ / 映像キャプチャデバイスは、例えば、一つ以上のカメラ、以前にキャプチャされたビデオ / 映像を含むビデオ / 映像アーカイブなどを含むことができる。ビデオ / 映像生成デバイスは、例えば、コンピュータ、タブレット、及びスマートフォンなどを含むことができ、(電子的に) ビデオ / 映像を生成することができる。例えば、コンピュータなどを介して仮想のビデオ / 映像が生成されることができ、この場合、ビデオ / 映像キャプチャ過程を関連データが生成される過程に代替されることができ。

【 0 0 4 1 】

エンコーディング装置は、入力ビデオ / 映像をエンコーディングすることができる。エンコーディング装置は、圧縮及びコーディング効率のために、予測、変換、量子化など、一連の手順を実行することができる。エンコーディングされたデータ (エンコーディングされたビデオ / 映像情報) は、ビットストリーム (b i t s t r e a m) 形態で出力されることができる。

【 0 0 4 2 】

送信部は、ビットストリーム形態で出力されたエンコーディングされたビデオ / 映像情報またはデータをファイルまたはストリーミング形態でデジタル格納媒体またはネットワークを介して受信デバイスの受信部に伝達できる。デジタル格納媒体は、U S B、S D、C D、D V D、ブルーレイ、H D D、S S D など、多様な格納媒体を含むことができる。送信部は、あらかじめ決められたファイルフォーマットを介してメディアファイルを生成するためのエレメントを含むことができ、放送 / 通信ネットワークを介した送信のためのエレメントを含むことができる。受信部は、前記ビットストリームを受信 / 抽出してデコーディング装置に伝達できる。

【 0 0 4 3 】

デコーディング装置は、エンコーディング装置の動作に対応する逆量子化、逆変換、予測など、一連の手順を実行してビデオ / 映像をデコーディングすることができる。

【 0 0 4 4 】

10

20

30

40

50

レンダラは、デコーディングされたビデオ/映像をレンダリングすることができる。レンダリングされたビデオ/映像は、ディスプレイ部を介してディスプレイされることができる。

【0045】

図2は、本文書の実施例が適用できるビデオ/映像エンコーディング装置の構成を概略的に説明する図である。以下、エンコーディング装置とは、映像エンコーディング装置及び/又はビデオエンコーディング装置を含む。

【0046】

図2に示すように、エンコーディング装置200は、画像分割部(image partitioner)210、予測部(predictor)220、レジデュアル処理部(residual processor)230、エントロピーエンコーディング部(entropy encoder)240、加算部(adder)250、フィルタリング部(filter)260、及びメモリ(memory)270を備えて構成されることができる。予測部220は、インター予測部221及びイントラ予測部222を備えることができる。レジデュアル処理部230は、変換部(transformer)232、量子化部(quantizer)233、逆量子化部(dequantizer)234、逆変換部(inverse transformer)235を備えることができる。レジデュアル処理部230は、減算部(subtractor、231)をさらに備えることができる。加算部250は、復元部(reconstructor)または復元ブロック生成部(reconstructed block generator)と呼ばれることができる。前述した画像分割部210、予測部220、レジデュアル処理部230、エントロピーエンコーディング部240、加算部250、及びフィルタリング部260は、実施形態によって1つ以上のハードウェアコンポーネント(例えば、エンコーダチップセットまたはプロセッサ)によって構成されることができる。また、メモリ270は、DPB(decoded picture buffer)を備えることができ、デジタル格納媒体によって構成されることができる。前記ハードウェアコンポーネントは、メモリ270を内/外部コンポーネントとしてさらに備えることもできる。

【0047】

画像分割部210は、エンコーディング装置200に入力された入力画像(または、ピクチャ、フレーム)を1つ以上の処理ユニット(processing unit)に分割することができる。一例として、前記処理ユニットは、コーディングユニット(coding unit、CU)と呼ばれることができる。この場合、コーディングユニットは、コーディングツリーユニット(coding tree unit、CTU)または最大コーディングユニット(largest coding unit、LCU)からQTBT(T Quad-tree binary-tree ternary-tree)構造によって再帰的に(recursively)分割されることができる。例えば、1つのコーディングユニットは、クアドツリー構造、バイナリツリー構造、及び/またはターナリ構造に基づいて下位(deeper)デプスの複数のコーディングユニットに分割されることができる。この場合、例えば、クアドツリー構造が先に適用され、バイナリツリー構造及び/またはターナリ構造がその後適用されることができる。または、バイナリツリー構造が先に適用されることがもできる。それ以上分割されない最終コーディングユニットに基づいて本開示に係るコーディング手順が行われ得る。この場合、画像特性によるコーディング効率などに基づいて、最大コーディングユニットが最終コーディングユニットとして使用されることができ、または、必要に応じてコーディングユニットは、再帰的に(recursively)もっと下位デプスのコーディングユニットに分割されて最適のサイズのコーディングユニットが最終コーディングユニットとして使用されることができる。ここで、コーディング手順とは、後述する予測、変換、及び復元などの手順を含むことができる。他の例として、前記処理ユニットは、予測ユニット(PU: Prediction Unit)または変換ユニット(TU: Transform Unit)をさらに備えることができる。この場合、前記予測ユニット及び前記変換ユニットは、各々前

10

20

30

40

50

述した最終コーディングユニットから分割またはパーティショニングされることができる。前記予測ユニットは、サンプル予測の単位であることができ、前記変換ユニットは、変換係数を導く単位及び/または変換係数からレジデュアル信号 (residual signal) を導く単位であることができる。

【0048】

ユニットは、場合によって、ブロック (block) または領域 (area) などの用語と混用して使用されることができる。一般的な場合、 $M \times N$ ブロックは、 M 個の列と N 個の行からなるサンプルまたは変換係数 (transform coefficient) 等の集合を示すことができる。サンプルは、一般的にピクセルまたはピクセルの値を示すことができ、輝度 (luma) 成分のピクセル/ピクセル値のみを示すことができ、彩度 (chroma) 成分のピクセル/ピクセル値のみを示すこともできる。サンプルは、1つのピクチャ (または、画像) をピクセル (pixel) またはペル (pel) に対応する用語として使用することができる。

【0049】

エンコーディング装置 200 は、入力映像信号 (原本ブロック、原本サンプルアレイ) からインター予測部 221 又はイントラ予測部 222 から出力された予測信号 (予測されたブロック、予測サンプルアレイ) を減算してレジデュアル信号 (residual signal、残余ブロック、残余サンプルアレイ) を生成し、生成されたレジデュアル信号は、変換部 232 に送信される。この場合、図示されているように、エンコーダ 200 内において入力映像信号 (原本ブロック、原本サンプルアレイ) から予測信号 (予測ブロック、予測サンプルアレイ) を減算するユニットは、減算部 231 と呼ばれてもよい。予測部は、処理対象ブロック (以下、現在ブロックという) に対する予測を行い、前記現在ブロックに対する予測サンプルを含む予測されたブロック (predicted block) を生成する。予測部は、現在ブロック又はCU単位でイントラ予測が適用されるか、又はインター予測が適用されるか決定することができる。予測部は、各予測モードに関する説明において後述するように、予測モード情報などの予測に関する様々な情報を生成してエントロピーエンコーディング部 240 に伝達する。予測に関する情報は、エントロピーエンコーディング部 240 においてエンコーディングされてビットストリーム形態で出力される。

【0050】

イントラ予測部 222 は、現在ピクチャ内のサンプルを参照して現在ブロックを予測することができる。前記参照されるサンプルは、予測モードによって前記現在ブロックに隣接 (neighbor) して位置することができ、または、離れて位置することもできる。イントラ予測で予測モードは、複数の非方向性モードと複数の方向性モードとを含むことができる。非方向性モードは、例えば、DCモード及びプラナーモード (Planar mode) を含むことができる。方向性モードは、予測方向の細かい程度によって、例えば、33個の方向性予測モードまたは65個の方向性予測モードを含むことができる。ただし、これは、例示であり、設定によってそれ以上またはそれ以下の個数の方向性予測モードが使用され得る。イントラ予測部 222 は、隣接ブロックに適用された予測モードを用いて、現在ブロックに適用される予測モードを決定することもできる。

【0051】

インター予測部 221 は、参照ピクチャ上で動きベクトルにより特定される参照ブロック (参照サンプルアレイ) に基づいて、現在ブロックに対する予測されたブロックを導くことができる。このとき、インター予測モードで送信される動き情報の量を減らすために、隣接ブロックと現在ブロックとの間の動き情報の相関性に基づいて、動き情報をブロック、サブブロック、またはサンプル単位で予測することができる。前記動き情報は、動きベクトル及び参照ピクチャインデックスを含むことができる。前記動き情報は、インター予測方向 (L0予測、L1予測、Bi予測等) 情報をさらに含むことができる。インター予測の場合に、隣接ブロックは、現在ピクチャ内に存在する空間的隣接ブロック (spatial neighboring block) と参照ピクチャに存在する時間的隣接ブ

10

20

30

40

50

ロック (temporal neighboring block) とを含むことができる。前記参照ブロックを含む参照ピクチャと前記時間的隣接ブロックを含む参照ピクチャとは同じであることができ、異なることもできる。前記時間的隣接ブロックは、同一位置参照ブロック (collocated reference block)、同一位置CU (col CU) などの名前では呼ばれることができ、前記時間的隣接ブロックを含む参照ピクチャは、同一位置ピクチャ (collocated picture、colPic) と呼ばれることもできる。例えば、インター予測部 221 は、隣接ブロックに基づいて動き情報候補リストを構成し、前記現在ブロックの動きベクトル及び/または参照ピクチャインデックスを導出するために、どの候補が使用されるかを指示する情報を生成することができる。様々な予測モードに基づいてインター予測が行われ得るし、例えば、スキップモードとマージモードとの場合に、インター予測部 221 は、隣接ブロックの動き情報を現在ブロックの動き情報として利用することができる。スキップモードの場合、マージモードとは異なり、レジデュアル信号が送信されないことがある。動き情報予測 (motion vector prediction、MVP) モードの場合、隣接ブロックの動きベクトルを動きベクトル予測子 (motion vector predictor) として用い、動きベクトル差分 (motion vector difference) をシグナリングすることにより、現在ブロックの動きベクトルを指示することができる。

10

【0052】

予測部 220 は後述する様々な予測方法に基づいて予測信号を生成する。例えば、予測部は1つのブロックに対する予測のためにイントラ予測又はインター予測を適用できるだけでなく、イントラ予測とインター予測を同時に適用する。これは、combined inter and intra prediction (CIIP) と呼ばれてもよい。また、予測部は、ブロックに対する予測のために、イントラブロックコピー (intra block copy、IBC) 予測モードに基づくことができ、又はパレットモード (palette mode) に基づくこともできる。前記IBC予測モード又はパレットモードは、例えば、SCC (screen content coding) などのようにゲームなどのコンテンツ映像/動画コーディングのために使用される。IBCは、基本的に現在ピクチャ内において予測を行うが、現在ピクチャ内において参照ブロックを導出する点においてインター予測と類似して行われる。すなわち、IBCは、本文書において説明されるインター予測技法のうち少なくとも1つを用いることができる。パレットモードはイントラコーディング又はイントラ予測の一例と見なすことができる。パレットモードが適用される場合、パレットテーブル及びパレットインデックスに関する情報に基づいてピクチャ内のサンプル値をシグナリングする。

20

30

【0053】

前記予測部 (インター予測部 221 及び/又は前記イントラ予測部 222 を含む) により生成された予測信号は、復元信号を生成するために用いられるか、レジデュアル信号を生成するために用いられる。変換部 232 は、レジデュアル信号に変換技法を適用して変換係数 (transform coefficients) を生成する。例えば、変換技法は、DCT (Discrete Cosine Transform)、DST (Discrete Sine Transform)、KLT (Karhunen - Loeve Transform)、GBT (Graph - Based Transform)、又はCNT (Conditionally Non - linear Transform) の少なくとも1つを含む。ここで、GBTは、ピクセル間の関係情報をグラフで表現する時、このグラフから得られた変換を意味する。CNTは、以前に復元された全てのピクセル (all previously reconstructed pixel) を用いて予測信号を生成し、それに基づいて取得される変換を意味する。また、変換過程は正方形の同一のサイズを有するピクセルブロックに適用されてもよく、正方形ではない可変サイズのブロックに適用されてもよい。

40

【0054】

量子化部 233 は、変換係数を量子化してエントロピーエンコーディング部 240 に送

50

信し、エントロピーエンコーディング部240は、量子化された信号（量子化された変換係数に関する情報）をエンコーディングしてビットストリームに出力する。前記量子化された変換係数に関する情報は、レジデュアル情報と呼ばれることができる。量子化部233は、係数スキャン順序（scan order）に基づいてブロック形態の量子化された変換係数を1次元ベクトル形態で再整列することができ、前記1次元ベクトル形態の量子化された変換係数に基づいて前記量子化された変換係数に関する情報を生成することもできる。エントロピーエンコーディング部240は、例えば、指数ゴロム（exponential Golomb）、CAVLC（context-adaptive variable length coding）、CABAC（context-adaptive binary arithmetic coding）などの様々なエンコーディング方法を行うことができる。エントロピーエンコーディング部240は、量子化された変換係数の他に、ビデオ/イメージ復元に必要な情報（例えば、シンタックス要素（syntax elements）の値など）を共に又は別にエンコーディングすることもできる。エンコーディングされた情報（例えば、エンコーディングされたビデオ/映像情報）は、ビットストリーム形態でNAL（network abstraction layer）ユニット単位で送信又は格納されることができる。前記ビデオ/映像情報は、アダプテーションパラメータセット（APS）、ピクチャパラメータセット（PPS）、シーケンスパラメータセット（SPS）、又はビデオパラメータセット（VPS）などの様々なパラメータセットに関する情報をさらに含んでもよい。また、前記ビデオ/映像情報は、一般制限情報（general constraint information）をさらに含んでもよい。本文書において、エンコーディング装置からデコーディング装置に伝達/シグナリングされる情報及び/又はシンタックス要素は、ビデオ/映像情報に含まれる。前記ビデオ/映像情報は、前述のエンコーディング手順によりエンコーディングされて前記ビットストリームに含まれる。前記ビットストリームは、ネットワークを介して送信されるか、又はデジタル格納媒体に格納されることができる。ここで、ネットワークは、放送網及び/又は通信網などを含み、デジタル格納媒体は、USB、SD、CD、DVD、ブルーレイ、HDD、SSDなどの様々な格納媒体を含む。エントロピーエンコーディング部240から出力された信号を送信する送信部（図示せず）及び/又は格納する格納部（図示せず）がエンコーディング装置200の内/外部エレメントとして構成されてもよく、又は、送信部はエントロピーエンコーディング部240に含まれてもよい。

【0055】

量子化部233から出力された量子化された変換係数は、予測信号を生成するために用いられることができる。例えば、量子化された変換係数に逆量子化部234及び逆変換部235を介して逆量子化及び逆変換を適用することにより、レジデュアル信号（レジデュアルブロック又はレジデュアルサンプル）を復元できる。加算部155は復元されたレジデュアル信号をインター予測部221又はイントラ予測部222から出力された予測信号に加えることにより復元（reconstructed）信号（復元ピクチャ、復元ブロック、復元サンプルアレイ）を生成する。スキップモードが適用された場合のように、処理対象ブロックに対するレジデュアルがない場合、予測されたブロックが復元ブロックとして使用されることができる。加算部250は、復元部又は復元ブロック生成部と呼ばれてもよい。生成された復元信号は、現在ピクチャ内の次の処理対象ブロックのイントラ予測のために使用されることができ、後述するように、フィルタリングを経て次のピクチャのインター予測のために使用されることがもできる。

【0056】

一方、ピクチャエンコーディング及び/または復元過程でLMCS（luma mapping with chroma scaling）が適用されることもできる。

【0057】

フィルタリング部260は、復元信号にフィルタリングを適用して主観的/客観的画質を向上させることができる。例えば、フィルタリング部260は、復元ピクチャに多様なフィルタリング方法を適用して修正された（modified）復元ピクチャを生成する

10

20

30

40

50

ことができ、前記修正された復元ピクチャをメモリ270、具体的に、メモリ270のDPBに格納することができる。前記多様なフィルタリング方法は、例えば、デブロッキングフィルタリング、サンプル適応的オフセット (sample adaptive offset、SAO)、適応的ループフィルタ (adaptive loop filter)、両方向フィルタ (bilateral filter) などを含むことができる。フィルタリング部260は、各フィルタリング方法に対する説明で後述するように、フィルタリングに関する多様な情報を生成してエントロピーエンコーディング部290に伝達できる。フィルタリングに関する情報は、エントロピーエンコーディング部290でエンコーディングされてビットストリーム形態で出力されることができる。

【0058】

メモリ270に送信された修正された復元ピクチャは、インター予測部280で参照ピクチャとして使われることができる。エンコーディング装置は、これを介してインター予測が適用される場合、エンコーディング装置200とデコーディング装置での予測ミスマッチを避けることができ、符号化効率も向上させることができる。

【0059】

メモリ270のDPBは、修正された復元ピクチャをインター予測部221での参照ピクチャとして使用するために格納することができる。メモリ270は、現在ピクチャ内の動き情報が導出された (または、エンコーディングされた) ブロックの動き情報及び/または既に復元されたピクチャ内のブロックの動き情報を格納することができる。前記格納された動き情報は、空間的隣接ブロックの動き情報または時間的隣接ブロックの動き情報として活用するためにインター予測部221に伝達できる。メモリ270は、現在ピクチャ内の復元されたブロックの復元サンプルを格納することができ、イントラ予測部222に伝達できる。

【0060】

図3は、本文書の実施例が適用できるビデオ/映像デコーディング装置の構成を概略的に説明する図である。以下、デコーディング装置とは、映像デコーディング装置及び/又はビデオデコーディング装置を含む。

【0061】

図3に示すように、デコーディング装置300は、エントロピーデコーディング部 (entropy decoder) 310、レジデュアル処理部 (residual processor) 320、予測部 (predictor) 330、加算部 (adder) 340、フィルタリング部 (filter) 350、及びメモリ (memory) 360を備えて構成されることができる。予測部330は、インター予測部332及びイントラ予測部331を備えることができる。レジデュアル処理部320は、逆量子化部 (dequantizer) 321及び逆変換部 (inverse transformer) 321を備えることができる。前述したエントロピーデコーディング部310、レジデュアル処理部320、予測部330、加算部340、及びフィルタリング部350は、実施形態によって1つのハードウェアコンポーネント (例えば、デコーダチップセットまたはプロセッサ) により構成されることができる。また、メモリ360は、DPB (decoded picture buffer) を備えることができ、デジタル格納媒体により構成されることもできる。前記ハードウェアコンポーネントは、メモリ360を内/外部コンポーネントとしてさらに備えることもできる。

【0062】

ビデオ/画像情報を含むビットストリームが入力されれば、デコーディング装置300は、図3のエンコーディング装置でビデオ/画像情報が処理されたプロセスに対応して画像を復元することができる。例えば、デコーディング装置300は、前記ビットストリームから取得したブロック分割関連情報に基づいてユニット/ブロックを導出することができる。デコーディング装置300は、エンコーディング装置で適用された処理ユニットを用いてデコーディングを行うことができる。従って、デコーディングの処理ユニットは、例えば、コーディングユニットであることができ、コーディングユニットは、コーディン

10

20

30

40

50

グツリーユニットまたは最大コーディングユニットからクアドツリー構造、バイナリツリー構造、及び/またはターナリツリー構造に従って分割されることができる。コーディングユニットから1つ以上の変換ユニットが導出され得る。そして、デコーディング装置300を介してデコーディング及び出力された復元画像信号は、再生装置を介して再生されることができる。

【0063】

デコーディング装置300は、図2のエンコーディング装置から出力された信号をビットストリーム形態で受信し、受信された信号は、エントロピーデコーディング部310を介してデコーディングされる。例えば、エントロピーデコーディング部310は、前記ビットストリームをパーシングして映像復元（又は、ピクチャ復元）に必要な情報（例えば、ビデオ/映像情報）を導出することができる。前記ビデオ/映像情報は、アダプテーションパラメータセット（APS）、ピクチャパラメータセット（PPS）、シーケンスパラメータセット（SPS）、又はビデオパラメータセット（VPS）などの様々なパラメータセットに関する情報をさらに含んでもよい。また、前記ビデオ/映像情報は、一般制限情報（general constraint information）をさらに含んでもよい。デコーディング装置は、前記パラメータセットに関する情報及び/又は前記一般制限情報に基づいてさらにピクチャをデコーディングすることができる。本文書において後述されるシグナリング/受信される情報及び/又はシンタックス要素は、前記デコーディング手順を介してデコーディングされて前記ビットストリームから取得されることができる。例えば、エントロピーデコーディング部310は、指数ゴロム符号化、CAVLC、又はCABACなどのコーディング方法に基づいてビットストリーム内の情報をデコーディングし、映像復元に必要なシンタックス要素の値、レジデュアルに関する変換係数の量子化された値を出力することができる。より詳細に、CABACエントロピーデコーディング方法は、ビットストリームにおいて各構文要素に該当するピンを受信し、デコーディング対象構文要素情報と周辺及びデコーディング対象ブロックのデコーディング情報あるいは以前ステップでデコーディングされたシンボル/ピンの情報を利用して文脈（context）モデルを決定し、決定された文脈モデルによってピン（bin）の発生確率を予測してピンの算術デコーディング（arithmetic decoding）を行うことにより各構文要素の値に該当するシンボルを生成できる。このとき、CABACエントロピーデコーディング方法は、文脈モデル決定後、次のシンボル/ピンの文脈モデルのためにデコーディングされたシンボル/ピンの情報を利用して文脈モデルをアップデートすることができる。エントロピーデコーディング部310でデコーディングされた情報のうち予測に関する情報は予測部（インター予測部332及びイントラ予測部331）に提供され、エントロピーデコーディング部310でエントロピーデコーディングが行われたレジデュアル値、すなわち、量子化された変換係数及び関連パラメータ情報は、レジデュアル処理部320に入力される。レジデュアル処理部320は、レジデュアル信号（レジデュアルブロック、レジデュアルサンプル、レジデュアルサンプルアレイ）を導出できる。また、エントロピーデコーディング部310でデコーディングされた情報のうちフィルタリングに関する情報は、フィルタリング部350に提供される。一方、エンコーディング装置から出力された信号を受信する受信部（図示せず）がデコーディング装置300の内/外部エレメントとしてさらに構成されてもよく、又は受信部は、エントロピーデコーディング部310の構成要素であってもよい。一方、本文書によるデコーディング装置は、ビデオ/映像/ピクチャデコーディング装置と呼ばれることができ、前記デコーディング装置は、情報デコーダ（ビデオ/映像/ピクチャ情報デコーダ）及びサンプルデコーダ（ビデオ/映像/ピクチャサンプルデコーダ）に区分することもできる。前記情報デコーダは、前記エントロピーデコーディング部310を含んでもよく、前記サンプルデコーダは、前記逆量子化部321、逆変換部322、加算部340、フィルタリング部350、メモリ360、インター予測部332、及びイントラ予測部331のうち少なくとも1つを含んでもよい。

【0064】

10

20

30

40

50

逆量子化部 3 2 1 では、量子化された変換係数を逆量子化して変換係数を出力することができる。逆量子化部 3 2 1 は、量子化された変換係数を 2 次元のブロック形態で再整理することができる。この場合、前記再整理は、エンコーディング装置で行われた係数スキャン順序に基づいて再整理を行うことができる。逆量子化部 3 2 1 は、量子化パラメータ（例えば、量子化ステップサイズ情報）を用いて量子化された変換係数に対する逆量子化を行い、変換係数 (transform coefficient) を取得することができる。

【0065】

逆変換部 3 2 2 では、変換係数を逆変換してレジデュアル信号 (レジデュアルブロック、レジデュアルサンプルアレイ) を取得するようになる。

10

【0066】

予測部は、現在ブロックに対する予測を行い、前記現在ブロックに対する予測サンプルを含む予測されたブロック (predicted block) を生成することができる。予測部は、エントロピーデコーディング部 3 1 0 から出力された前記予測に関する情報に基づいて、前記現在ブロックにイントラ予測が適用されるか、または、インター予測が適用されるかを決定することができ、具体的なイントラ/インター予測モードを決定する。

【0067】

予測部 3 2 0 は、後述する様々な予測方法に基づいて予測信号を生成できる。例えば、予測部は、1つのブロックに対する予測のためにイントラ予測又はインター予測を適用できるだけでなく、イントラ予測とインター予測とを同時に適用することができる。これは、combined inter and intra prediction (CIIP) と呼ばれることができる。また、予測部は、ブロックに対する予測のために、イントラブロックコピー (intra block copy、IBC) 予測モードに基づくことができ、またはパレットモード (palette mode) に基づくこともできる。前記 IBC 予測モード又はパレットモードは、例えば、SCC (screen content coding) などのように、ゲームなどのコンテンツ映像/動画コーディングのために使用されることができる。IBC は、基本的に現在ピクチャ内において予測を行うが、現在ピクチャ内において参照ブロックを導出する点でインター予測と類似して行われることができる。すなわち、IBC は、本文書において説明されるインター予測技法のうち少なくとも1つを利用することができる。パレットモードは、イントラコーディング又はイントラ予測の一例と見なすことができる。パレットモードが適用される場合、パレットテーブル及びパレットインデックスに関する情報が前記ビデオ/映像情報に含まれてシグナリングされる。

20

30

【0068】

イントラ予測部 3 3 1 は、現在ピクチャ内のサンプルを参照して現在ブロックを予測することができる。前記参照されるサンプルは、予測モードによって前記現在ブロックの隣接 (neighbor) に位置でき、または離れて位置することもできる。イントラ予測において、予測モードは、複数の非方向性モードと複数の方向性モードとを含むことができる。イントラ予測部 3 3 1 は、隣接ブロックに適用された予測モードを利用し、現在ブロックに適用される予測モードを決定することもできる。

40

【0069】

インター予測部 3 3 2 は、参照ピクチャ上で動きベクトルにより特定される参照ブロック (参照サンプルアレイ) に基づいて、現在ブロックに対する予測されたブロックを誘導することができる。このとき、インター予測モードで送信される動き情報の量を減らすために、隣接ブロックと現在ブロックとの間の動き情報の相関性に基づいて、動き情報をブロック、サブブロックまたはサンプル単位で予測できる。前記動き情報は、動きベクトル及び参照ピクチャインデックスを含むことができる。前記動き情報は、インター予測方向 (L0 予測、L1 予測、Bi 予測等) 情報をさらに含むことができる。インター予測の場合に、隣接ブロックは、現在ピクチャ内に存在する空間的隣接ブロック (spatial neighboring block) と、参照ピクチャ内に存在する時間的隣接ブロック

50

(temporal neighboring block)と、を含むことができる。例えば、インター予測部332は、隣接ブロックに基づいて動き情報候補リストを構成し、受信した候補選択情報に基づいて前記現在ブロックの動きベクトル及び/または参照ピクチャインデックスを導出することができる。多様な予測モードに基づいてインター予測が実行されることができ、前記予測に関する情報は、前記現在ブロックに対するインター予測のモードを指示する情報を含むことができる。

【0070】

加算部340は、取得されたレジデュアル信号を、予測部330から出力された予測信号(予測されたブロック、予測サンプルアレイ)に加えることによって復元信号(復元ピクチャ、復元ブロック、復元サンプルアレイ)を生成することができる。スキップモードが適用された場合のように処理対象ブロックに対するレジデュアルがない場合、予測されたブロックが復元ブロックとして使われることができる。

10

【0071】

加算部340は、復元部または復元ブロック生成部と呼ばれることができる。生成された復元信号は、現在ピクチャ内の次の処理対象ブロックのイントラ予測のために使われることができ、後述するように、フィルタリングを経て出力されることもでき、または、次のピクチャのインター予測のために使われることもできる。

【0072】

一方、ピクチャデコーディング過程でLMCS(luma mapping with chroma scaling)が適用されることもできる。

20

【0073】

フィルタリング部350は、復元信号にフィルタリングを適用して主観的/客観的画質を向上させることができる。例えば、フィルタリング部350は、復元ピクチャに多様なフィルタリング方法を適用して修正された(modified)復元ピクチャを生成することができ、前記修正された復元ピクチャをメモリ60、具体的に、メモリ360のDPBに送信できる。前記多様なフィルタリング方法は、例えば、デブロッキングフィルタリング、サンプル適応的オフセット(sample adaptive offset)、適応的ループフィルタ(adaptive loop filter)、両方向フィルタ(bilateral filter)などを含むことができる。

【0074】

メモリ360のDPBに格納された(修正された)復元ピクチャは、インター予測部332で参照ピクチャとして使われることができる。メモリ360は、現在ピクチャ内の動き情報が導出された(または、デコーディングされた)ブロックの動き情報及び/または既に復元されたピクチャ内のブロックの動き情報を格納することができる。前記格納された動き情報は、空間的隣接ブロックの動き情報または時間的隣接ブロックの動き情報として活用するためにインター予測部332に伝達できる。メモリ360は、現在ピクチャ内の復元されたブロックの復元サンプルを格納することができ、イントラ予測部331に伝達できる。

30

【0075】

本文書において、エンコーディング装置200のフィルタリング部260、インター予測部221及びイントラ予測部222において説明された実施例は、それぞれデコーディング装置300のフィルタリング部350、インター予測部332及びイントラ予測部331にも同一又は対応するように適用される。

40

【0076】

前述したように、ビデオコーディングを実行するにあたって圧縮効率を上げるために予測を実行する。それによって、コーディング対象ブロックである現在ブロックに対する予測サンプルを含む予測されたブロックを生成することができる。ここで、前記予測されたブロックは、空間ドメイン(または、ピクセルドメイン)での予測サンプルを含む。前記予測されたブロックは、エンコーディング装置及びデコーディング装置で同様に導出され、前記エンコーディング装置は、原本ブロックの原本サンプル値自体でない前記原本プロ

50

ックと前記予測されたブロックとの間のレジデュアルに関する情報（レジデュアル情報）をデコーディング装置にシグナリングすることで画像コーディング効率を上げることができる。デコーディング装置は、前記レジデュアル情報に基づいてレジデュアルサンプルを含むレジデュアルブロックを導出し、前記レジデュアルブロックと前記予測されたブロックを加算して復元サンプルを含む復元ブロックを生成することができ、復元ブロックを含む復元ピクチャを生成することができる。

【0077】

前記レジデュアル情報は、変換及び量子化手順を介して生成されることができる。例えば、エンコーディング装置は、前記原本ブロックと前記予測されたブロックとの間のレジデュアルブロックを導出し、前記レジデュアルブロックに含まれているレジデュアルサンプル（レジデュアルサンプルアレイ）に変換手順を実行して変換係数を導出し、前記変換係数に量子化手順を実行して量子化された変換係数を導出することで、関連したレジデュアル情報を（ビットストリームを介して）デコーディング装置にシグナリングすることができる。ここで、前記レジデュアル情報は、前記量子化された変換係数の値情報、位置情報、変換技法、変換カーネル、量子化パラメータなどの情報を含むことができる。デコーディング装置は、前記レジデュアル情報に基づいて逆量子化／逆変換手順を実行してレジデュアルサンプル（または、レジデュアルブロック）を導出することができる。デコーディング装置は、予測されたブロックと前記レジデュアルブロックに基づいて復元ピクチャを生成することができる。また、エンコーディング装置は、以後ピクチャのインター予測のための参照のために量子化された変換係数を逆量子化／逆変換してレジデュアルブロックを導出し、これに基づいて復元ピクチャを生成することができる。

【0078】

以下の図面は本文書の具体的な一例を説明するために作成された。図面に記載された具体的な装置の名称又は具体的な用語や名称（例えば、シンタックスの名称など）は例示的に示されたものであるので、本文書の技術的特徴が以下の図面に用いられた具体的な名称に限られない。

【0079】

図4は、本文書の実施例が適用可能な概略的なビデオ／映像エンコーディング方法の一例を示す。

【0080】

図4に開示された方法は、前述の図2のエンコーディング装置200により行われることができる。具体的に、S400はエンコーディング装置200のインター予測部221又はイントラ予測部222により行われ、S410、S420、S430、S440はそれぞれエンコーディング装置200の減算部231、変換部232、量子化部233、エントロピーエンコーディング部240により行われる。

【0081】

図4に示すように、エンコーディング装置は現在ブロックに対する予測により予測サンプルを導出する（S400）。エンコーディング装置は、現在ブロックにインター予測を行うか又はイントラ予測を行うかを決定し、具体的なインター予測モード又は具体的なイントラ予測モードをRDコストに基づいて決定する。決定されたモードに応じてエンコーディング装置は現在ブロックに対する予測サンプルを導出する。

【0082】

エンコーディング装置は、現在ブロックに対する原本サンプルと予測サンプルを比較してレジデュアルサンプルを導出する（S410）。

【0083】

エンコーディング装置は、レジデュアルサンプルに対する変換手順を介して変換係数を導出し（S420）、導出された変換係数を量子化して量子化された変換係数を導出する（S430）。

【0084】

量子化は、量子化パラメータに基づいて行われる。変換手順及び／又は量子化手順は省

10

20

30

40

50

略されてもよい。変換手順が省略される場合、レジデュアルサンプルに対する（量子化された）（レジデュアル）係数が後述するレジデュアルコーディング技法に従ってコーディングされる。（量子化された）（レジデュアル）係数も用語の統一のために（量子化された）変換係数と呼ばれてもよい。

【 0 0 8 5 】

エンコーディング装置は、予測情報及びレジデュアル情報を含む映像情報をエンコーディングし、エンコーディングされた映像情報をビットストリーム形態で出力する（S 4 4 0）。予測情報は予測手順に関する情報であり、予測モード情報及び動き情報に関する情報（例えば、インター予測が適用される場合）などを含む。レジデュアル情報は量子化された変換係数に関する情報を含む。レジデュアル情報はエントロピーコーディングされる

10

【 0 0 8 6 】

出力されたビットストリームは、格納媒体又はネットワークを介してデコーディング装置に伝達される。

【 0 0 8 7 】

図 5 は、本文書の実施例が適用可能な概略的なビデオ / 映像デコーディング方法の一例を示す。

【 0 0 8 8 】

図 5 に開示された方法は、前述の図 3 のデコーディング装置 3 0 0 により行われることができる。具体的に、S 5 0 0 はデコーディング装置 3 0 0 のインター予測部 3 3 2 又はイントラ予測部 3 3 1 により行われる。S 5 0 0 で、ビットストリームに含まれた予測情報をデコーディングして関連シンタックス要素の値を導出する手順は、デコーディング装置 3 0 0 のエントロピーデコーディング部 3 1 0 により行われる。S 5 1 0、S 5 2 0、S 5 3 0、S 5 4 0 は、それぞれデコーディング装置 3 0 0 のエントロピーデコーディング部 3 1 0、逆量子化部 3 2 1、逆変換部 3 2 2、加算部 3 4 0 により行われる。

20

【 0 0 8 9 】

図 5 に示すように、デコーディング装置はエンコーディング装置において行われた動作と対応する動作を行う。デコーディング装置は、受信された予測情報に基づいて現在ブロックに対するインター予測又はイントラ予測を行って予測サンプルを導出する（S 5 0 0）。

30

【 0 0 9 0 】

デコーディング装置は、受信されたレジデュアル情報に基づいて現在ブロックに対する量子化された変換係数を導出する（S 5 1 0）。デコーディング装置は、エントロピーデコーディングによりレジデュアル情報から量子化された変換係数を導出する。

【 0 0 9 1 】

デコーディング装置は、量子化された変換係数を逆量子化して変換係数を導出する（S 5 2 0）。逆量子化は量子化パラメータに基づいて行われる。

【 0 0 9 2 】

デコーディング装置は、変換係数に対する逆変換手順によりレジデュアルサンプルを導出する（S 5 3 0）。

40

【 0 0 9 3 】

逆変換手順及び / 又は逆量子化手順は省略されてもよい。逆変換手順が省略された場合、レジデュアル情報から（量子化された）（レジデュアル）係数が導出され、（量子化された）係数に基づいてレジデュアルサンプルが導出されることができる。

【 0 0 9 4 】

デコーディング装置は予測サンプル及びレジデュアルサンプルに基づいて現在ブロックに対する復元サンプルを生成し、これに基づいて復元ピクチャを生成する（S 5 4 0）。その後、復元ピクチャにインループフィルタリング手順がさらに適用できることは前述の通りである。

50

【0095】

一方、前述のように、エンコーディング装置は、例えば、指数ゴロム (exponential Golomb)、CAVLC (context-adaptive variable length coding)、CABAC (context-adaptive binary arithmetic coding) などの様々なエンコーディング方法に基づいてエントロピーエンコーディングを行う。また、デコーディング装置は、指数ゴロム符号化、CAVLC又はCABACなどのコーディング方法に基づいてエントロピーデコーディングを行う。以下では、エントロピーエンコーディング/デコーディング手順について説明する。

【0096】

図6は、本文書の実施例が適用可能なエントロピーエンコーディング方法の一例を概略的に示し、図7は、エンコーディング装置内のエントロピーエンコーディング部を概略的に示す。図7のエンコーディング装置内のエントロピーエンコーディング部は前述の図2のエンコーディング装置200のエントロピーエンコーディング部240にも同一又は対応するように適用できる。

【0097】

図6及び図7に示すように、エンコーディング装置 (エントロピーエンコーディング部) は、映像/ビデオ情報に関するエントロピーコーディング手順を行う。映像/ビデオ情報は、パーティショニング関連情報、予測関連情報 (例えば、インター/イントラ予測区分情報、イントラ予測モード情報、インター予測モード情報など)、レジデュアル情報、インループフィルタリング関連情報などを含み、又はそれに関する多様なシンタックス要素を含む。エントロピーコーディングはシンタックス要素単位で行われる。S600ないしS610は、前述した図2のエンコーディング装置200のエントロピーエンコーディング部240により行われることができる。

【0098】

エンコーディング装置は対象シンタックス要素に対する二進化を行う (S600)。ここで、二進化は、Truncated Rice binarization process、Fixed-length binarization processなどの多様な二進化方法に基づいて行われ、対象シンタックス要素に対する二進化方法は予め定義されることができる。二進化手順はエントロピーエンコーディング部240内の二進化部242により行われる。

【0099】

エンコーディング装置は、対象シンタックス要素に対するエントロピーエンコーディングを行う (S610)。エンコーディング装置は、CABAC (context-adaptive arithmetic coding) 又はCAVLC (context-adaptive variable length coding) などのエントロピーコーディング技法に基づいて対象シンタックス要素のビンストリング (bin string) を正規コーディングベース (コンテキストベース) 又はバイパスコーディングベースエンコーディングすることができ、その出力はビットストリームに含まれる。エントロピーエンコーディング手順はエントロピーエンコーディング部240内のエントロピーエンコーディング処理部243により行われる。ビットストリームは (デジタル) 格納媒体又はネットワークを介してデコーディング装置に伝達できることを前述の通りである。

【0100】

図8は、本文書の実施例が適用可能なエントロピーデコーディング方法の一例を概略的に示し、図9は、デコーディング装置内のエントロピーデコーディング部を概略的に示す。図9のデコーディング装置内のエントロピーデコーディング部は前述の図3のデコーディング装置300のエントロピーデコーディング部310にも同一又は対応するように適用できる。

【0101】

図8及び図9に示すように、デコーディング装置 (エントロピーデコーディング部) は

10

20

30

40

50

エンコーディングされた映像/ビデオ情報をデコーディングする。映像/ビデオ情報は、パーティショニング関連情報、予測関連情報（例えば、インター/イントラ予測区分情報、イントラ予測モード情報、インター予測モード情報など）、レジデュアル情報、インループフィルタリング関連情報などを含み、又はそれに関する多様なシンタックス要素を含む。エントロピーコーディングはシンタックス要素単位で行われる。S 8 0 0ないしS 8 1 0は、前述の図3のデコーディング装置3 0 0のエントロピーデコーディング部3 1 0により行われることができる。

【0 1 0 2】

デコーディング装置は、対象シンタックス要素に対する二進化を行う（S 8 0 0）。ここで、二進化はTruncated Rice binarization process、Fixed-length binarization processなどの様々な二進化方法に基づいて行われ、対象シンタックス要素に対する二進化方法は予め定義されることができる。デコーディング装置は、二進化手順により対象シンタックス要素の可用値に対する可用ピンストリング（ピンストリング候補）を導出する。二進化手順は、エントロピーデコーディング部3 1 0内の二進化部3 1 2により行われることができる。

10

【0 1 0 3】

デコーディング装置は、対象シンタックス要素に対するエントロピーデコーディングを行う（S 8 1 0）。デコーディング装置は、ビットストリーム内の入力ビット（ら）から対象シンタックス要素に対する各ピンを順次デコーディング及びパーシングしながら、導出されたピンストリングを当該シンタックス要素に対する可用ピンストリングと比較する。もし、導出されたピンストリングが可用ピンストリングのうち1つと同じであるとすると、そのピンストリングに対応する値が当該シンタックス要素の値として導出されることができる。もし、そうでないとすると、ビットストリーム内の次のビットをさらにパーシングした後、前述の手順を再び行う。このような過程によりビットストリーム内に特定情報（特定シンタックス要素）に対するスタートビットやエンドビットを使わなくても可変長ビットを利用して当該情報をシグナリングすることができる。これにより、低い値に対しては相対的に少ないビットを割り当てることができ、全般的なコーディング効率を高めることができる。

20

【0 1 0 4】

デコーディング装置は、C A B A C又はC A V L Cなどのエントロピーコーディング技法に基づいてビットストリームからピンストリング内の各ピンをコンテキストベース又はバイパスベースデコーディングを行うことができる。ここで、ビットストリームは、前述のように映像/ビデオデコーディングのための様々な情報を含む。ビットストリームは、（デジタル）格納媒体又はネットワークを介してデコーディング装置に伝達できることは前述の通りである。

30

【0 1 0 5】

本文書においてエンコーディング装置からデコーディング装置への情報のシグナリングを示すために、シンタックス要素を含む表（シンタックス表）が使用される。本文書において使用されるシンタックス表のシンタックス要素の順序は、ビットストリームからシンタックス要素のパーシング順序（parsing order）を示す。エンコーディング装置は、シンタックス要素がパーシング順序でデコーディング装置においてパーシングされるようにシンタックス表を構成及びエンコーディングすることができ、デコーディング装置はビットストリームから当該シンタックス表のシンタックス要素をパーシング順序に従ってパーシング及びデコーディングしてシンタックス要素の値を取得することができる。

40

【0 1 0 6】

一方、前述のように、レジデュアルサンプルは、変換、量子化過程を経て量子化された変換係数として導出される。量子化された変換係数は変換係数と呼ばれてもよい。この場合、ブロック内の変換係数はレジデュアル情報の形態でシグナリングされることができる。レジデュアル情報は、レジデュアルコーディングシンタックスを含む。すなわち、エン

50

コーディング装置は、レジデュアル情報でレジデュアルコーディングシンタックスを構成し、これをエンコーディングしてビットストリーム形態に出力し、デコーディング装置は、ビットストリームからレジデュアルコーディングシンタックスをデコーディングしてレジデュアル（量子化された）変換係数を導出する。レジデュアルコーディングシンタックスは、後述のように、当該ブロックに対して変換が適用されているか、ブロック内の最後の有効変換係数の位置がどこであるか、サブブロック内に有効変換係数が存在するか、有効変換係数のサイズ／符号がどうなのかなどを示すシンタックス要素（`syntax elements`）を含む。

【0107】

例えば、（量子化された）変換係数は、レジデュアル情報に含まれた `last_sig__coeff__x__prefix`、`last_sig__coeff__y__prefix`、`last_sig__coeff__x__suffix`、`last_sig__coeff__y__suffix`、`coded__sub__block__flag`、`sig__coeff__flag`、`par__level__flag`、`abs__level__gtX__flag`、`abs__remainder`、`coeff__sign__flag`、`dec__abs__level` などのシンタックス要素（`syntax elements`）に基づいてエンコーディング及び／又はデコーディングされることができる。これは、レジデュアル（データ）コーディング又は（変換）係数コーディングとも呼ばれる。この時、変換／量子化過程は省略されてもよい。この場合、レジデュアルサンプルの値が定めた方法に従ってコーディングされてシグナリングされることができる。レジデュアルデータエンコーディング／デコーディングと関連したシンタックス要素は以下の表1のように示すことができる。

【0108】

10

20

30

40

50

【表 1 - 1】

residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {	Descriptor
if((tu_mts_idx[x0][y0] > 0 (cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6) && cIdx = 0 && log2TbWidth > 4)	
log2ZoTbWidth = 4	
else	
log2ZoTbWidth = Min(log2TbWidth, 5)	
MaxCcbS = 2 * (1 << log2TbWidth) * (1 << log2TbHeight)	
if(tu_mts_idx[x0][y0] > 0 (cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6) && cIdx = 0 && log2TbHeight > 4)	
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	
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--	

10

20

30

40

【 0 1 0 9 】

50

【表 1 - 2】

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))	
if(lastSubBlock == 0 && log2TbWidth >= 2 && log2TbHeight >= 2 && !transform_skip_flag[x0][y0] && lastScanPos > 0)	10
LfnstDcOnly = 0	
if((lastSubBlock > 0 && log2TbWidth >= 2 && log2TbHeight >= 2) (lastScanPos > 7 && (log2TbWidth == 2 log2TbWidth == 3) && log2TbWidth == log2TbHeight))	
LfnstZeroOutSigCoeffFlag = 0	
QState = 0	
for(i = lastSubBlock; i >= 0; i--) {	
startQStateSb = QState	
xS = DiagScanOrder[log2TbWidth - log2SbW][log2TbHeight - log2SbH] [i][0]	
yS = DiagScanOrder[log2TbWidth - log2SbW][log2TbHeight - log2SbH] [i][1]	
inferSbDcSigCoeffFlag = 0	20
if((i < lastSubBlock) && (i > 0)) {	
coded_sub_block_flag [xS][yS]	ac(v)
inferSbDcSigCoeffFlag = 1	
}	
firstSigScanPosSb = numSbCoeff	
lastSigScanPosSb = -1	
firstPosMode0 = (i == lastSubBlock ? lastScanPos : numSbCoeff - 1)	
firstPosMode1 = -1	
for(n = firstPosMode0; n >= 0 && remBinsPass1 >= 4; n--) {	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]	30
if(coded_sub_block_flag[xS][yS] && (n > 0 !inferSbDcSigCoeffFlag) && (xC != LastSignificantCoeffX yC != LastSignificantCoeffY)) {	
sig_coeff_flag [xC][yC]	ac(v)
remBinsPass1--	
if(sig_coeff_flag[xC][yC])	
inferSbDcSigCoeffFlag = 0	
}	
if(sig_coeff_flag[xC][yC]) {	
abs_level_gtx_flag [n][0]	ac(v)
remBinsPass1--	
if(abs_level_gtx_flag[n][0]) {	40

【 0 1 1 0 】

【表 1 - 3】

par_level_flag[n]	ac(v)	
remBinsPass1--		
abs_level_gtx_flag[n][1]	ac(v)	
remBinsPass1--		
}		
if(lastSigScanPosSb == -1)		
lastSigScanPosSb = n		
firstSigScanPosSb = n		
}		
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]		10
if(dep_quant_enabled_flag)		
QState = QStateTransTable[QState][AbsLevelPass1[xC][yC] & 1]		
if(remBinsPass1 < 4)		
firstPosMode1 = n - 1		
}		
for(n = numSbCoeff - 1; n >= firstPosMode1; 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]	ac(v)	
AbsLevel[xC][yC] = AbsLevelPass1[xC][yC] + 2 * abs_remainder[n]		20
}		
for(n = firstPosMode1; n >= 0; n--) {		
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]		
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]		
dec_abs_level[n]	ac(v)	
if(AbsLevel[xC][yC] > 0)		
firstSigScanPosSb = n		
if(dep_quant_enabled_flag)		
QState = QStateTransTable[QState][AbsLevel[xC][yC] & 1]		
}		
if(dep_quant_enabled_flag !sign_data_hiding_enabled_flag)		
signHidden = 0		30
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]	ac(v)	
}		
if(dep_quant_enabled_flag) {		
QState = startQStateSb		
for(n = numSbCoeff - 1; n >= 0; n--) {		40

【 0 1 1 1 】

【表 1 - 4】

<code>xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]</code>	
<code>yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]</code>	
<code>if(AbsLevel[xC][yC] > 0)</code>	
<code>TransCoeffLevel[x0][y0][cIdx][xC][yC] =</code> <code>(2 * AbsLevel[xC][yC] - (QState > 1 ? 1 : 0)) *</code> <code>(1 - 2 * coeff_sign_flag[n])</code>	
<code>QState = QStateTransTable[QState][par_level_flag[n]]</code>	
<code>} else {</code>	
<code>sumAbsLevel = 0</code>	
<code>for(n = numSbCoeff - 1; n >= 0; n--) {</code>	10
<code>xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][n][0]</code>	
<code>yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][n][1]</code>	
<code>if(AbsLevel[xC][yC] > 0) {</code>	
<code>TransCoeffLevel[x0][y0][cIdx][xC][yC] =</code> <code>AbsLevel[xC][yC] * (1 - 2 * coeff_sign_flag[n])</code>	
<code>if(signHidden) {</code>	
<code>sumAbsLevel += AbsLevel[xC][yC]</code>	
<code>if((n == firstSigScanPosSb) && (sumAbsLevel % 2) == 1))</code>	
<code>TransCoeffLevel[x0][y0][cIdx][xC][yC] =</code> <code>-TransCoeffLevel[x0][y0][cIdx][xC][yC]</code>	20
<code>}</code>	
<code>}</code>	
<code>}</code>	
<code>}</code>	
<code>}</code>	
<code>}</code>	

【 0 1 1 2】

表1を参照すると、last_sig_coeff_x_prefix、last_sig_coeff_y_prefix、last_sig_coeff_x_suffix、last_sig_coeff_y_suffixは、関連したブロック(associated block)内の最後の非ゼロ係数の(x, y)位置情報をエンコーディングするシンタックス要素である。関連したブロックはCB(coding block)又はTB(Transform block)であり得る。変換(及び量子化)及びレジデュアルコーディング手順に関して、CBとTBは混用されてもよい。例えば、CBに対してレジデュアルサンプルが導出され、レジデュアルサンプルに対する変換及び量子化により(量子化された)変換係数が導出できることは前述の通りであり、レジデュアルコーディング手順により(量子化された)変換係数(の位置、サイズ、符号など)を効率的に示す情報(又は、シンタックス要素)が生成及びシグナリングされることが出来る。量子化された変換係数は簡単に変換係数と呼ばれてもよい。一般に、CBが最大TBより大きくない場合、CBのサイズはTBのサイズと同じことがあり、この場合、変換(及び量子化)及びレジデュアルコーディングされる対象ブロックはCBと呼ばれてもよく、TBと呼ばれてもよい。または、CBが最大TBより大きい場合は、変換(及び量子化)及びレジデュアルコーディングされる対象ブロックはTBと呼ばれてもよい。以下、レジデュアルコーディングに関連したシンタックス要素が変換ブロック(TB)単位でシグナリングされることと説明するが、これは例示であり、TBはコーディングブロック(CB)と混用できることは前述の通りである。

【 0 1 1 3】

一方、レジデュアルコーディングのために変換スキップが適用されるか否かに応じて相異なるレジデュアルコーディング方式が適用される。一実施例として、変換スキップが適

10

20

30

40

50

用されるか否かは変換スキップフラグシンタックス要素を使用して示し、変換スキップフラグのシンタックス要素 `transform_skip_flag` の値によってレジデュアルコーディングが分岐されることが出来る。すなわち、変換スキップフラグの値に基づいて（変換スキップが適用されるか否かに基づいて）レジデュアルコーディングのために相異なるシンタックス要素が使用される。変換スキップが適用されない場合（すなわち、変換が適用された場合）に使用されるレジデュアルコーディングは、レギュラーレジデュアルコーディング（Regular Residual Coding、RRC）と呼ばれ、変換スキップが適用されない場合（すなわち、変換が適用されない場合）のレジデュアルコーディングは、変換スキップレジデュアルコーディング（Transform Skip Residual Coding、TSRC）と呼ばれる。

10

【0114】

次の表2は、変換スキップフラグのシンタックス要素に基づいてレジデュアルコーディングが分岐される過程を示す。

【0115】

【表2】

transform_unit(x0, y0, tbWidth, tbHeight, treeType, subTuIndex, chType) {	Descriptor
transform_skip_flag [x0][y0][0]	ae(v)
if(!transform_skip_flag[x0][y0][0] slice_ts_residual_coding_disabled_flag)	
residual_coding(x0, y0, Log2(tbWidth), Log2(tbHeight), 0)	
else	
residual_ts_coding(x0, y0, Log2(tbWidth), Log2(tbHeight), 0)	
}	
if(tu_cbf_cb[xC][yC] && treeType != DUAL_TREE_LUMA) {	
if(sps_transform_skip_enabled_flag && !BdpcmFlag[x0][y0][1] && wC <= MaxTsSize && hC <= MaxTsSize && !cu_sbt_flag)	
transform_skip_flag [xC][yC][1]	ae(v)
if(!transform_skip_flag[xC][yC][1] slice_ts_residual_coding_disabled_flag)	
residual_coding(xC, yC, Log2(wC), Log2(hC), 1)	
else	
residual_ts_coding(xC, yC, Log2(wC), Log2(hC), 1)	
}	
if(tu_cbf_cr[xC][yC] && treeType != DUAL_TREE_LUMA && !(tu_cbf_cb[xC][yC] && tu_joint_cbr_residual_flag[xC][yC])) {	
if(sps_transform_skip_enabled_flag && !BdpcmFlag[x0][y0][2] && wC <= MaxTsSize && hC <= MaxTsSize && !cu_sbt_flag)	
transform_skip_flag [xC][yC][2]	ae(v)
if(!transform_skip_flag[xC][yC][2] slice_ts_residual_coding_disabled_flag)	
residual_coding(xC, yC, Log2(wC), Log2(hC), 2)	
else	
residual_ts_coding(xC, yC, Log2(wC), Log2(hC), 2)	
}	
}	

20

30

【0116】

前記表2を参照すると、変換スキップが適用されない場合（例：`transform_skip_flag`の値が0である場合）はレギュラーレジデュアルコーディングが行われ、これは前述の表1に開示されたシンタックス要素に基づいて行われる。または、変換スキップが適用される場合（例：`transform_skip_flag`の値が1である場合）は変換スキップレジデュアルコーディングが行われ、これは次の表3に開示されたシンタックス要素に基づいて行われる。

40

【0117】

次の表3は、変換スキップレジデュアルコーディングのためのシンタックス要素を示す。

【0118】

50

【表 3 - 1】

	Descriptor	
residual_ts_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {		
log2SbSize = (Min(log2TbWidth, log2TbHeight) < 2 ? 1 : 2)		
numSbCoeff = 1 << (log2SbSize << 1)		
lastSubBlock = (1 << (log2TbWidth + log2TbHeight - 2 * log2SbSize)) - 1		
inferSbCbf = 1		
MaxCbs = 2 * (1 << log2TbWidth) * (1 << log2TbHeight)		
for(i = 0; i <= lastSubBlock; i++) {		
xS = DiagScanOrder[log2TbWidth-log2SbSize][log2TbHeight-log2SbSize][i][0]		
yS = DiagScanOrder[log2TbWidth-log2SbSize][log2TbHeight-log2SbSize][i][1]		
if((i != lastSubBlock !inferSbCbf) {		10
coded_sub_block_flag [xS][yS]	ae(v)	
}		
if(coded_sub_block_flag[xS][yS] && i < lastSubBlock)		
inferSbCbf = 0		
/* First scan pass */		
inferSbSigCoeffFlag = 1		
for(n = 0; n <= numSbCoeff - 1; n++) {		
xC = (xS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][0]		
yC = (yS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][1]		
if(coded_sub_block_flag[xS][yS] &&		
(n != numSbCoeff - 1 !inferSbSigCoeffFlag)) {		20
sig_coeff_flag [xC][yC]	ae(v)	
MaxCbs--		
if(sig_coeff_flag[xC][yC])		
inferSbSigCoeffFlag = 0		
}		
CoeffSignLevel[xC][yC] = 0		
if(sig_coeff_flag[xC][yC] {		
coeff_sign_flag [n]	ae(v)	
MaxCbs--		
CoeffSignLevel[xC][yC] = (coeff_sign_flag[n] > 0 ? -1 : 1)		
abs_level_gtx_flag [n][0]	ae(v)	
MaxCbs--		
if(abs_level_gtx_flag[n][0]) {		30
par_level_flag [n]	ae(v)	
MaxCbs--		
}		
}		
AbsLevelPassX[xC][yC] = sig_coeff_flag[xC][yC] + par_level_flag[n] +		
abs_level_gtx_flag[n][0]		
}		
/* Greater than X scan pass (numGtXFlags=5) */		
for(n = 0; n <= numSbCoeff - 1; n++) {		
xC = (xS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][0]		
yC = (yS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][1]		
for(j = 1; j < 5; j++) {		40

【 0 1 1 9 】

【表 3 - 2】

if(abs_level_gtx_flag[n][j - 1])	
abs_level_gtx_flag[n][j]	ae(v)
MaxCcbs--	
AbsLevelPassX[xC][yC] += 2 * abs_level_gtx_flag[n][j]	
}	
}	
/* remainder scan pass */	
for(n = 0; n <= numSbCoeff - 1; n++) {	
xC = (xS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][0]	
yC = (yS << log2SbSize) + DiagScanOrder[log2SbSize][log2SbSize][n][1]	
if(abs_level_gtx_flag[n][4])	
abs_remainder[n]	ae(v)
if(intra_bdpcm_flag == 0) {	
absRightCoeff = abs(TransCoeffLevel[x0][y0][cIdx][xC - 1][yC])	
absBelowCoeff = abs(TransCoeffLevel[x0][y0][cIdx][xC][yC - 1])	
predCoeff = Max(absRightCoeff, absBelowCoeff)	
if(AbsLevelPassX[xC][yC] + abs_remainder[n] == 1 && predCoeff >	
0)	
TransCoeffLevel[x0][y0][cIdx][xC][yC] =	
(1 - 2 * coeff_sign_flag[n]) * predCoeff	
else if(AbsLevelPassX[xC][yC] + abs_remainder[n] <= predCoeff)	
TransCoeffLevel[x0][y0][cIdx][xC][yC] = (1 - 2 *	
coeff_sign_flag[n]) *	
(AbsLevelPassX[xC][yC] + abs_remainder[n] - 1)	
else	
TransCoeffLevel[x0][y0][cIdx][xC][yC] = (1 - 2 *	
coeff_sign_flag[n]) *	
(AbsLevelPassX[xC][yC] + abs_remainder[n])	
} else	
TransCoeffLevel[x0][y0][cIdx][xC][yC] = (1 - 2 *	
coeff_sign_flag[n]) *	
(AbsLevelPassX[xC][yC] + abs_remainder[n])	
}	
}	
}	
}	

10

20

30

【 0 1 2 0 】

例えば、変換ブロックの変換スキップの可否を指示する変換スキップフラグがパーズングされ、変換スキップフラグが1であるか否かが判断される。変換スキップフラグの値が1である場合、表3に示すように、変換ブロックのレジデュアル係数に関するシンタックス要素 sig_coeff_flag、coeff_sign_flag、abs_level_gtx_flag、par_level_flag 及び/又は abs_remainder がパーズングされ、シンタックス要素に基づいてレジデュアル係数が導出される。この場合、前記シンタックス要素は順次パーズングされてもよく、パーズング順序が変更されてもよい。ここで、abs_level_gtx_flag は abs_level_gtx1_flag、abs_level_gtx3_flag、abs_level_gtx5_flag、abs_level_gtx7_flag 及び/又は abs_level_gtx9_flag を示すことができる。例えば、abs_level_gtx_flag[n][j] はスキャン位置 n において変換係数レベル（又は、変換係数レベルを右側に1だけシフトした値）の絶対値が (j < 1) + 1 より大きいかが否かを示すフラグであり得る。前記 (j < 1) + 1 は、場合によっては、第1臨界値、第2臨界値などの所定の臨界値に代替されることもできる。

40

【 0 1 2 1 】

また、変換スキップフラグの値が0である場合、表1に示すように、変換ブロックのレ

50

レジデュアル係数に関するシンタックス要素 `sig__coeff__flag`、`abs__level__gtx__flag`、`par__level__flag`、`abs__remainder`、`dec__abs__level`、`coeff__sign__flag` がパーシングされ、前記シンタックス要素に基づいてレジデュアル係数が導出される。この場合、前記シンタックス要素は順次パーシングされてもよく、パーシング順序が変更されてもよい。ここで、`abs__level__gtx__flag` は `abs__level__gt1__flag` 及び / 又は `abs__level__gt3__flag` を示すことができる。

【0122】

一方、前述のように、エンコーディング装置は、イントラ/インター/IBC/パレット予測などにより予測されたブロック（予測サンプル）に基づいてレジデュアルブロック（レジデュアルサンプル）を導出し、導出されたレジデュアルサンプルに変換及び量子化を適用して量子化された変換係数を導出する。量子化された変換係数に関する情報（レジデュアル情報）はレジデュアルコーディングシンタックスに含まれてエンコーディング後にビットストリーム形態で出力される。デコーディング装置は、前記ビットストリームから前記量子化された変換係数に関する情報（レジデュアル情報）を取得し、デコーディングして量子化された変換係数を導出する。デコーディング装置は、量子化された変換係数に基づいて逆量子化/逆変換を経てレジデュアルサンプルを導出する。前述のように、前記量子化/逆量子化及び/又は変換/逆変換の少なくとも1つは省略されてもよい。前記変換/逆変換が省略される場合、前記変換係数は係数又はレジデュアル係数と呼ばれてもよく、又は表現の統一性のために変換係数と依然として呼ばれてもよい。前記変換/逆変換が省略されるか否かは、`transform__skip__flag` に基づいてシグナリングされる。例えば、`transform__skip__flag` の値が1である場合は前記変換/逆変換が省略されることを示し、これを変換スキップモードという。

【0123】

一般的に、ビデオ/映像コーディングにおいては量子化率を変化させることができ、変換された量子化率を利用して圧縮率を調節することができる。実現の観点からは、複雑度を考慮して量子化率を直接使用する代わりに量子化パラメータ（`quantization__parameter`、QP）を使用できる。例えば、0から63までの整数値の量子化パラメータが使用されることができ、各量子化パラメータ値は実際の量子化率に対応できる。また、例えば、ルマ成分（ルマサンプル）に対する量子化パラメータ（QP_Y）とクロマ成分（クロマサンプル）に対する量子化パラメータ（QP_C）は異なる設定が可能である。

【0124】

量子化過程は変換係数（C）を入力とし、量子化率（Q_{step}）で割って、これに基づいて量子化された変換係数（C'）が得られる。この場合、計算複雑度を考慮して量子化率にスケールを掛け算して整数形態にし、スケール値に該当する値の分だけシフト演算を行うことができる。量子化率とスケール値の積に基づいて量子化スケール（`quantization__scale`）が導出できる。すなわち、QPによって前記量子化スケールが導出される。例えば、前記変換係数（C）に前記量子化スケールを適用して、これに基づいて量子化された変換係数（C'）が導出されることもできる。

【0125】

逆量子化過程は量子化過程の逆過程であり、量子化された変換係数（C'）に量子化率（Q_{step}）を掛け算して、これに基づいて復元された変換係数（C''）が得られる。この場合、前記量子化パラメータによってレベルスケール（`level__scale`）が導出され、前記量子化された変換係数（C'）に前記レベルスケールを適用して、これに基づいて復元された変換係数（C''）が導出される。復元された変換係数（C''）は、変換及び/又は量子化過程での損失（loss）により最初変換係数（C）と多少差がある。従って、エンコーディング装置においてもデコーディング装置と同様に逆量子化を行う。

【0126】

また、予測の実行においてパレットコーディング（`palette__coding`）に

基づいてもよい。パレットコーディングは少ない数の固有色の値を含むブロックを示すために有用な技術である。ブロックに対して予測及び変換を適用する代わりに、パレットモードは各サンプルの値を示すためにインデックスをシグナリングする。パレットモードはビデオメモリバッファ空間の節約に有用である。パレットモード（例えば、MODE_PLT）を使用してブロックがコーディングされることができる。このようにエンコーディングされたブロックをデコーディングするために、デコーダはパレットエントリ及びインデックスをデコーディングしなければならない。パレットエントリは、パレットテーブルにより示され、パレットテーブルコーディングツールによりエンコーディングされる。

【0127】

パレットコーディングは、（イントラ）パレットモード又は（イントラ）パレットコーディングモードなどと呼ばれる。パレットコーディング又はパレットモードによって現在ブロックが復元されることができる。パレットコーディングはイントラコーディングの一例であり、又はイントラ予測方法の1つでもある。ただし、前述のスキップモードと類似するように、当該ブロックに対する別途のレジデュアル値はシグナリングされないことがある。

10

【0128】

例えば、パレットモードが選択された場合、パレットテーブルに関する情報がシグナリングされることができる。パレットテーブルはそれぞれのピクセルに対応するインデックスを含む。パレットテーブルは、以前のブロックにおいて使用されていたピクセル値からパレット予測テーブルを構成することができる。例えば、以前に使われていたピクセル値が特定のバッファに格納されていて（パレット予測子）、このバッファから現在パレットを構成するためのパレット予測子情報（palette_predictor_run）が受信される。すなわち、パレット予測子は現在ブロックのパレットインデックスマップ（palette_index_map）の少なくとも一部に対するインデックスを示すデータを含む。現在ブロックを表現するためのパレットエントリがパレット予測子から構成されたパレット予測エントリで十分でない場合、現在パレットエントリに関するピクセル情報が別途に送信される。

20

【0129】

パレットモードは、CUレベルでシグナリングされ、一般的にCU内の大部分のピクセルが代表ピクセル値のセットとして示されることができる場合に使用できる。すなわち、パレットモードはCU内のサンプルが代表ピクセル値のセットとして表現されることができる。このようなセットはパレットと呼ばれてもよい。パレット内のピクセル値に近い値を有するサンプルの場合、パレット内のピクセル値に対応するパレットインデックス（palette_idx_idx）又はインデックスを指す情報（run_copy_flag、copy_above_indices_flag）がシグナリングされることができる。パレットエントリ以外のピクセル値を有するサンプルの場合、そのサンプルはエスケープシンボル（escape_symbol）と表示され、量子化されたサンプル値が直接的にシグナリングされることができる。本文書において、ピクセル又はピクセル値はサンプル又はサンプル値として称されてもよい。

30

【0130】

パレットモードでコーディングされたブロックをデコーディングするために、デコーダはパレットエントリ情報及びパレットインデックス情報が必要である。パレットインデックスがエスケープシンボルに該当する場合、追加構成要素（component）に（量子化された）エスケープ値がシグナリングされる。また、エンコーダにおいては当該CUに対する適切なパレットを導出してデコーダに伝達しなければならない。

40

【0131】

パレットエントリの効率的なコーディングのために、パレット予測子（palette_predictor）が維持される。パレット予測子及びパレットの最大サイズがSPSにおいてシグナリングされることができる。または、パレット予測子及びパレット最大サイズが既に定義されることができる。例えば、パレット予測子及びパレット最大サイズは

50

、現在ブロックがシングルツリーであるかデュアルツリーであるかに応じてそれぞれ31、15と定義される。VVC標準においてはパレットモードが可用であるか否かを示す `sp_s_palette_enabled_flag` が送信されることができ。そうすると、現在コーディングユニットがパレットモードでコーディングされたものであるか否かを示す `pred_mode_plt_coding` フラグが送信される。パレット予測子は、各ブリック (`brick`) 又は各スライスの開始部分において初期化される。

【0132】

パレット予測子内の各エントリに対して、再使用フラグ (`reuse_flag`) がシグナリングされて現在パレットの一部であるか否かを示すことができる。再使用フラグは0の `run-length` コーディングを使用して送信できる。その後、新しいパレットエントリの個数は、 0^{th} orderの指数ゴロム (`exponential Golomb`) コーディングを使ってシグナリングされることができ。最後に、新しいパレットエントリに対するコンポーネント値 (`component_value`) がシグナリングされることができ。現在CUをエンコーディングした後、パレット予測子は現在パレットを使ってアップデートされ、現在パレットにおいて再使用されていない以前パレット予測子のエントリは、許容される最大サイズに到達するまで新しいパレット予測子の最後に追加されることができ (`palette_stuffing`、パレットスタッフィング)。

【0133】

パレットインデックスマップをコーディングするために水平及び垂直トラバーススキャン (`traverse_scans`) を使用してインデックスをコーディングすることができる。スキャン順序 (`scan_order`) はフラグ情報 (例えば、`palette_transpose_flag`) を使ってビットストリームから明示的にシグナリングされることができ。

【0134】

一方、パレットインデックスは2種類のパレットサンプルモードを使ってコーディングでされるが、例えば、「INDEX」モード及び「COPY_ABOVE」モードを使用することができる。このようなパレットモードは、「INDEX」モードであるか「COPY_ABOVE」モードであるかを示すフラグを使用してシグナリングされることができ。この時、エスケープシンボルは「INDEX」モードでシグナリングされ、現在パレットサイズと同じインデックスが割り当てられることができる。例えば、現在パレットのサイズが10であると、インデックス0番から9番まではパレット内のエントリインデックスを意味し、10番インデックスはエスケープシンボルに対するインデックスを意味する。水平スキャンが使用される場合は、上端行 (`top_row`) を除いてフラグをシグナリングすることができ、垂直スキャンが使用される場合又は以前モードが「COPY_ABOVE」モードである場合は、1番目の列 (`first_column`) を除いてフラグをシグナリングすることができる。「COPY_ABOVE」モードでは、上にある行のサンプルのパレットインデックスがコピーされる。「INDEX」モードでは、パレットインデックスが明示的にシグナリングされる。「INDEX」モード及び「COPY_ABOVE」モードの両方ともに対して、同一のモードを使用してコーディングされる次のサンプルの個数を示すラン値がシグナリングされる。エスケープシンボルが「INDEX」モード又は「COPY_ABOVE」モードにおいてランの一部である場合、エスケープコンポーネント値が各エスケープシンボルに対してシグナリングされることができ。

【0135】

パレットインデックスに対するコーディングは以下のようなものである。まず、CUに対するインデックスの数がシグナリングされる。その次に、固定長コーディング (`fixed_length_coding`) を使用して全体CUに対する実際のインデックスがシグナリングされる。インデックス数及びインデックスがバイパスモードでコーディングされる。これはインデックス関連バイパスピンと一緒にグループ化されることができ。次に、

パレットサンプルモード (`copy__above__palette__indices__flag`) 及びランがインターリーブ方式 (`interleaved manner`) でシグナリングされる。最後に、全体CUに対するエスケープサンプルに対応するコンポーネントエスケープ値と一緒にグループ化され、バイパスモードでコーディングされる。

【0136】

次の表4は、コーディングユニットに対するパレットモードベースコーディングに関連するシンタックス要素を含むシンタックス構造の一例を示し、次の表5は、表4のシンタックス要素 (`syntax elements`) に対するセマンティックス (`semantics`) を示す。

【0137】

10

20

30

40

50

【表 4 - 1】

palette_coding(x0, y0, cbWidth, cbHeight, treeType) {	Descriptor
startComp = (treeType == DUAL_TREE_CHROMA) ? 1 : 0	
numComps = (treeType == SINGLE_TREE) ? (sps_chroma_format_idc == 0 ? 1 : 3) : (treeType == DUAL_TREE_CHROMA) ? 2 : 1	
maxNumPaletteEntries = (treeType == SINGLE_TREE) ? 31 : 15	
palettePredictionFinished = 0	
NumPredictedPaletteEntries = 0	
for(predictorEntryIdx = 0; predictorEntryIdx < PredictorPaletteSize[startComp] && !palettePredictionFinished && NumPredictedPaletteEntries < maxNumPaletteEntries; predictorEntryIdx++) {	
palette_predictor_run	ae(v)
if(palette_predictor_run != 1) {	
if(palette_predictor_run > 1)	
predictorEntryIdx += palette_predictor_run - 1	
PalettePredictorEntryReuseFlags[predictorEntryIdx] = 1	
NumPredictedPaletteEntries++	
} else	
palettePredictionFinished = 1	
}	
if(NumPredictedPaletteEntries < maxNumPaletteEntries)	
num_signalled_palette_entries	ae(v)
for(cIdx = startComp; cIdx < (startComp + numComps); cIdx++)	
for(i = 0; i < num_signalled_palette_entries; i++)	
new_palette_entries[cIdx][i]	ae(v)
if(CurrentPaletteSize[startComp] > 0)	
palette_escape_val_present_flag	ae(v)
if(MaxPaletteIndex > 0) {	
adjust = 0	
palette_transpose_flag	ae(v)
}	
if(treeType != DUAL_TREE_CHROMA && palette_escape_val_present_flag)	
if(pps_cu_qp_delta_enabled_flag && !IsCuQpDeltaCoded) {	
cu_qp_delta_abs	ae(v)
if(cu_qp_delta_abs)	
cu_qp_delta_sign_flag	ae(v)
}	
if(treeType != DUAL_TREE_LUMA && palette_escape_val_present_flag)	
if(sh_cu_chroma_qp_offset_enabled_flag && !IsCuChromaQpOffsetCoded) {	
cu_chroma_qp_offset_flag	ae(v)
if(cu_chroma_qp_offset_flag && pps_chroma_qp_offset_list_len_minus1 > 0)	
cu_chroma_qp_offset_idx	ae(v)
}	
PreviousRunPosition = 0	
PreviousRunType = 0	
for(subSetId = 0; subSetId <= (cbWidth * cbHeight - 1) / 16; subSetId++) {	
minSubPos = subSetId * 16	

10

20

30

40

【 0 1 3 8 】

50

【表 4 - 2】

if(minSubPos + 16 > cbWidth * cbHeight)	
maxSubPos = cbWidth * cbHeight	
else	
maxSubPos = minSubPos + 16	
RunCopyMap[x0][y0] = 0	
PaletteScanPos = minSubPos	
log2CbWidth = Log2(cbWidth)	
log2CbHeight = Log2(cbHeight)	
while(PaletteScanPos < maxSubPos) {	
xC = x0+TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos][0]	10
yC = y0+TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos][1]	
if(PaletteScanPos > 0) {	
xcPrev = x0 +	
TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos-1][0]	
ycPrev = y0 +	
TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos-1][1]	
}	
if(MaxPaletteIndex > 0 && PaletteScanPos > 0) {	
run_copy_flag	ae(v)
RunCopyMap[xC][yC] = run_copy_flag	
}	
CopyAboveIndicesFlag[xC][yC] = 0	20
if(MaxPaletteIndex > 0 && !RunCopyMap[xC][yC]) {	
if((!palette_transpose_flag && yC > y0)	
(palette_transpose_flag && xC > x0)) &&	
CopyAboveIndicesFlag[xcPrev][ycPrev] == 0 &&	
PaletteScanPos > 0) {	
copy_above_palette_indices_flag	ae(v)
CopyAboveIndicesFlag[xC][yC] = copy_above_palette_indices_flag	
}	
PreviousRunType = CopyAboveIndicesFlag[xC][yC]	
PreviousRunPosition = PaletteScanPos	
} else if(PaletteScanPos > 0)	30
CopyAboveIndicesFlag[xC][yC] = CopyAboveIndicesFlag[xcPrev][ycPrev]	
PaletteScanPos ++	
}	
PaletteScanPos = minSubPos	
while(PaletteScanPos < maxSubPos) {	
xC = x0 +	
TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos][0]	
yC = y0 +	
TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos][1]	
if(PaletteScanPos > 0) {	
xcPrev =x0 +	
TraverseScanOrder[log2CbWidth][log2CbHeight][PaletteScanPos-1][0]	40

【 0 1 3 9 】

【表 5 - 1】

<p>In the following semantics, the array indices x_0, y_0 specify the location (x_0, y_0) of the top-left luma sample of the considered coding block relative to the top-left luma sample of the picture. The array indices x_C, y_C specify the location (x_C, y_C) of the sample relative to the top-left luma sample of the picture, when <code>treeType</code> is equal to <code>SINGLE_TREE</code> or <code>DUAL_TREE_LUMA</code>; and relative to the top-left chroma sample of the picture, when <code>treeType</code> is equal to <code>DUAL_TREE_CHROMA</code>. The array index <code>startComp</code> specifies the first colour component of the current palette table. <code>startComp</code> equal to 0 indicates the Y component; <code>startComp</code> equal to 1 indicates the Cb component; <code>startComp</code> equal to 2 indicates the Cr component. <code>numComps</code> specifies the number of colour components in the current palette table.</p>	10
<p>The predictor palette consists of palette entries from previous coding units that are used to predict the entries in the current palette.</p>	
<p><code>PredictorPaletteSize[startComp]</code> specifies the size of the predictor palette for the first colour component of the current palette table <code>startComp</code>. <code>PredictorPaletteSize[startComp]</code> is derived as specified in subclause 8.4.5.3.</p>	20
<p><code>PalettePredictorEntryReuseFlags[i]</code> equal to 1 specifies that the i-th entry in the predictor palette is reused in the current palette. <code>PalettePredictorEntryReuseFlags[i]</code> equal to 0 specifies that the i-th entry in the predictor palette is not an entry in the current palette. All elements of the array <code>PalettePredictorEntryReuseFlags[i]</code> are initialized to 0.</p>	
<p>palette_predictor_run is used to determine the number of zeros that precede a non-zero entry in the array <code>PalettePredictorEntryReuseFlags</code>.</p>	30
<p>It is a requirement of bitstream conformance that the value of <code>palette_predictor_run</code> shall be in the range of 0 to (<code>PredictorPaletteSize[startComp] - predictorEntryIdx</code>), inclusive, where <code>predictorEntryIdx</code> corresponds to the current position in the array <code>PalettePredictorEntryReuseFlags</code>. The variable <code>NumPredictedPaletteEntries</code> specifies the number of entries in the current palette that are reused from the predictor palette. The value of <code>NumPredictedPaletteEntries</code> shall be in the range of 0 to <code>maxNumPaletteEntries</code>, inclusive.</p>	40

【 0 1 4 1 】

【表 5 - 2】

num_signalled_palette_entries specifies the number of entries in the current palette that are explicitly signalled for the first colour component of the current palette table startComp.

When num_signalled_palette_entries is not present, it is inferred to be equal to 0.

The variable CurrentPaletteSize[startComp] specifies the size of the current palette for the first colour component of the current palette table startComp and is derived as follows:

$$\text{CurrentPaletteSize[startComp]} = \text{NumPredictedPaletteEntries} + \text{num_signalled_palette_entries} \quad (176)$$

The value of CurrentPaletteSize[startComp] shall be in the range of 0 to maxNumPaletteEntries, inclusive.

new_palette_entries[cIdx][i] specifies the value for the i-th signalled palette entry for the colour component cIdx.

The variable LocalDualTreeFlag is derived as follows:

$$\text{LocalDualTreeFlag} = (\text{treeType} \neq \text{SINGLE_TREE} \ \&\& (\text{sh_slice_type} \neq \text{I} \ || \ (\text{sh_slice_type} = \text{I} \ \&\& \text{sps_qtbtt_dual_tree_intra_flag} = 0))) ? 1 : 0 \quad (177)$$

The variable PredictorPaletteEntries[cIdx][i] specifies the i-th element in the predictor palette for the colour component cIdx.

The variable CurrentPaletteEntries[cIdx][i] specifies the i-th element in the current palette for the colour component cIdx and is derived as follows:

```

numPredictedPaletteEntries = 0
for( i = 0; i < PredictorPaletteSize[ startComp ]; i++ )
    if( PalettePredictorEntryReuseFlags[ i ] ) {
        for( cIdx = LocalDualTreeFlag ? 0 : startComp;
            cIdx < LocalDualTreeFlag ? 3 : ( startComp + numComps ); cIdx++ )
            CurrentPaletteEntries[ cIdx ][ numPredictedPaletteEntries ] =
                PredictorPaletteEntries[ cIdx ][ i ]
    }

```

【 0 1 4 2 】

10

20

30

40

50

【表 5 - 3】

<pre> numPredictedPaletteEntries++ } for(cIdx = startComp; cIdx < (startComp + numComps); cIdx++) (178) for(i = 0; i < num_signalled_palette_entries; i++) CurrentPaletteEntries[cIdx][numPredictedPaletteEntries + i] = new_palette_entries[cIdx][i] </pre>	10
<p>palette_escape_val_present_flag equal to 1 specifies that the current coding unit contains at least one escape coded sample. palette_escape_val_present_flag equal to 0 specifies that there are no escape coded samples in the current coding unit. When not present, the value of palette_escape_val_present_flag is inferred to be equal to 1.</p> <p>The variable MaxPaletteIndex specifies the maximum possible value for a palette index for the current coding unit. The value of MaxPaletteIndex is set equal to CurrentPaletteSize[startComp] - 1 + palette_escape_val_present_flag.</p>	20
<p>palette_idx_idc is an indication of an index to the palette table, CurrentPaletteEntries. The value of palette_idx_idc shall be in the range of 0 to MaxPaletteIndex, inclusive, for the first index in the block and in the range of 0 to (MaxPaletteIndex - 1), inclusive, for the remaining indices in the block.</p> <p>When palette_idx_idc is not present, it is inferred to be equal to 0.</p>	30
<p>palette_transpose_flag equal to 1 specifies that vertical traverse scan is applied for scanning the indices for samples in the current coding unit. palette_transpose_flag equal to 0 specifies that horizontal traverse scan is applied for scanning the indices for samples in the current coding unit. When not present, the value of palette_transpose_flag is inferred to be equal to 0.</p> <p>The array TraverseScanOrder specifies the scan order array for palette coding. If palette_transpose_flag is equal to 0, TraverseScanOrder is assigned the horizontal scan order HorTravScanOrder. Otherwise (palette_transpose_flag is equal to 1), TraverseScanOrder is assigned the vertical scan order VerTravScanOrder.</p>	40
<p>run_copy_flag equal to 1 specifies that the palette run type is the same as the run type at the previously scanned position and palette index is the same as the index at the previous scanned position if CopyAboveIndicesFlag[xC][yC] is equal to 0.</p>	50

【 0 1 4 3 】

【表 5 - 4】

Otherwise, `run_copy_flag` equal to 0 specifies that the palette run type is different from the run type at the previously scanned position.

`copy_above_palette_indices_flag` equal to 1 specifies that the palette index is equal to the palette index at the same location in the row above if horizontal traverse scan is used or the same location in the left column if vertical traverse scan is used. `copy_above_palette_indices_flag` equal to 0 specifies that an indication of the palette index of the sample is coded in the bitstream or inferred.

10

The variable `CopyAboveIndicesFlag[xC][yC]` equal to 1 specifies that the palette index is copied from the palette index in the row above (horizontal scan) or left column (vertical scan). `CopyAboveIndicesFlag[xC][yC]` equal to 0 specifies that the palette index is explicitly coded in the bitstream or inferred.

The variable `PaletteIndexMap[xC][yC]` specifies a palette index, which is an index to the array represented by `CurrentPaletteEntries`. The value of `PaletteIndexMap[xC][yC]` shall be in the range of 0 to `MaxPaletteIndex`, inclusive.

20

The variable `adjustedRefPaletteIndex` is derived as follows:

```

adjustedRefPaletteIndex = MaxPaletteIndex + 1
if( PaletteScanPos > 0 ) {
    xcPrev = x0 +
        TraverseScanOrder[log2CbWidth][log2bHeight][PaletteScanPos-1][0]
    ycPrev = y0 +
        TraverseScanOrder[log2CbWidth][log2bHeight][PaletteScanPos-1][1]
    if( CopyAboveIndicesFlag[ xcPrev ][ ycPrev ] == 0 )
        adjustedRefPaletteIndex = PaletteIndexMap[xcPrev][ycPrev]    (179)
    else {
        if( !palette_transpose_flag )
            adjustedRefPaletteIndex = PaletteIndexMap[ xC ][ yC - 1 ]
        else
            adjustedRefPaletteIndex = PaletteIndexMap[ xC - 1 ][ yC ]
    }
}

```

30

When `CopyAboveIndicesFlag[xC][yC]` is equal to 0, the variable `CurrPaletteIndex`

40

【 0 1 4 4 】

【表 5 - 5】

is derived as follows:

$$\text{if(CurrPaletteIndex } \geq \text{ adjustedRefPaletteIndex)}$$

$$\quad \text{CurrPaletteIndex++} \quad (180)$$

palette_escape_val specifies the quantized escape coded sample value for a component.

The variable `PaletteEscapeVal[cIdx][xC][yC]` specifies the escape value of a sample for which `PaletteIndexMap[xC][yC]` is equal to `MaxPaletteIndex` and `palette_escape_val_present_flag` is equal to 1. The array index `cIdx` specifies the colour component.

It is a requirement of bitstream conformance that `PaletteEscapeVal[cIdx][xC][yC]` shall be in the range of 0 to $(1 \ll \text{BitDepth}) - 1$, inclusive.

【0145】

前記表 4 及び表 5 を参照すると、現在ブロック（すなわち、現在コーディングユニット）に対してパレットモードが適用される場合、前記表 4 に示すようなパレットコーディングシンタックス（例：`palette_coding()`）がパーシング/シグナリングされる。

【0146】

例えば、パレットエントリ情報に基づいてパレットテーブルを構成することができる。パレットエントリ情報は、`palette_predictor_run`、`num_signalled_palette_entries`、`new_palette_entries`などのシンタックス要素を含む。

【0147】

また、パレットインデックス情報に基づいて、現在ブロックに対するパレットインデックスマップを構成することができる。パレットインデックス情報は、`num_palette_indices_minus_1`、`palette_idx_idc`、`palette_transpose_flag`などのシンタックス要素を含む。前記のようなパレットインデックス情報に基づいて、トラバーススキャン方向（垂直方向又は水平方向）に沿って巡回しながら現在ブロック内のサンプルに対してパレットインデックス（例：`PaletteIndexIdc`）を導出してパレットインデックスマップ（例：`PaletteIndexMap`）を構成することができる。

【0148】

また、パレットインデックスマップに基づいてパレットテーブル内のパレットエントリに対するサンプル値を導出し、前記パレットエントリにマッピングされたサンプル値に基づいて現在ブロックの復元サンプルを生成することができる。

【0149】

また、現在ブロック内にエスケープ値を有するサンプルが存在する場合（すなわち、`palette_escape_val_present_flag`の値が1である場合）、エスケープ情報に基づいて現在ブロックに対するエスケープ値を導出する。エスケープ情報は`palette_escape_val_present_flag`、`palette_escape_val`などのシンタックス要素を含む。例えば、量子化されたエスケープ値情報（例：`palette_escape_val`）に基づいて現在ブロック内のエスケープコーディングされたサンプルに対するエスケープ値を導出する。前記エスケ

10

20

30

40

50

ープ値に基づいて現在ブロックの復元サンプルを生成することができる。

【0150】

一方、エンコーディング/デコーディング過程において、BDPCM (block differential pulse coded modulation 又は Block-based Delta Pulse Code Modulation) 技法が使用される。BDPCMはRDPCM (quantized Residual block-based Delta Pulse Code Modulation) と名付けられてもよい。

【0151】

BDPCMを適用してブロックを予測する場合、ブロックの行又は列をラインバイラインで予測するために復元されたサンプルが活用できる。このとき、使用された参照サンプルはフィルタリングされていないサンプルであり得る。BDPCM方向は、垂直方向又は水平方向予測が使用されているかを示す。すなわち、BDPCMが適用される場合、垂直方向又は水平方向がBDPCM方向に選択され、前記BDPCM方向に予測が行われることができる。予測エラー (prediction error) は空間的ドメインにおいて量子化され、サンプルは予測 (すなわち、予測サンプル) に逆量子化された予測エラーを加えることにより復元される。前記予測エラーはレジデュアル (residual) を意味する。このようなBDPCMの代案として量子化されたレジデュアルドメインBDPCMが提案されることができ、予測方向やシグナリングは空間的ドメインに適用されたBDPCMと同一であり得る。すなわち、量子化されたレジデュアルドメインBDPCMを介して量子化係数自体をDPCM (Delta Pulse Code Modulation) のように積み上げた後、逆量子化によりレジデュアルが復元されることができ、従って、量子化されたレジデュアルドメインBDPCMは、レジデュアルコーディング段においてDPCMを適用するという意味で使用されることができ、以下で使われる量子化されたレジデュアルドメインは、予測に基づいて導出されたレジデュアルが変換なしに量子化されたものであり、量子化されたレジデュアルサンプルに対するドメインを意味する。例えば、量子化されたレジデュアルドメインは、変換スキップが適用される、すなわち、レジデュアルサンプルに対して変換はスキップされるが量子化は適用される量子化されたレジデュアル (又は、量子化されたレジデュアル係数) を含む。または、例えば、量子化されたレジデュアルドメインは量子化された変換係数を含む。

【0152】

前記のように、BDPCMは量子化されたレジデュアルドメインに適用されることができ、量子化されたレジデュアルドメインは量子化されたレジデュアル (又は、量子化されたレジデュアル係数) を含み、このとき、レジデュアルに対しては変換スキップが適用される。すなわち、BDPCMが適用される場合は、レジデュアルサンプルに対して変換はスキップされ、量子化は適用される。または、量子化されたレジデュアルドメインは量子化された変換係数を含むこともできる。BDPCMが適用可能であるか否かに関するフラグはシーケンスレベル (SPS) でシグナリングされ、このようなフラグはSPSで変換スキップモードが可能であるとシグナリングされる場合にのみシグナリングされることもできる。前記フラグは、BDPCM可用フラグ又はSPS BDPCM可用フラグと呼ばれてもよい。

【0153】

BDPCM適用の時、イントラ予測は、イントラ予測方向と類似の予測方向 (例えば、垂直予測又は水平予測) によるサンプル複製 (sample copy) により全体ブロックに行われる。原本と予測ブロックの差分値であるレジデュアルは変換がスキップされて量子化され、量子化されたレジデュアルと水平又は垂直方向に対する予測子 (すなわち、水平又は垂直方向の量子化されたレジデュアル) 間のデルタ値、すなわち、差分値がコーディングされる。

【0154】

BDPCMが適用可能であると、CUサイズがルマサンプルに対するMaxTSSiz

e (最大変換スキップブロックサイズ)より小さいか等しく、CUがイントラ予測でコーディングされる場合、フラグ情報がCUレベルで送信されることができる。前記フラグ情報はBDPCMフラグと呼ばれてもよい。ここで、MaxTsSizeは、変換スキップモードが許容されるための最大ブロックサイズを意味し得る。前記フラグ情報は、通常のイントラコーディングが適用されるか又はBDPCMが適用されるかを指示することができる。BDPCMが適用されると、予測方向が水平方向であるか垂直方向であるかを指示するBDPCM予測方向フラグが送信されることができる。前記BDPCM予測方向フラグはBDPCM方向フラグとも呼ばれる。以後、ブロックはフィルタリングされていない参照サンプルを利用した通常の水平又は垂直イントラ予測過程を通じて予測されることができる。また、レジデュアルは量子化され、各量子化されたレジデュアルとその予測子、例えば、BDPCM予測方向によって水平又は垂直方向にある周辺位置の既に量子化されたレジデュアル間の差異値がコーディングされることができる。

10

【0155】

一方、前述のように、本文書において開示されるシンタックステーブル内の情報(シンタックス要素)は、映像/ビデオ情報に含まれ、エンコーディング装置において構成/エンコーディングされてビットストリーム形態でデコーディング装置に伝達される。デコーディング装置は、当該シンタックステーブル内の情報(シンタックス要素)をパーシング/デコーディングする。デコーディング装置は、デコーディングされた情報に基づいて現在ブロックに対するデコーディング手順(予測、(変換スキップベース)レジデュアル処理、BDPCM、パレットコーディングなど)を行うことができる。

20

【0156】

以下、本文書においては変換スキップ関連ハイレベルシンタックス要素(high level syntax element)及び/又はパレットコーディング関連ハイレベルシンタックス要素について、従属性のあるシンタックス要素(syntax element)をパーシング/シグナリングする効率的な方案を提案する。すなわち、本文書の実施例によると、ビデオ/映像コーディング時に変換スキップ及び/又はパレットコーディングの実行において必須であるか補助的に使用される情報の従属性及び非従属性によってコーディングの可否を区分することができ、これにより、効率的なコーディングを行うことができる。

【0157】

ビデオ符号化においては、符号化コーディングツールのスイッチが特定ハイレベルシンタックス(high level syntax、HLS)内において定義できる。従来のVVCの場合、各コーディングツールに関するフラグ情報がSPSにおいて定義されることができる。また、VVCにおいては、各ハイレベルシンタックスセット(例えば、VPS(video parameter set)、SPS(sequence parameter set)、PPS(picture parameter set)、APS(adaptation parameter set)、DPS(decoding parameter set)、スライスヘッダ(Slice header)など)間には独立性を有することを志向して標準化が進んできた。従って、コーディングツールのフラグが存在するハイレベルシンタックスセットの内部においては従属性を有するシンタックス要素が多数存在するようになる。本文書の実施例(ら)においては変換スキップ及び/又はパレットコーディングによって従属性を有するハイレベルシンタックス要素をパーシング/シグナリングする方法を提案する。

30

40

【0158】

一実施例として、本文書は、変換スキップ関連ハイレベルシンタックス要素(high level syntax element)に対して従属性のあるシンタックス要素(syntax element)が従属条件に応じてパーシング/シグナリングするか否かを決定することにより、送信されるビットを節約する方法を提案する。一例として、変換スキップ(可用)フラグによる変換スキップの使用可否に応じて従属性を有するハイレベルシンタックス要素をパーシングする方法を提案する。

50

【 0 1 5 9 】

例えば、変換スキップベースコーディングに従属されるシンタックス要素としては、変換スキップ（可用）フラグ（例えば、`sps_transform_skip_enabled_flag`）、変換スキップのための最小量子化パラメータ情報（例えば、`min_qp_prime_ts_minus4`）、BDPCMの適用可否に関する情報（例えば、`sps_bdpcm_enabled_flag`）などがある。一例として、変換スキップ（可用）フラグの値が1に定義された場合、関連フラグ又は情報シンタックス要素は必ず送信されなければならないが、変換スキップ（可用）フラグの値が0に定義された場合は、変換スキップ（可用）フラグシンタックス要素を除いた前記シンタックス要素を送信しないことがある。

10

【 0 1 6 0 】

すなわち、高級構文HLS（例えば、VPS、SPS、PPS、APS、DPS、Slice headerなど）内の変換スキップ（可用）フラグの値に応じて変換スキップする時、変換スキップブロックに対する最小量子化パラメータ情報、BDPCMの適用可否などの変換スキップ可否に従属されたハイレベルシンタックス要素を送信する方法を提案する。また、提案する方法は、本実施例で言及されたシンタックス要素のみに限定されず、変換スキップを行うか否かに応じて従属性を有しながら変換スキップ（可用）フラグが含まれたハイレベルシンタックスセットにおいて定義されている全てのハイレベルシンタックス要素を含むことができる。

20

【 0 1 6 1 】

前述のように変換スキップベースコーディングに関連するシンタックス要素がハイレベルシンタックスセットにおいて定義されることができ、次の表6の実施例においてのようにSPS（`sequence parameter set`）の内部で定義されることができる。

【 0 1 6 2 】

【表6】

<code>seq_parameter_set_rbsp()</code> {	Descriptor
(...)	
<code>sps_transform_skip_enabled_flag</code>	u(1)
<code>if(sps_transform_skip_enabled_flag) {</code>	
<code>sps_bdpcm_enabled_flag</code>	u(1)
<code>min_qp_prime_ts_minus4</code>	ue(v)
<code>}</code>	
(...)	
<code>rbsp_trailing_bits()</code>	
<code>}</code>	

30

【 0 1 6 3 】

また、例えば、前記SPSシンタックスのシンタックス要素のうち前述の実施例に関するシンタックス要素のセマンティック（`semantic`）は次の表7のように示すことができる。

40

【 0 1 6 4 】

【表 7】

<p>sps_transform_skip_enabled_flag equal to 1 specifies that transform_skip_flag may be present in the transform unit syntax. sps_transform_skip_enabled_flag equal to 0 specifies that transform_skip_flag is not present in the transform unit syntax.</p>	
<p>sps_bdpcm_enabled_flag equal to 1 specifies that intra_bdpcm_flag may be present in the coding unit syntax for intra coding units. sps_bdpcm_enabled_flag equal to 0 specifies that intra_bdpcm_flag is not present in the coding unit syntax for intra coding units. When not present, the value of sps_bdpcm_enabled_flag is inferred to be equal to 0.</p>	10
<p>min_qp_prime_ts_minus4 specifies the minimum allowed quantization parameter for transform skip mode as follows:</p>	20
$QpPrimeTsMin = 4 + \text{min_qp_prime_ts_minus4}$	

【0165】

前記表 6 及び表 7 を参照すると、SPS において変換スキップに関連したシンタックス要素が定義されることができ、sps_transform_skip_enabled_flag、sps_bdpcm_enabled_flag、min_qp_prime_ts_minus4 シンタックス要素を含む。

30

【0166】

sps_transform_skip_enabled_flag シンタックス要素は、その値が 0 であるか 1 であるかに基づいて変換スキップが可用であるか否かを示す。例えば、sps_transform_skip_enabled_flag の値が 1 である場合は変換スキップが可用であることを示し、この場合、transform_skip_flag が変換ユニットシンタックス (transform unit syntax) を介してパーシング/シグナリングされることができる。ここで、transform_skip_flag シンタックス要素は、当該変換ブロック (associated transform block) に変換が適用できるか否かを示す。sps_transform_skip_enabled_flag の値が 0 である場合は変換スキップが可用でないことを示し、この場合、transform_skip_flag が変換ユニットシンタックスにおいてパーシング/シグナリングされない。言い換えれば、変換スキップ可用フラグ sps_transform_skip_enabled_flag に基づいて変換ユニットシンタックスにおいて transform_skip_flag が存在しているか否かを示すことができる。

40

【0167】

sps_bdpcm_enabled_flag シンタックス要素は、その値が 0 であるか 1 であるかに基づいて BDPCM が可用であるか否かを示すことができる。例えば、sps_bdpcm_enabled_flag の値が 1 である場合は BDPCM が可用

50

であることを示し、この場合、`intra_bdpcm_flag` (又は、`intra_bdpcm_luma_flag`及び`intra_bdpcm_chroma_flag`) がイントラコーディングユニットに対するコーディングユニットシンタックス (`coding_unit_syntax`) を介してパーシング/シグナリングされることができ
 る。ここで、`intra_bdpcm_flag`シンタックス要素はBDPCMが現在コー
 ディングブロックに適用されるか否かを示す。`sps_bdpcm_enabled_flag`
 の値が0である場合はBDPCMが可用でないことを示し、この場合、`intra`
`_bdpcm_flag` (又は、`intra_bdpcm_luma_flag`及び`intra`
`_bdpcm_chroma_flag`) がイントラコーディングユニットに対す
 るコーディングユニットシンタックスにおいてパーシング/シグナリングされ
 ない。言い換
 えれば、BDPCM可用フラグ`sps_bdpcm_enabled_flag`に基づい
 てコーディングユニットシンタックスにおいて`intra_bdpcm_flag` (又は
 、`intra_bdpcm_luma_flag`及び`intra_bdpcm_chro`
`ma_flag`) が存在しているか否かを示すことができる。

10

【0168】

`min_qp_prime_ts_minus4`シンタックス要素は、変換スキップモ
 ードに対して許容される最小量子化パラメータ (`minimum allowed qua`
`ntization parameter`) を示す。例えば、`min_qp_prime`
`_ts_minus4`シンタックス要素に基づいて変換スキップモードでの最小量子化パ
 ラメータ値 (例: `QpPrimeTsMin`) が導出される。前記変換スキップモード
 の最小量子化パラメータに基づいてスケーリング過程 (逆量子化過程) で使用される量子
 化パラメータが導出される。また、前記量子化パラメータに基づいて現在ブロックに対し
 てスケーリング過程 (逆量子化過程) を行ってスケーリングされた変換係数 (逆量子化さ
 れた変換係数) を導出し、これに基づいて現在ブロックのレジデュアルサンプルを導出す
 ることができる。

20

【0169】

また、SPSにおいては変換スキップに関連するシンタックス要素のうち変換スキップ
 可用フラグシンタックス要素 (例: `sps_transform_skip_enabled_flag`) に対して従属性のあるシンタックス要素を定義することができる。例え
 ば、前記表6及び表7に開示されているように、SPSにおいて変換スキップ可用フラグ
 (例: `sps_transform_skip_enabled_flag`) の値に基づ
 いて変換スキップモードで変換スキップブロックに対する最小量子化パラメータ情報を示
 す`min_qp_prime_ts_minus4`シンタックス要素、BDPCMが可用
 であるか否かを示す`sps_bdpcm_enabled_flag`シンタックス要素が
 従属性を有することができる。一例として、変換スキップ可用フラグ (例: `sps_tr`
`ansform_skip_enabled_flag`) の値が1である場合、`min_`
`qp_prime_ts_minus4`シンタックス要素及び`sps_bdpcm_en`
`abled_flag`シンタックス要素がパーシング/シグナリングされることが
 できる。または、変換スキップ可用フラグ (例: `sps_transform_skip_en`
`abled_flag`) の値が0である場合、`min_qp_prime_ts_min`
`us4`シンタックス要素及び`sps_bdpcm_enabled_flag`シンタッ
 クス要素がパーシング/シグナリングされない。

30

40

【0170】

また、一実施例として、本文書は、変換スキップ関連ハイレベルシンタックス要素 (`high level syntax element`) に対して従属性のあるシンタックス
 要素 (`syntax element`) が従属条件に応じてパーシング/シグナリングす
 るか否かを決定することにより、送信されるビットを節約する方法を提案する。一例とし
 て、変換スキップ (可用) フラグによる変換スキップの使用可否によって従属性を有する
 ハイレベルシンタックス要素をパーシングする方法を提案する。

【0171】

50

例えば、変換スキップベースコーディングに従属されるシンタックス要素としては、変換スキップ（可用）フラグ（例えば、`sps_transform_skip_enabled_flag`）、変換スキップ適用サイズに関する情報（例えば、`log2_transform_skip_max_size_minus2`）、変換スキップのための最小量子化パラメータ情報（例えば、`min_qp_prime_ts_minus4`）、BDPCM適用可否情報（例えば、`sps_bdpcm_enabled_flag`）などがある。一例として、変換スキップ（可用）フラグの値が1に定義された場合、関連フラグ又は情報シンタックス要素は必ず送信されなければならないが、変換スキップ（可用）フラグの値が0に定義された場合は、変換スキップ（可用）フラグシンタックス要素を除いた前記シンタックス要素を送信しない。

10

【0172】

すなわち、高級構文HLS（例えば、VPS、SPS、PPS、APS、DPS、Slice headerなど）内の変換スキップ（可用）フラグの値に応じて変換スキップ適用最大サイズに関する情報、変換スキップ時の最小量子化パラメータ情報、BDPCM適用可否などの変換スキップ可否に従属されたハイレベルシンタックス要素を送信する方法を提案する。また、提案する方法は、本実施例において言及されたシンタックス要素のみに限定されず、変換スキップを行うか否かに応じて従属性を有しながら変換スキップ（可用）フラグが含まれたハイレベルシンタックスセットにおいて定義されている全てのハイレベルシンタックス要素を含む。

20

【0173】

前述のように、変換スキップベースコーディングに関連するシンタックス要素がハイレベルシンタックスセットにおいて定義されることができ、次の表8の実施例においてのようにSPS（sequence parameter set）内部において定義されることができる。ただし、従来は、PPS（picture parameter set）において定義されている変換スキップのための最大ブロックサイズ情報がHLS間の従属性を止揚するためにSPSに新たに定義されることができ、これは次の表8のように示すことができる。

【0174】

【表8】

	Descriptor
<code>seq_parameter_set_rbsp() {</code>	
(...)	
<code>sps_transform_skip_enabled_flag</code>	u(1)
if(<code>sps_transform_skip_enabled_flag</code>) {	
<code>sps_bdpcm_enabled_flag</code>	u(1)
<code>min_qp_prime_ts_minus4</code>	ue(v)
<code>log2_transform_skip_max_size_minus2</code>	ue(v)
}	
(...)	
<code>rsbsp_trailing_bits()</code>	
}	

30

40

【0175】

また、例えば、前記SPSシンタックスのシンタックス要素のうち前述の実施例に関するシンタックス要素のセマンティック（semantic）は次の表9のように示すことができる。

【0176】

50

【表 9】

<p>sps_transform_skip_enabled_flag equal to 1 specifies that transform_skip_flag may be present in the transform unit syntax. sps_transform_skip_enabled_flag equal to 0 specifies that transform_skip_flag is not present in the transform unit syntax.</p>	
<p>sps_bdpcm_enabled_flag equal to 1 specifies that intra_bdpcm_flag may be present in the coding unit syntax for intra coding units. sps_bdpcm_enabled_flag equal to 0 specifies that intra_bdpcm_flag is not present in the coding unit syntax for intra coding units. When not present, the value of sps_bdpcm_enabled_flag is inferred to be equal to 0.</p>	10
<p>min_qp_prime_ts_minus4 specifies the minimum allowed quantization parameter for transform skip mode as follows:</p> $QpPrimeTsMin = 4 + \text{min_qp_prime_ts_minus4}$	20
<p>log2_transform_skip_max_size_minus2 specifies the maximum block size used for transform skip, and shall be in the range of 0 to 3.</p> <p>When not present, the value of log2_transform_skip_max_size_minus2 is inferred to be equal to 0.</p>	30
<p>The variable MaxTsSize is set equal to $1 \ll (\text{log2_transform_skip_max_size_minus2} + 2)$.</p>	

【0177】

前記表 8 及び表 9 を参照すると、SPS において変換スキップに関連したシンタックス要素が定義されることができ、sps_transform_skip_enabled_flag、sps_bdpcm_enabled_flag、min_qp_prime_ts_minus4、log2_transform_skip_max_size_minus2 シンタックス要素を含む。

40

【0178】

ここで、sps_transform_skip_enabled_flag、sps_bdpcm_enabled_flag、min_qp_prime_ts_minus4 シンタックス要素は、前記表 6 及び表 7 において詳細に説明しているため、本実施例においては説明の便宜のために具体的な説明を省略する。

【0179】

50

`log2_transform_skip_max_size_minus2` シンタックス要素は変換スキップモードで使用される最大ブロックサイズを示す。この時、`log2_transform_skip_max_size_minus2` シンタックス要素は0ないし3の範囲内にあり得る。例えば、前記表9に開示されているように、変換スキップモードにおいて使用される最大ブロックサイズ(例: `MaxTsSize`)は $1 < < (\text{log2_transform_skip_max_size_minus2} + 2)$ のような計算に基づいて導出される。

【0180】

また、SPSにおいては変換スキップに関連するシンタックス要素のうち変換スキップ
 10 可用フラグシンタックス要素(例: `sps_transform_skip_enabled_flag`)に対して従属性のあるシンタックス要素を定義することができる。例えば、前記表8及び表9に開示されているように、SPSにおいて変換スキップ可用フラグ
 (例: `sps_transform_skip_enabled_flag`)の値に基づ
 いてBDPCMが可用であるか否かを示す `sps_bdpcm_enabled_flag` シンタックス要素、変換スキップモードにおいて変換スキップブロックに対する最小量
 量子化パラメータ情報を示す `min_qp_prime_ts_minus4` シンタックス
 要素、変換スキップモードにおいて使用される最大ブロックサイズを示す `log2_transform_skip_max_size_minus2` シンタックス要素が従属性
 を有することができる。一例として、変換スキップ可用フラグ(例: `sps_transform_skip_enabled_flag`)の値が1である場合、`sps_bdpcm_enabled_flag`、`min_qp_prime_ts_minus4`、`log2_transform_skip_max_size_minus2` シンタックス
 20 要素がパーシング/シグナリングされることができる。または、変換スキップ可用フラグ
 (例: `sps_transform_skip_enabled_flag`)の値が0で
 ある場合、`sps_bdpcm_enabled_flag`、`min_qp_prime_ts_minus4`、`log2_transform_skip_max_size_minus2` シンタックス要素がパーシング/シグナリングされない。

【0181】

また、一実施例として、本文書は、変換スキップ関連ハイレベルシンタックス要素及び
 パレットコーディング関連ハイレベルシンタックス要素に対して従属性のあるシンタックス
 30 要素が従属条件に応じてパーシング/シグナリングするか否かを決定することにより送
 信されるビットを節約する方法を提案する。一例として、変換スキップ(可用)フラッグ
 及び/又はパレットコーディング(可用)フラッグにより従属性を有するハイレベルシンタ
 ックス要素をパーシングする方法を提案する。

【0182】

例えば、変換スキップベースコーディングに従属されるシンタックス要素としては、変
 換スキップ(可用)フラグ(例えば、`sps_transform_skip_enabled_flag`)、変換スキップ適用サイズに関する情報(例えば、`log2_transform_skip_max_size_minus2`)、変換スキップ時の最小量
 40 量子化パラメータ情報(例えば、`min_qp_prime_ts_minus4`)、BD
 PCM適用可否情報(例えば、`sps_bdpcm_enabled_flag`)などが
 ある。また、前述したように、パレットコーディングの時にエスケープ値も変換されない
 ので、量子化の実行において変換スキップのための最小量子化パラメータ情報が使用され
 ることができる。従って、パレットモードベースコーディングに対してパレットコーディ
 ング(可用)フラグ(例えば、`sps_palette_enabled_flag`)、
 変換スキップ時の最小量子化パラメータ情報(例えば、`min_qp_prime_ts_minus4`)が従属性を有することができる。一例として、変換スキップ(可用)フ
 ラグ又はパレットコーディング(可用)フラグの値が1に定義された場合、関連フラッグ
 又は情報シンタックス要素は必ず送信されなければならないが、変換スキップ(可用)フ
 ラグ又はパレットコーディング(可用)フラグの値が0に定義された場合は各フラグシン
 50

タックス要素を除いた前記シンタックス要素を送信しない。

【0183】

すなわち、高級構文（例えば、VPS、SPS、PPS、APS、DPS、Slice header など）内の変換スキップ（可用）フラグ及び/又はパレットコーディング（可用）フラグの値に応じて変換スキップ適用最大サイズに関する情報、変換スキップ時の最小量子化パラメータ情報、BDPCM適用可否などの変換スキップ可否又はパレットコーディング可否に従属されたハイレベルシンタックス要素を送信する方法を提案する。

【0184】

例えば、(i) 変換スキップ（可用）フラグとパレットコーディング（可用）フラグが全て1に定義された時、変換スキップ（可用）フラグとパレットコーディング（可用）フラグに従属されたシンタックス要素の和集合に該当するシンタックス要素がパーシングされることができる。(ii) 変換スキップ（可用）フラグが1に定義され、パレットコーディング（可用）フラグが0である場合、変換スキップ（可用）フラグに従属されたシンタックス要素がパーシングされることができる。(iii) 変換スキップ（可用）フラグが0に定義され、パレットコーディング（可用）フラグが1である時は、パレットコーディング（可用）フラグに従属されたシンタックス要素がパーシングされることができる。(iv) 変換スキップ（可用）フラグ及びパレットコーディング（可用）フラグが全て0である場合、2種類のコーディングツールに従属性を有する他のハイレベルシンタックス要素はパーシングされない。

10

【0185】

本実施例で言及したシンタックス要素のパーシング順序は特定に限定されず、シンタックス要素間の従属性に応じてパーシングするか否かが決定される場合は一致するものとみなす。また、提案する方法は、本実施例で言及されたシンタックス要素のみに限定されず、変換スキップ可否又はパレットコーディング可否に応じて従属性を有しながら変換スキップ（可用）フラグ、パレットコーディング（可用）フラグが含まれたハイレベルシンタックスセットにおいて定義されている全てのハイレベルシンタックス要素を含む。

20

【0186】

前述のように変換スキップベースコーディング及び/又はパレットモードベースコーディングに関連するシンタックス要素がハイレベルシンタックスセットにおいて定義されることができ、次の表10の実施例のようにSPS(sequence parameter set)内部において定義されることができる。

30

【0187】

【表10】

seq_parameter_set_rbsp() {	Descriptor
(...)	
sps_transform_skip_enabled_flag	u(1)
if(sps_transform_skip_enabled_flag)	
sps_bdpcm_enabled_flag	u(1)
(...)	
if(chroma_format_idc == 3)	
sps_palette_enabled_flag	u(1)
(...)	
if(sps_transform_skip_enabled_flag sps_palette_enabled_flag)	
min_qp_prime_ts_minus4	ue(v)
(...)	
}	

40

【0188】

また、例えば、前記SPSシンタックスのシンタックス要素のうち前述の実施例に関する

50

るシンタックス要素のセマンティック (s e m a n t i c) は次の表 1 1 のように示すことができる。

【 0 1 8 9 】

【表 1 1】

<p>sps_transform_skip_enabled_flag equal to 1 specifies that transform_skip_flag may be present in the transform unit syntax. sps_transform_skip_enabled_flag equal to 0 specifies that transform_skip_flag is not present in the transform unit syntax.</p> <p>sps_bdpcm_enabled_flag equal to 1 specifies that intra_bdpcm_flag may be present in the coding unit syntax for intra coding units. sps_bdpcm_enabled_flag equal to 0 specifies that intra_bdpcm_flag is not present in the coding unit syntax for intra coding units. When not present, the value of sps_bdpcm_enabled_flag is inferred to be equal to 0.</p> <p>min_qp_prime_ts_minus4 specifies the minimum allowed quantization parameter for transform skip mode as follows:</p> $QpPrimeTsMin = 4 + min_qp_prime_ts_minus4$ <p>sps_palette_enabled_flag equal to 1 specifies that pred_mode_plt_flag may be present in the coding unit syntax. sps_palette_enabled_flag equal to 0 specifies that pred_mode_plt_flag is not present in the coding unit syntax. When sps_palette_enabled_flag is not present, it is inferred to be equal to 0.</p>	<p>10</p> <p>20</p> <p>30</p>
--	-------------------------------

【 0 1 9 0 】

前記表 1 0 及び表 1 1 を参照すると、SPS において変換スキップ及び/又はパレットコーディングに関連したシンタックス要素が定義されることができ、sps_transform_skip_enabled_flag、sps_bdpcm_enabled_flag、sps_palette_enabled_flag、min_qp_prime_ts_minus4 シンタックス要素を含む。

【 0 1 9 1 】

ここで、sps_transform_skip_enabled_flag、sps_bdpcm_enabled_flag、min_qp_prime_ts_minus4 シンタックス要素は、前記表 6 ないし表 9 において詳細に説明しているので、本実施例においては説明の便宜のために具体的な説明を省略する。

【 0 1 9 2 】

sps_palette_enabled_flag シンタックス要素は、その値が 0

10

20

30

40

50

であるか1であるかに基づいてパレットコーディング(すなわち、パレット予測モード)が可用であるか否かを示す。例えば、`sps_palette_enabled_flag`の値が1である場合はパレットコーディングが可用であることを示し、この場合、`pred_mode_plt_flag`がコーディングユニットシンタックス(`coding_unit_syntax`)を介してパーシング/シグナリングされる。ここで、`pred_mode_plt_flag`シンタックス要素は現在コーディングユニットにパレットモードが使用できるか否かを示す。`sps_palette_enabled_flag`の値が0である場合はパレットコーディングが可能ではないことを示し、この場合、`pred_mode_plt_flag`がコーディングユニットシンタックスにおいてパーシング/シグナリングされない。言い換えれば、パレットコーディング可用フラグ`sps_palette_enabled_flag`に基づいてコーディングユニットシンタックスにおいて`pred_mode_plt_flag`が存在するか否かを示す。

10

【0193】

また、SPSにおいては変換スキップ及び/又はパレットコーディングに関連するシンタックス要素のうち変換スキップ可用フラグシンタックス要素(例:`sps_transform_skip_enabled_flag`)に対して従属性があるシンタックス要素を定義することができる。例えば、前記表10及び表11に開示されているように、SPSにおいて変換スキップ可用フラグ(例:`sps_transform_skip_enabled_flag`)の値に基づいてBDPCMが可用であるか否かを示す`sps_bdpcm_enabled_flag`シンタックス要素が従属性を有することができる。一例として、変換スキップ可用フラグ(例:`sps_transform_skip_enabled_flag`)の値が1である場合、`sps_bdpcm_enabled_flag`シンタックス要素がパーシング/シグナリングされる。または、変換スキップ可用フラグ(例:`sps_transform_skip_enabled_flag`)の値が0である場合、`sps_bdpcm_enabled_flag`シンタックス要素がパーシング/シグナリングされない。

20

【0194】

また、SPSにおいては変換スキップ及び/又はパレットコーディングに関連するシンタックス要素のうちパレットコーディング可用フラグシンタックス要素(例:`sps_palette_enabled_flag`)と関連して従属条件を定義することができる。例えば、前記表10及び表11に開示されているように、SPSにおいて後述する`chroma_format_idc`シンタックス要素に基づいてパレットコーディング可用フラグシンタックス要素(例:`sps_palette_enabled_flag`)がパーシング/シグナリングされることができる。一例として、`chroma_format_idc`シンタックス要素の値が3である場合、`sps_palette_enabled_flag`シンタックス要素がパーシング/シグナリングされる。

30

【0195】

また、SPSにおいては変換スキップ及び/又はパレットコーディングに関連するシンタックス要素のうち変換スキップ可用フラグシンタックス要素(例:`sps_transform_skip_enabled_flag`)及び/又はパレットコーディング可用フラグシンタックス要素(例:`sps_palette_enabled_flag`)に対して従属性があるシンタックス要素を定義することができる。例えば、前記表10及び表11に開示されているように、SPSにおいて変換スキップ可用フラグ(例:`sps_transform_skip_enabled_flag`)及び/又はパレットコーディング可用フラグシンタックス要素(例:`sps_palette_enabled_flag`)の値に基づいて変換スキップモードに対する最小量子化パラメータ情報を示す`min_qp_prime_ts_minus4`シンタックス要素が従属性を有することができる。一例として、変換スキップ可用フラグ(例:`sps_transform_skip_enabled_flag`)の値が1であるか又はパレットコーディング可用フラグ(例:`sps_palette_enabled_flag`)の値が1である場合、m

40

50

in_qp_prime_ts_minus4 シンタックス要素がパーシング/シグナリングされる。

【0196】

また、前述のように変換スキップベースコーディング及び/又はパレットモードベースコーディングに関連するシンタックス要素がハイレベルシンタックスセットにおいて定義されることができ、次の表12の実施例のようにSPS(sequence parameter set)内部で定義されることができ、ただし、従来はPPS(picture parameter set)で定義されている変換スキップのための最大ブロックサイズ情報がHLS間の従属性を止揚するためにSPSに新たに定義されることができ、これは次の表12のように示すことができる。

10

【0197】

【表12】

seq_parameter_set_rbsp() {	Descriptor
(...)	
sps_transform_skip_enabled_flag	u(1)
if(sps_transform_skip_enabled_flag) {	
sps_bdpcm_enabled_flag	u(1)
log2_transform_skip_max_size_minus2	ue(v)
}	
(...)	
if(chroma_format_idc == 3)	
sps_palette_enabled_flag	u(1)
(...)	
if(sps_transform_skip_enabled_flag sps_palette_enabled_flag)	
min_qp_prime_ts_minus4	ue(v)
(...)	
}	

20

【0198】

また、例えば、前記SPSシンタックスのシンタックス要素のうち前述の実施例に対するシンタックス要素のセマンティック(semantic)は、下表13のように示すことができる。

30

【0199】

40

50

【表 1 3】

<p>sps_transform_skip_enabled_flag equal to 1 specifies that <code>transform_skip_flag</code> may be present in the transform unit syntax. <code>sps_transform_skip_enabled_flag</code> equal to 0 specifies that <code>transform_skip_flag</code> is not present in the transform unit syntax.</p>	
<p>sps_bdpcm_enabled_flag equal to 1 specifies that <code>intra_bdpcm_flag</code> may be present in the coding unit syntax for intra coding units. <code>sps_bdpcm_enabled_flag</code> equal to 0 specifies that <code>intra_bdpcm_flag</code> is not present in the coding unit syntax for intra coding units. When not present, the value of <code>sps_bdpcm_enabled_flag</code> is inferred to be equal to 0.</p>	10
<p>min_qp_prime_ts_minus4 specifies the minimum allowed quantization parameter for transform skip mode as follows:</p> $QpPrimeTsMin = 4 + \text{min_qp_prime_ts_minus4}$	20
<p>log2_transform_skip_max_size_minus2 specifies the maximum block size used for transform skip, and shall be in the range of 0 to 3.</p> <p>When not present, the value of <code>log2_transform_skip_max_size_minus2</code> is inferred to be equal to 0.</p>	
<p>The variable <code>MaxTsSize</code> is set equal to $1 \ll (\text{log2_transform_skip_max_size_minus2} + 2)$.</p>	30
<p>sps_palette_enabled_flag equal to 1 specifies that <code>pred_mode_plt_flag</code> may be present in the coding unit syntax. <code>sps_palette_enabled_flag</code> equal to 0 specifies that <code>pred_mode_plt_flag</code> is not present in the coding unit syntax. When <code>sps_palette_enabled_flag</code> is not present, it is inferred to be equal to 0.</p>	40

【0 2 0 0】

前記の表 1 2 及び表 1 3 を参照すると、SPS において変換スキップ及びノ又はパレットコーディングに関連するシンタックス要素が定義されることができ、`sps_transform_skip_enabled_flag`、`sps_bdpcm_enabled_flag`、`log2_transform_skip_max_size_minus2`、`sps_palette_enabled_flag`、`min_qp_prime_ts_minus4` シンタックス要素を含む。

【0 2 0 1】

ここで、`sps_transform_skip_enabled_flag`、`sps` 50

`__bdpcm_enabled_flag`、`log2_transform_skip_max_size_minus2`、`sps_palette_enabled_flag`、`min_qp_prime_ts_minus4` シンタックス要素は前記表 6 ないし表 11 において詳細に説明しているの、本実施例においては説明の便宜のために具体的な説明は省略する。

【0202】

前記表 12 及び表 13 の実施例に開示されているように、SPS においては、変換スキップ及び/又はパレットコーディングに関連するシンタックス要素のうち変換スキップ可用フラグシンタックス要素(例：`sps_transform_skip_enabled_flag`)に対して従属性のあるシンタックス要素を定義することができる。例えば、前記表 12 及び表 13 に開示されているように、SPS において変換スキップ可用フラグ(例：`sps_transform_skip_enabled_flag`)の値に基づいてBDPCMが可用であるか否かを示す`sps_bdpcm_enabled_flag`シンタックス要素、変換スキップモードにおいて使用される最大ブロックサイズを示す`log2_transform_skip_max_size_minus2`シンタックス要素が従属性を有することができる。一例として、変換スキップ可用フラグ(例：`sps_transform_skip_enabled_flag`)の値が1である場合、`sps_bdpcm_enabled_flag`、`log2_transform_skip_max_size_minus2`シンタックス要素がパーシング/シグナリングされる。または、変換スキップ可用フラグ(例：`sps_transform_skip_enabled_flag`)の値が0である場合、`sps_bdpcm_enabled_flag`、`log2_transform_skip_max_size_minus2`シンタックス要素がパーシング/シグナリングされない。

10

20

【0203】

また、SPS においては変換スキップ及び/又はパレットコーディングに関連するシンタックス要素のうち変換スキップ可用フラグシンタックス要素(例：`sps_transform_skip_enabled_flag`)及び/又はパレットコーディング可用フラグシンタックス要素(例：`sps_palette_enabled_flag`)に対して従属性があるシンタックス要素を定義することができる。例えば、前記表 12 及び表 13 に開示されているように、SPS において変換スキップ可用フラグ(例：`sps_transform_skip_enabled_flag`)及び/又はパレットコーディング可用フラグシンタックス要素(例：`sps_palette_enabled_flag`)の値に基づいて変換スキップモードに対する最小量子化パラメータ情報を示す`min_qp_prime_ts_minus4`シンタックス要素が従属性を有することができる。一例として、変換スキップ可用フラグ(例：`sps_transform_skip_enabled_flag`)の値が1であるか又はパレットコーディング可用フラグ(例：`sps_palette_enabled_flag`)の値が1である場合、`min_qp_prime_ts_minus4`シンタックス要素がパーシング/シグナリングされる。

30

【0204】

一方、ソース又はコーディングされたピクチャ/映像はルマ成分アレイを含み、場合によって2つのクロマ成分(`cb`、`cr`)アレイをさらに含んでもよい。すなわち、ピクチャ/映像の1つのピクセルはルマサンプル及びクロマサンプル(`cb`、`cr`)を含む。

40

【0205】

カラーフォーマットはルマ成分とクロマ成分(`cb`、`cr`)の構成フォーマットを示し、クロマフォーマットと呼ばれてもよい。前記カラーフォーマット(又は、クロマフォーマット)は予め決まってもよく、または適応的にシグナリングされてもよい。例えば、前記クロマフォーマットは、以下の表 14 のように、`chroma_format_idc` 及び `separate_colour_plane_flag` の少なくとも1つに基づいてシグナリングされることができる。

50

【 0 2 0 6 】

【 表 1 4 】

chroma_format_idc	separate_colour_plane_flag	Chroma format	SubWidthC	SubHeightC
0	0	Monochrome	1	1
1	0	4:2:0	2	2
2	0	4:2:2	2	1
3	0	4:4:4	1	1
3	1	4:4:4	1	1

10

【 0 2 0 7 】

前記表 1 4 を参照すると、単色サンプリング (monochrome sampling) には名目上 (nominally) ルマアレイと見なされる 1 つのサンプルアレイのみがある。

【 0 2 0 8 】

4 : 2 : 0 サンプリングにおいて、2 つのクロマアレイそれぞれはルマアレイの高さの半分及び幅の半分を有する。

【 0 2 0 9 】

4 : 2 : 2 サンプリングにおいて、2 つのクロマアレイそれぞれはルマアレイの同一の高さ及び幅の半分を有する。

20

【 0 2 1 0 】

4 : 4 : 4 サンプリングにおいて、separate_colour_plane_flag の値に応じて以下が適用される。

【 0 2 1 1 】

- もし、separate_colour_plane_flag の値が 0 である場合、2 つのクロマアレイそれぞれはルマアレイと同一の高さ及び幅を有する。

【 0 2 1 2 】

- そうではなく、separate_colour_plane_flag の値が 1 である場合、3 つのカラープレーン (colour planes) が単色サンプリングされたピクチャ (monochrome sampled pictures) に別途処理される。

30

【 0 2 1 3 】

SubWidthC 及び SubHeightC はルマサンプルとクロマサンプル間の割合を示すことができる。例えば、chroma_format_idc が 3 である場合、クロマフォーマットは 4 : 4 : 4 であり、この場合、ルマサンプルブロックの幅が 16 であるとき、当該クロマサンプルブロックの幅は $16 / \text{SubWidthC}$ であり得る。一般的に、クロマサンプル関連シンタックス及びビットストリームはクロマアレイタイプ (例 : chromaArrayType) が 0 でない場合にのみパーズングされる。

【 0 2 1 4 】

また、一実施例として、本文書は変換スキップ関連ハイレベルシンタックス要素及びパレットコーディング関連ハイレベルシンタックス要素に対して従属性があるシンタックス要素が従属条件に応じてパーズング / シグナリングされるか否かを決定することにより送信されるビットを節約する方法を提案する。一例として、変換スキップ (可用) フラグ及び / 又はパレットコーディング (可用) フラグにより従属性を有するハイレベルシンタックス要素をパーズングする方法を提案する。

40

【 0 2 1 5 】

例えば、変換スキップベースコーディングに従属されるシンタックス要素としては、変換スキップ (可用) フラグ (例えば、sps_transform_skip_enabled_flag)、変換スキップ適用サイズに関する情報 (例えば、log2_transform_skip_max_size_minus2)、変換スキップ時の最小量

50

子化パラメータ情報（例えば、`min__qp__prime__ts__minus4`）、BDPCM適用可否情報（例えば、`sps__bdpcm__enabled__flag`）などがある。また、前述したように、パレットコーディング時にエスケープ値も変換されないの
 で、量子化の実行において変換スキップのための最小量子化パラメータ情報が使用される
 ことができる。

【0216】

前述の実施例で示したように、変換スキップ（可用）フラグ又はパレットコーディング
 （可用）フラグの値が1に定義された場合、関連フラグ又は情報シンタックス要素は必
 ず送信されなければならないが、変換スキップ（可用）フラグ又はパレットコーディング
 （可用）フラグの値が0に定義された場合は、各フラグシンタックス要素を除いた前記シ
 ンタックス要素を送信しないことがある。つまり、高級構文（例えば、VPS、SPS、
 PPS、APS、DPS、Slice headerなど）内の変換スキップ（可用）フ
 ラグ及びパレットコーディング（可用）フラグの値に応じて変換スキップ又はパレットコ
 ーディング時の最小量子化パラメータ情報、BDPCM適用可否などの変換スキップ可否
 又はパレットコーディング可否に従属されたハイレベルシンタックス要素を送信する方
 法を提案する。

10

【0217】

例えば、(i)変換スキップ（可用）フラグとパレットコーディング（可用）フラグが
 全て1に定義された時、変換スキップ（可用）フラグとパレットコーディング（可用）フ
 ラグに従属されたシンタックス要素の和集合に該当するシンタックス要素がパーシングさ
 れることができる。(ii)変換スキップ（可用）フラグが1に定義され、パレットコ
 ーディング（可用）フラグが0である場合、変換スキップ（可用）フラグに従属されたシ
 ンタックス要素がパーシングされることができる。(iii)変換スキップ（可用）フラグ
 が0に定義され、パレットコーディング（可用）フラグが1である時はパレットコ
 ーディング（可用）フラグに従属されたシンタックス要素がパーシングされることができ
 る。(iv)変換スキップ（可用）フラグ及びパレットコーディング（可用）フラグが全て0で
 ある場合、2つコーディングツールに従属性を有する他のハイレベルシンタックス要素は
 パーシングされないことがある。

20

【0218】

本実施例で言及したシンタックス要素のパーシング順序は特定に限定されず、シンタッ
 クス要素間の従属性に応じてパーシングするか否かが決定される場合、一致するものとみ
 なす。また、提案する方法は、本実施例で言及されたシンタックス要素のみに限定せず、
 変換スキップ可否又はパレットコーディング可否に応じて従属性を有しながら変換スキ
 ップ（可用）フラグ、パレットコーディング（可用）フラグが含まれたハイレベルシンタ
 ックスセットにおいて定義されている全てのハイレベルシンタックス要素を含むことが
 できる。

30

【0219】

前述のように変換スキップベースコーディング及び/又はパレットモードベースコーデ
 ィングに関連するシンタックス要素がハイレベルシンタックスセットにおいて定義される
 ことができ、次の表15の実施例のようにSPS(sequence parameter
 set)内部において定義されることができる。

40

【0220】

【表 1 5】

	Descriptor
seq_parameter_set_rbsp() {	
(...)	
sps_transform_skip_enabled_flag	u(1)
if(sps_transform_skip_enabled_flag)	
sps_bdpcm_enabled_flag	u(1)
(...)	
if(chroma_format_idc == 3)	
sps_palette_enabled_flag	u(1)
(...)	
if(sps_palette_enabled_flag)	
min_qp_prime_ts_chroma_minus4	ue(v)
(...)	
if(sps_transform_skip_enabled_flag sps_palette_enabled_flag)	
min_qp_prime_ts_luma_minus4	ue(v)
(...)	
}	

10

【 0 2 2 1】

20

また、例えば、前記 S P S シンタックスのシンタックス要素のうち前述の実施例に関するシンタックス要素のセマンティック (s e m a n t i c) は次の表 1 6 のように示す。

【 0 2 2 2】

30

40

50

【表 1 6】

<p>sps_transform_skip_enabled_flag equal to 1 specifies that transform_skip_flag may be present in the transform unit syntax. sps_transform_skip_enabled_flag equal to 0 specifies that transform_skip_flag is not present in the transform unit syntax.</p> <p>sps_bdpcm_enabled_flag equal to 1 specifies that intra_bdpcm_flag may be present in the coding unit syntax for intra coding units. sps_bdpcm_enabled_flag equal to 0 specifies that intra_bdpcm_flag is not present in the coding unit syntax for intra coding units. When not present, the value of sps_bdpcm_enabled_flag is inferred to be equal to 0.</p> <p>min_qp_prime_ts_luma_minus4 specifies the minimum allowed quantization parameter for transform skip mode in the luma component as follows:</p> $QpPrimeTsMin = 4 + \text{min_qp_prime_ts_luma_minus4}$ <p>min_qp_prime_ts_chroma_minus4 specifies the minimum allowed quantization parameter for transform skip mode in the chroma component as follows:</p> $QpPrimeTsMin = 4 + \text{min_qp_prime_ts_chroma_minus4}$ <p>sps_palette_enabled_flag equal to 1 specifies that pred_mode_plt_flag may be present in the coding unit syntax. sps_palette_enabled_flag equal to 0 specifies that pred_mode_plt_flag is not present in the coding unit syntax. When sps_palette_enabled_flag is not present, it is inferred to be equal to 0.</p>	<p>10</p> <p>20</p> <p>30</p>
--	-------------------------------

【0 2 2 3】

前記表 1 5 及び表 1 6 を参照すると、SPS において変換スキップ及び/又はパレットコーディングに関連したシンタックス要素が定義されることができ、sps_transform_skip_enabled_flag、sps_bdpcm_enabled_flag、sps_palette_enabled_flag、min_qp_prime_ts_luma_minus4、min_qp_prime_ts_chroma_minus4 シンタックス要素を含む。

【0 2 2 4】

ここで、sps_transform_skip_enabled_flag、sps_bdpcm_enabled_flag、sps_palette_enabled_flag シンタックス要素は、前記表 6 ないし表 1 1 で詳細に説明しているので、本実施例では説明の便宜のための具体的な説明を省略する。

【0 2 2 5】

10

20

30

40

50

前記表 15 及び表 16 の実施例に開示されているように、SPS においては変換スキップ及び/又はパレットコーディングに関連するシンタックス要素のうち変換スキップ可用フラグシンタックス要素(例: `sps__transform__skip__enabled__flag`)に対して従属性があるシンタックス要素を定義することができる。例えば、前記表 15 及び表 16 に開示されているように、SPS において変換スキップ可用フラグ(例: `sps__transform__skip__enabled__flag`)の値に基づいてBDPCMが可用であるか否かを示す `sps__bdpcm__enabled__flag` シンタックス要素が従属性を有することができる。一例として、変換スキップ可用フラグ(例: `sps__transform__skip__enabled__flag`)の値が1である場合、`sps__bdpcm__enabled__flag` シンタックス要素がパーシング/シグナリングされる。または、変換スキップ可用フラグ(例: `sps__transform__skip__enabled__flag`)の値が0である場合、`sps__bdpcm__enabled__flag` シンタックス要素がパーシング/シグナリングされない。
【0226】

10

また、SPS においては変換スキップ及び/又はパレットコーディングに関連するシンタックス要素のうちパレットコーディング可用フラグシンタックス要素(例: `sps__palette__enabled__flag`)に対して従属性があるシンタックス要素を定義することができる。例えば、前記表 15 及び表 16 に開示されているように、SPS においてパレットコーディング可用フラグ(例: `sps__palette__enabled__flag`)の値に基づいてクロマ成分に対する変換スキップモードにおいての最小量子化パラメータ情報を示す `min__qp__prime__ts__chroma__minus4` シンタックス要素が従属性を有することができる。一例として、パレットコーディング可用フラグ(例: `sps__palette__enabled__flag`)の値が1である場合、`min__qp__prime__ts__chroma__minus4` シンタックス要素がパーシング/シグナリングされる。または、パレットコーディング可用フラグ(例: `sps__palette__enabled__flag`)の値が0である場合、`min__qp__prime__ts__chroma__minus4` シンタックス要素がパーシング/シグナリングされない。
【0227】

20

また、SPS においては変換スキップ及び/又はパレットコーディングに関連するシンタックス要素のうち変換スキップ可用フラグシンタックス要素(例: `sps__transform__skip__enabled__flag`)及び/又はパレットコーディング可用フラグシンタックス要素(例: `sps__palette__enabled__flag`)に対して従属性があるシンタックス要素を定義することができる。例えば、前記表 15 及び表 16 に開示されているように、SPS において変換スキップ可用フラグ(例: `sps__transform__skip__enabled__flag`)及び/又はパレットコーディング可用フラグシンタックス要素(例: `sps__palette__enabled__flag`)の値に基づいてルマ成分に対する変換スキップモードにおいての最小量子化パラメータ情報を示す `min__qp__prime__ts__luma__minus4` シンタックス要素が従属性を有することができる。一例として、変換スキップ可用フラグ(例: `sps__transform__skip__enabled__flag`)の値が1であるか又はパレットコーディング可用フラグ(例: `sps__palette__enabled__flag`)の値が1である場合、`min__qp__prime__ts__luma__minus4` シンタックス要素がパーシング/シグナリングされる。
【0228】

30

40

また、一実施例として、本文書は、変換スキップ関連ハイレベルシンタックス要素及びパレットコーディング関連ハイレベルシンタックス要素に対して従属性があるシンタックス要素が従属条件に応じてパーシング/シグナリングするか否かを決定することにより送信されるビットを節約する方法を提案する。一例として、変換スキップ(可用)フラグ及び/又はパレットコーディング(可用)フラグにより従属性を有するハイレベルシンタ

50

ックス要素をパーシングする方法を提案する。

【0229】

例えば、変換スキップベースコーディングに従属されるシンタックス要素としては、変換スキップ（可用）フラグ（例えば、`sps__transform__skip__enabled__flag`）、変換スキップ適用サイズに関する情報（例えば、`log2__transform__skip__max__size__minus2`）、変換スキップ時の最小量子化パラメータ情報（例えば、`min__qp__prime__ts__minus4`）、BDPCM適用可否情報（例えば、`sps__bdpcm__enabled__flag`）などがある。また、前述したように、パレットコーディング時にエスケープ値も変換されないの
10
ので、量子化の実行において変換スキップのための最小量子化パラメータ情報が使用される
ことができる。

【0230】

前述の実施例に示したように、変換スキップ（可用）フラグ又はパレットコーディング（可用）フラグの値が1に定義された場合、関連フラグあるいは情報シンタックス要素は必ず送信されなければならないが、変換スキップ（可用）フラグ又はパレットコーディング（可用）フラグの値が0に定義された場合は、各フラグシンタックス要素を除いた前記シンタックス要素を送信しない。すなわち、高級構文（例えば、VPS、SPS、PPS、APS、DPS、Slice headerなど）内の変換スキップ（可用）フラグ及びパレットコーディング（可用）フラグの値に応じて変換スキップ適用最大サイズに関する情報、変換スキップ又はパレットコーディング時の少なくとも量子化パラメータ情報
20
、BDPCM適用可否などの変換スキップ可否又はパレットコーディング可否に従属された
ハイレベルシンタックス要素を送信する方法を提案する。

【0231】

例えば、(i)変換スキップ（可用）フラグとパレットコーディング（可用）フラグが
全て1に定義された時、変換スキップ（可用）フラグとパレットコーディング（可用）
フラグに従属されたシンタックス要素の和集合に該当するシンタックス要素がパーシング
される
ことができる。(ii)変換スキップ（可用）フラグが1に定義され、パレットコー
ディング（可用）フラグが0である場合、変換スキップ（可用）フラグに従属されたシン
タックス要素がパーシングされる。(iii)変換スキップ（可用）フラグが0に定義され、
パレットコーディング（可用）フラグが1である時は、パレットコーディング（可用）
30
フラグに従属されたシンタックス要素がパーシングされる。(iv)変換スキップ（可
用）フラグ及びパレットコーディング（可用）フラグが全て0である場合、2つのコー
ディングツールに従属性を有する他のハイレベルシンタックス要素はパーシングされない。

【0232】

本実施例で言及したシンタックス要素のパーシング順序は特定に限定せず、シンタック
ス要素間の従属性に応じてパーティションするか否かが決定される場合、一致するものと
みなす。また、提案する方法は、本実施例で言及されたシンタックス要素のみに限定せず
、変換スキップ可否又はパレットコーディング可否によって従属性を有しながら変換スキ
ップ（可用）フラグ、パレットコーディング（可用）フラグが含まれたハイレベルシンタ
ックスセットにおいて定義されている全てのハイレベルシンタックス要素を含む。
40

【0233】

前述のように、変換スキップベースコーディング及び/又はパレットモードベースコー
ディングに関連するシンタックス要素がハイレベルシンタックスセットにおいて定義され
ることができ、次の表17の実施例のようにSPS（`sequence parameter set`）内部において定義されることができ
る。本実施例では、従来PPS（`picture parameter set`）において定義されている変換スキップ最大サイ
ズに関する情報がHLS間の従属性を止揚するためにSPSに新たに定義されることができ
、既存に使用される変換スキップ及びパレットコーディング関連シンタックス要素の従
属性に基づいてパーシング/シグナリングされる方法を提案する。

【0234】

10

20

30

40

50

【表 1 7】

	Descriptor
seq_parameter_set_rbsp() {	
(...)	
sps_transform_skip_enabled_flag	u(1)
if(sps_transform_skip_enabled_flag) {	
sps_bdpcm_enabled_flag	u(1)
log2_transform_skip_max_size_minus2	ue(v)
}	
(...)	
if(chroma_format_idc == 3)	
sps_palette_enabled_flag	u(1)
(...)	
if(sps_palette_enabled_flag)	
min_qp_prime_ts_chroma_minus4	ue(v)
(...)	
if(sps_transform_skip_enabled_flag sps_palette_enabled_flag)	
min_qp_prime_ts_luma_minus4	ue(v)
(...)	
}	

10

20

【 0 2 3 5】

また、例えば、前記 S P S シンタックスのシンタックス要素のうち前述の実施例に関するシンタックス要素のセマンティック (s e m a n t i c) は次の表 1 8 のように示す。

【 0 2 3 6】

30

40

50

【表 1 8 - 1】

<p>sps_transform_skip_enabled_flag equal to 1 specifies that transform_skip_flag may be present in the transform unit syntax. sps_transform_skip_enabled_flag equal to 0 specifies that transform_skip_flag is not present in the transform unit syntax.</p>	
<p>sps_bdpcm_enabled_flag equal to 1 specifies that intra_bdpcm_flag may be present in the coding unit syntax for intra coding units. sps_bdpcm_enabled_flag equal to 0 specifies that intra_bdpcm_flag is not present in the coding unit syntax for intra coding units. When not present, the value of sps_bdpcm_enabled_flag is inferred to be equal to 0.</p>	10
<p>log2_transform_skip_max_size_minus2 specifies the maximum block size used for transform skip, and shall be in the range of 0 to 3.</p> <p>When not present, the value of log2_transform_skip_max_size_minus2 is inferred to be equal to 0.</p> <p>The variable MaxTsSize is set equal to $1 \ll (\log_2 \text{transform_skip_max_size_minus2} + 2)$.</p>	20
<p>min_qp_prime_ts_luma_minus4 specifies the minimum allowed quantization parameter for transform skip mode in the luma component as follows:</p> $\text{QpPrimeTsMin} = 4 + \text{min_qp_prime_ts_luma_minus4}$	30
<p>min_qp_prime_ts_chroma_minus4 specifies the minimum allowed quantization parameter for transform skip mode in the chroma component as follows:</p> $\text{QpPrimeTsMin} = 4 + \text{min_qp_prime_ts_chroma_minus4}$	40

【0 2 3 7】

【表 18 - 2】

`sps_palette_enabled_flag` equal to 1 specifies that `pred_mode_plt_flag` may be present in the coding unit syntax. `sps_palette_enabled_flag` equal to 0 specifies that `pred_mode_plt_flag` is not present in the coding unit syntax. When `sps_palette_enabled_flag` is not present, it is inferred to be equal to 0.

10

【0238】

前記表 17 及び表 18 を参照すると、SPS において変換スキップ及び/又はパレットコーディングに関連したシンタックス要素が定義されることができ、`sps_transform_skip_enabled_flag`、`sps_bdpcm_enabled_flag`、`log2_transform_skip_max_size_minus_2`、`sps_palette_enabled_flag`、`min_qp_prime_ts_luma_minus_4`、`min_qp_prime_ts_chroma_minus_4` シンタックス要素を含む。

【0239】

ここで、`sps_transform_skip_enabled_flag`、`sps_bdpcm_enabled_flag`、`log2_transform_skip_max_size_minus_2`、`sps_palette_enabled_flag`、`min_qp_prime_ts_luma_minus_4`、`min_qp_prime_ts_chroma_minus_4` シンタックス要素は、前記表 6 ないし表 11 で詳細に説明しているため、本実施例では説明の便宜のために具体的な説明を省略する。

20

【0240】

前記表 17 及び表 18 の実施例に開示されているように、SPS においては変換スキップ及び/又はパレットコーディングに関連するシンタックス要素のうち変換スキップ可用フラグシンタックス要素（例：`sps_transform_skip_enabled_flag`）に対して従属性があるシンタックス要素を定義することができる。例えば、前記表 17 及び表 18 に開示されているように、SPS において変換スキップ可用フラグ（例：`sps_transform_skip_enabled_flag`）の値に基づいて BDPCM が可用であるか否かを示す `sps_bdpcm_enabled_flag` シンタックス要素、変換スキップモードにおいて使用される最大ブロックサイズを示す `log2_transform_skip_max_size_minus_2` シンタックス要素が従属性を有することができる。一例として、変換スキップ可用フラグ（例：`sps_transform_skip_enabled_flag`）の値が 1 である場合、`sps_bdpcm_enabled_flag`、`log2_transform_skip_max_size_minus_2` シンタックス要素がパーシング/シグナリングされる。または、変換スキップ可用フラグ（例：`sps_transform_skip_enabled_flag`）の値が 0 である場合、`sps_bdpcm_enabled_flag`、`log2_transform_skip_max_size_minus_2` シンタックス要素がパーシング/シグナリングされない。また、SPS においては変換スキップ及び/又はパレットコーディングに関連するシンタックス要素のうちパレットコーディング可用フラグシンタックス要素（例：`sps_palette_enabled_flag`）に対して従属性があるシンタックス要素を定義することができる。例えば、前記表 17 及び表 18 に開示されているように、SPS においてパレットコーディング可用フラグ（例：`sps_palette_enabled_flag`）の値に基づいてクロマ成分に対する変換スキップモードにおいて最小量子化パラメータ情報示す `min_qp_prime_ts_chroma_minus_4` シンタックス要素が従属性を有

30

40

50

することができる。一例として、パレットコーディング可用フラグ（例：`sps_palette_enabled_flag`）の値が1である場合、`min_qp_prime_ts_chroma_minus4`シンタックス要素がパーシング/シグナリングされる。または、パレットコーディング可用フラグ（例：`sps_palette_enabled_flag`）の値が0である場合、`min_qp_prime_ts_chroma_minus4`シンタックス要素がパーシング/シグナリングされない。

【0241】

また、SPSにおいては変換スキップ及び/又はパレットコーディングに関連するシンタックス要素のうち変換スキップ可用フラグシンタックス要素（例：`sps_transform_skip_enabled_flag`）及び/又はパレットコーディング可用フラグシンタックス要素（例：`sps_palette_enabled_flag`）に対して従属性があるシンタックス要素を定義することができる。例えば、前記表17及び表18に開示されているように、SPSにおいて変換スキップ可用フラグ（例：`sps_transform_skip_enabled_flag`）及び/又はパレットコーディング可用フラグシンタックス要素（例：`sps_palette_enabled_flag`）の値に基づいてルマ成分に対する変換スキップモードにおいての最小量子化パラメータ情報を示す`min_qp_prime_ts_luma_minus4`シンタックス要素が従属性を有することができる。一例として、変換スキップ可用フラグ（例：`sps_transform_skip_enabled_flag`）の値が1であるか又はパレットコーディング可用フラグ（例：`sps_palette_enabled_flag`）の値が1である場合、`min_qp_prime_ts_luma_minus4`シンタックス要素がパーシング/シグナリングされることができる。

【0242】

以下の図面は本文書の具体的な一例を説明するために作成された。図面に記載された具体的な装置の名称や具体的な用語や名称（例えば、シンタックス/シンタックス要素の名称など）は例示的に提示されたものであるので、本文書の技術的特徴が以下の図面に使用された具体的な名称に制限されない。

【0243】

図10及び図11は、本文書の実施例(ら)によるビデオ/映像エンコーディングの方法及び関連コンポーネントの一例を概略的に示す。

【0244】

図10に開示された方法は、図2又は図11において開示されたエンコーディング装置200により行われることができる。ここで、図11に開示されたエンコーディング装置200は、図2に開示されたエンコーディング装置200を簡略に示したものである。具体的に、図10のステップS1000～S1010は、図2に開示されたレジデュアル処理部230により行われ、図10のステップS1020は、図2に開示されたエントロピーエンコーディング部240により行われる。また、図示されてはいないが、予測サンプルを導出する過程は、前記エンコーディング装置200の予測部220により行われ、前記現在ブロックに対するレジデュアルサンプルと予測サンプルに基づいて前記現在ブロックに対する復元サンプル及び復元ピクチャを生成する過程は、前記エンコーディング装置200の加算部250により行われ、前記現在ブロックに対する予測情報をエンコーディングする過程は、前記エンコーディング装置200のエントロピーエンコーディング部240により行われる。また、図10に開示された方法は、本文書において前述の実施例を含めて行われる。従って、図10においては、前述の実施例と重複する内容に関して具体的な説明を省略するか簡単にする。

【0245】

図10に示すように、エンコーディング装置は、変換スキップ可用情報に基づいて現在ブロックに対して変換を適用するか否かを決定する（S1000）。

【0246】

一実施例として、まず、エンコーディング装置は現在ブロックに対する予測モードを決

10

20

30

40

50

定し、予測サンプルを導出する。例えば、エンコーディング装置は、現在ブロックにインター予測を行うか又はイントラ予測を行うかを決定することができ、また、RDコストに基づいて具体的なインター予測モード又は具体的なイントラ予測モードを決定することができる。または、エンコーディング装置は、現在ブロックにC I I Pモード、I B Cモード、B D P C Mモード又はパレットモードなどに基づいて予測を行うか否かを決定することができる。エンコーディング装置は決定された予測モードにより予測を行って現在ブロックに対する予測サンプルを導出することができる。この時、インター予測又はイントラ予測など本文書において開示された多様な予測方法が適用される。また、エンコーディング装置は現在ブロックに適用された予測に関する情報（例えば、予測モード情報）を生成し、エンコーディングすることができる。

10

【0247】

そして、エンコーディング装置は、現在ブロックに対する原本サンプルと予測サンプルを比較してレジデュアルサンプルを導出することができる。エンコーディング装置は、レジデュアルサンプルに対する変換過程を通じて変換係数を導出することができる。この時、エンコーディング装置は、コーディング効率を考慮して現在ブロックに対する変換適用可否を決定することができる。すなわち、エンコーディング装置は、現在ブロックのレジデュアルサンプルに対して変換が適用されるか否かを決定することができる。

【0248】

例えば、エンコーディング装置は、変換スキップ可用情報に基づいて現在ブロック（レジデュアルサンプル）に変換を適用するか又は変換スキップモードを適用するかを決定する。

20

【0249】

変換スキップ可用情報は、前述のように変換スキップが可用であるか否かに関する情報であり、前記表6ないし表18に開示されているように `sps__transform__skip__enabled__flag` シンタックス要素で示すことができる。例えば、`sps__transform__skip__enabled__flag` の値が1である場合は変換スキップが可用であることを示し、この場合、`transform__skip__flag` が変換ユニットシンタックス (`transform unit syntax`) を介してパーズング/シグナリングされる。ここで、`transform__skip__flag` シンタックス要素は、当該変換ブロック (`associated transform block`) に変換が適用できるか否かを示す。`sps__transform__skip__enabled__flag` の値が0である場合は変換スキップが可用ではないことを示し、この場合、`transform__skip__flag` が変換ユニットシンタックスにおいてパーズング/シグナリングされない。変換スキップ可用情報（例：`sps__transform__skip__enabled__flag`）は、SPSに含まれてデコーディング装置にシグナリングされることができる。すなわち、SPSに含まれた変換スキップ可用情報（例：`sps__transform__skip__enabled__flag`）の値が1であることに基づいて、変換ユニットシンタックスが変換スキップフラグ（例：`transform__skip__flag`）を含む。この時、変換ユニットシンタックスに含まれた変換スキップフラグ（例：`transform__skip__flag`）の値が1である場合、現在ブロックに対して変換が適用されないモード（変換スキップモード）が実行される。または、変換ユニットシンタックスに含まれた変換スキップフラグ（例：`transform__skip__flag`）の値が0である場合、現在ブロックに対して変換が適用される。

30

40

【0250】

例えば、変換スキップ可用情報の値が1である場合（すなわち、変換スキップが可用であることを示す変換スキップ可用情報について）、エンコーディング装置は現在ブロックに変換を適用するか否かを決定することができる。すなわち、エンコーディング装置は、変換スキップ可用情報の値が1であることに基づいて現在ブロックに変換を適用するか否かに関する情報（変換スキップフラグ）を生成し、前記変換スキップフラグを、変換ユニ

50

ットシンタックスを介してシグナリングすることができる。このとき、現在ブロックに変換を適用しない場合（すなわち、変換スキップモードである場合）、エンコーディング装置は、その値が1である変換スキップフラグを生成し、これを変換ユニットシンタックスに含めることができる。または、現在ブロックに変換を適用する場合、エンコーディング装置は、その値が0である変換スキップフラグを生成し、これを変換ユニットシンタックスに含めることができる。

【0251】

エンコーディング装置は変換を適用するか否かに基づいて現在ブロックに対するレジデュアル情報を生成する（S1010）。

【0252】

一実施例として、エンコーディング装置は、現在ブロックのレジデュアルサンプルを導出し、変換を適用するか否かに基づいて現在ブロックのレジデュアルサンプルに変換又は変換スキップを適用してレジデュアル情報を生成する。例えば、変換スキップフラグの値が1である現在ブロックに対するレジデュアルサンプルに対して、エンコーディング装置は変換スキップモードを適用することができる。この場合、エンコーディング装置は現在ブロックのレジデュアルサンプルを変換係数として導出することができる。または、変換スキップフラグの値が0である現在ブロックに対するレジデュアルサンプルに対して、エンコーディング装置は変換を行って変換係数を導出する。エンコーディング装置は、変換スキップ又は変換により導出された変換係数に基づいて量子化過程を行って量子化された変換係数を導出する。エンコーディング装置は、量子化された変換係数に基づいてレジデュアル情報を生成することができる。

【0253】

ここで、レジデュアル情報は、変換及び/又は量子化手順により生成された情報として、量子化された変換係数に関する情報であり、例えば、量子化された変換係数の値情報、位置情報、変換技法、変換カーネル、量子化パラメータなどの情報を含む。

【0254】

エンコーディング装置は映像情報（又は、ビデオ情報）をエンコーディングする（S1020）。

【0255】

ここで、映像情報は前記レジデュアル情報を含む。また、映像情報は、前記予測サンプルの導出に使用された前記予測に関する情報（例えば、予測モード情報）を含む。また、映像情報は、前記変換スキップに関する情報、例えば、変換スキップ可用情報、変換スキップフラグ情報を含む。すなわち、映像情報はエンコーディング過程で導出される様々な情報を含み、このような様々な情報を含んでエンコーディングされる。

【0256】

また、映像情報は、本文書において前述の実施例による様々な情報を含み、前述の表1ないし図18の少なくとも1つに開示された情報を含む。

【0257】

例えば、映像情報はSPS（Sequence Parameter Set）を含む。SPSは、変換スキップ関連情報、パレットコーディング関連情報などを含む。一例として、変換スキップ関連情報は変換スキップ可用情報（例：sps_transform_skip_enabled_flag）、BDPCM可用情報（例：sps_bdpcm_enabled_flag）、変換スキップモードにおいて使用される最大ブロックサイズに関する情報（例：log2_transform_skip_max_size_minus2）、変換スキップモードに対する最小許容量子化パラメータに関連した最小量子化パラメータ情報（例：min_qp_prime_ts_minus4）などを含む。また、一例として、パレットコーディングに関する情報はパレットコーディング可用情報（例：sps_palette_enabled_flag）、変換スキップモードに対する最小許容量子化パラメータに関連した最小量子化パラメータ情報（例：min_qp_prime_ts_minus4）などを含む。

10

20

30

40

50

【0258】

また、例えば、前述のようにSPSに含まれる変換スキップ及び/又はパレットコーディングに関する情報のうち、変換スキップ可用情報（例：`sps__transform__skip__enabled__flag`）に対して従属性がある情報を定義することができる。

【0259】

一例として、SPSにおいて変換スキップ可用情報（例：`sps__transform__skip__enabled__flag`）の値に基づいてBDPCMが可用であるか否かに関するBDPCM可用フラグ情報（例：`sps__bdpcm__enabled__flag`）をパーシング/シグナリングできるように構成することができる。この場合、変換スキップ可用フラグ情報（例：`sps__transform__skip__enabled__flag`）の値が1である場合、BDPCM可用フラグ情報（例：`sps__bdpcm__enabled__flag`）がSPSに含まれ、前記情報（例：`sps__bdpcm__enabled__flag`）がSPSからパーシング/シグナリングされることができる。または、変換スキップ可用フラグ情報（例：`sps__transform__skip__enabled__flag`）の値が0である場合、BDPCM可用フラグ情報（例：`sps__bdpcm__enabled__flag`）がSPSからパーシング/シグナリングされない。

10

【0260】

また、SPSにおいてのBDPCM可用フラグ情報（例：`sps__bdpcm__enabled__flag`）の値に基づいて、現在ブロックに対してBDPCMを適用するか否かに関するBDPCMフラグ情報（例：`intra__bdpcm__flag`）がコーディングユニットシンタックスを介してパーシング/シグナリングされることができる。この場合、BDPCM可用フラグ情報（例：`sps__bdpcm__enabled__flag`）の値が1である場合、BDPCMフラグ情報（例：`intra__bdpcm__flag`）がコーディングユニットシンタックスに含まれ、前記情報（例：`intra__bdpcm__flag`）がコーディングユニットシンタックスからパーシング/シグナリングされる。または、BDPCM可用フラグ情報（例：`sps__bdpcm__enabled__flag`）の値が0である場合、BDPCMフラグ情報（例：`intra__bdpcm__flag`）がコーディングユニットシンタックスからパーシング/シグナリングされない。

20

30

【0261】

または、一例として、SPSにおいて変換スキップ可用情報（例：`sps__transform__skip__enabled__flag`）の値に基づいて変換スキップモードにおいて使用される最大ブロックサイズに関する情報（例：`log2__transform__skip__max__size__minus2`）をパーシング/シグナリングできるように構成することができる。この場合、変換スキップ可用フラグ情報（例：`sps__transform__skip__enabled__flag`）の値が1である場合、変換スキップモードにおいて使用される最大ブロックサイズに関する情報（例：`log2__transform__skip__max__size__minus2`）がSPSに含まれ、前記情報（例：`log2__transform__skip__max__size__minus2`）がSPSからパーシング/シグナリングされる。または、変換スキップ可用フラグ情報（例：`sps__transform__skip__enabled__flag`）の値が0である場合、変換スキップモードにおいて使用される最大ブロックサイズに関する情報（例：`log2__transform__skip__max__size__minus2`）がSPSからパーシング/シグナリングされない。

40

【0262】

または、一例として、SPSにおいて定義された変換スキップ可用フラグ情報（例：`sps__transform__skip__enabled__flag`）の値に基づいて、現在ブロックに対して変換スキップを適用するか否かに関する変換スキップフラグ情報（例：`transform__skip__flag`）が変換ユニットシンタックスを介してパー

50

ジング/シグナリングされることができる。この場合、変換スキップ可用フラグ情報（例：`sps__transform__skip__enabled__flag`）の値が1である場合、変換スキップフラグ情報（例：`transform__skip__flag`）が変換ユニットシンタックスに含まれ、前記情報（例：`transform__skip__flag`）が変換ユニットシンタックスからパーシング/シグナリングされる。または、変換スキップ可用フラグ情報（例：`sps__transform__skip__enabled__flag`）の値が0である場合、変換スキップフラグ情報（例：`transform__skip__flag`）が変換ユニットシンタックスからパーシング/シグナリングされない。

【0263】

また、例えば、前述のようにSPSに含まれる変換スキップ及び/又はパレットコーディングに関する情報のうち、パレットコーディングが可用であるか否かに関するパレットコーディング可用情報（例：`sps__palette__enabled__flag`）に対して従属性がある情報を定義することができる。一例として、SPSにおいて定義されたパレットコーディング可用情報（例：`sps__palette__enabled__flag`）の値に基づいて、現在ブロックに対してパレットコーディング（パレット予測モード）を適用するか否かに関するパレット予測モードのフラグ情報（例：`pred__mode__plt__flag`）がコーディングユニットシンタックスを介してパーシング/シグナリングされることができる。この場合、パレットコーディング可用情報（例：`sps__palette__enabled__flag`）の値が1である場合、パレット予測モードフラグ情報（例：`pred__mode__plt__flag`）がコーディングユニットシンタックスに含まれ、前記情報（例：`pred__mode__plt__flag`）がコーディングユニットシンタックスからパーシング/シグナリングされる。または、パレットコーディング可用情報（例：`sps__palette__enabled__flag`）の値が0である場合、パレット予測モードのフラグ情報（例：`pred__mode__plt__flag`）がコーディングユニットシンタックスからパーシング/シグナリングされない。

【0264】

また、例えば、前述のようにSPSに含まれる変換スキップ及び/又はパレットコーディングに関する情報のうち、変換スキップ可用情報（例：`sps__transform__skip__enabled__flag`）及び/又はパレットコーディング可用情報（例：`sps__palette__enabled__flag`）に対して従属性がある情報を定義することができる。

【0265】

一例として、SPSにおいて変換スキップ可用情報（例：`sps__transform__skip__enabled__flag`）及び/又はパレットコーディング可用情報（例：`sps__palette__enabled__flag`）のうち少なくとも1つに基づいて、変換スキップモードに対する最小許容量子化パラメータに関連した最小量子化パラメータ情報（例：`min__qp__prime__ts__minus4`）をパーシング/シグナリングすることができる。言い換えれば、変換スキップ可用情報（例：`sps__transform__skip__enabled__flag`）の値が1又はパレットコーディング可用情報（例：`sps__palette__enabled__flag`）の値が1である条件に基づいて、最小量子化パラメータ情報（例：`min__qp__prime__ts__minus4`）がSPSに含まれ、前記条件を満足させる場合に限り最低量子化パラメータ情報（例：`min__qp__prime__ts__minus4`）がパーシング/シグナリングされることができる。

【0266】

ここで、最小量子化パラメータ情報（例：`min__qp__prime__ts__minus4`）は前述のように変換スキップモードに対する最小許容量子化パラメータに関する情報であり、これに基づいて現在ブロックに対する量子化パラメータを導出することができる。

【0267】

10

20

30

40

50

例えば、現在ブロックについて変換スキップモードが適用される場合、少なくとも量子化パラメータ情報（例：`min_qp_prime_ts_minus4`）に基づいて現在ブロックに対する量子化パラメータを導出し、前記量子化パラメータに基づいて量子化過程を行って量子化された変換係数を導出することができる。

【0268】

または、例えば、現在ブロックに対してパレットコーディングモードが適用される場合、最小量子化パラメータ情報（例：`min_qp_prime_ts_minus4`）に基づいて現在ブロックのエスケープ値に対する量子化パラメータを導出することができる。この場合、前記量子化パラメータに基づいて前記現在ブロックのエスケープ値に適用して量子化されたエスケープ値（例：`palette_escape_val`）を導出することができる。パレットコーディングモードが適用される過程は、前記表4及び表5に開示されているように行われる。

10

【0269】

前述したような多様な情報を含む映像情報はエンコーディングされてビットストリーム形態で出力される。ビットストリームは、ネットワーク又は（デジタル）格納媒体を介してデコーディング装置に送信される。ここで、ネットワークは、放送網及び/又は通信網などを含み、デジタル格納媒体はUSB、SD、CD、DVD、ブルーレイ、HDD、SSDなどの多様な格納媒体を含む。

【0270】

図12及び図13は、本文書の実施例（ら）によるビデオ/映像デコーディング方法及び関連コンポーネントの一例を概略的に示す。

20

【0271】

図12に開示された方法は、図3又は図13に開示されたデコーディング装置300により行われることができる。ここで、図13に開示されたデコーディング装置300は、図3に開示されたデコーディング装置300を簡略に示したものである。具体的に、図12のステップS1200は図3に開示されたエンロピーデコーディング部310により行われ、図12のステップS1210～S1220は図3に開示されたレジデュアル処理部320により行われ、図12のステップS1230は図3に開示された加算部340により行われる。また、図示されていないが、現在ブロックに対する予測情報を受信する過程は、前記デコーディング装置300のエンロピーデコーディング部310により行われ、現在ブロックの予測サンプルを導出する過程は、前記デコーディング装置300の予測部330により行われる。また、図12で開示された方法は、本文書において前述の実施例を含んで行われることができる。従って、図12では、前述の実施例と重複する内容に関して、具体的な説明を省略するか簡単にする。

30

【0272】

図12に示すように、デコーディング装置はビットストリームから映像情報（又は、ビデオ情報）を受信する（S1200）。

【0273】

一実施例として、デコーディング装置はビットストリームをパーシングして映像復元（又は、ピクチャ復元）に必要な情報（例えば、ビデオ/映像情報）を導出する。この時、映像情報はレジデュアル情報を含み、レジデュアル情報は量子化された変換係数の値情報、位置情報、変換技法、変換カーネル、量子化パラメータなどの情報を含む。また、映像情報は予測に関する情報（例えば、予測モード情報）を含む。また、映像情報は、前記変換スキップに関する情報、例えば、変換スキップ可用情報、変換スキップフラグ情報を含む。すなわち、映像情報は、デコーディング過程に必要な様々な情報を含み、指数ゴロム符号化、CAVLC又はCABACなどのコーディング方法に基づいてデコーディングされることことができる。

40

【0274】

また、映像情報は、本文書において前述の実施例（ら）による様々な情報を含み、前述の表1ないし図18の少なくとも1つに開示された情報を含む。

50

【0275】

例えば、映像情報はSPS (Sequence Parameter Set) を含む。SPSは、変換スキップ関連情報、パレットコーディング関連情報などを含む。一例として、変換スキップ関連情報は変換スキップ可用情報 (例: `sps__transform__skip__enabled__flag`)、BDPCM可用情報 (例: `sps__bdpcm__enabled__flag`)、変換スキップモードにおいて使用される最大ブロックサイズに関する情報 (例: `log2__transform__skip__max__size__minus2`)、変換スキップモードに対する最小許容量子化パラメータに関する最小量子化パラメータ情報 (例: `min__qp__prime__ts__minus4`) などを含む。また、一例として、パレットコーディング関連情報はパレットコーディング可用情報 (例: `sps__palette__enabled__flag`)、変換スキップモードに対する最小許容量子化パラメータに関する最小量子化パラメータ情報 (例: `min__qp__prime__ts__minus4`) などを含む。

10

【0276】

また、例えば、前述のように、SPSに含まれる変換スキップ及び/又はパレットコーディングに関する情報のうち、変換スキップ可用情報 (例: `sps__transform__skip__enabled__flag`) に対して従属性がある情報を定義することができる。

【0277】

一例として、SPSにおいて変換スキップ可用情報 (例: `sps__transform__skip__enabled__flag`) の値に基づいてBDPCMが可用であるか否かに関するBDPCM可用フラグ情報 (例: `sps__bdpcm__enabled__flag`) をパーシング/シグナリングできるように構成することができる。この場合、変換スキップ可用フラグ情報 (例: `sps__transform__skip__enabled__flag`) の値が1である場合、BDPCM可用フラグ情報 (例: `sps__bdpcm__enabled__flag`) がSPSに含まれ、前記情報 (例: `sps__bdpcm__enabled__flag`) がSPSからパーシング/シグナリングされる。または、変換スキップ可用フラグ情報 (例: `sps__transform__skip__enabled__flag`) の値が0である場合、BDPCM可用フラグ情報 (例: `sps__bdpcm__enabled__flag`) がSPSからパーシング/シグナリングされない。

20

30

【0278】

また、SPSにおいてのBDPCM可用フラグ情報 (例: `sps__bdpcm__enabled__flag`) の値に基づいて、現在ブロックに対してBDPCMを適用するか否かに関するBDPCMフラグ情報 (例: `intra__bdpcm__flag`) がコーディングユニットシンタックスを介してパーシング/シグナリングされることができる。この場合、BDPCM可用フラグ情報 (例: `sps__bdpcm__enabled__flag`) の値が1である場合、BDPCMフラグ情報 (例: `intra__bdpcm__flag`) がコーディングユニットシンタックスに含まれ、前記情報 (例: `intra__bdpcm__flag`) がコーディングユニットシンタックスからパーシング/シグナリングされる。または、BDPCM可用フラグ情報 (例: `sps__bdpcm__enabled__flag`) の値が0である場合、BDPCMフラグ情報 (例: `intra__bdpcm__flag`) がコーディングユニットシンタックスからパーシング/シグナリングされない。

40

【0279】

または、一例として、SPSにおいて変換スキップ可用情報 (例: `sps__transform__skip__enabled__flag`) の値に基づいて変換スキップモードにおいて使用される最大ブロックサイズに関する情報 (例: `log2__transform__skip__max__size__minus2`) をパーシング/シグナリングできるように構成することができる。この場合、変換スキップ可用フラグ情報 (例: `sps__transform__skip__enabled__flag`) の値が1である場合、変換スキップモードにおいて使用される最大ブロックサイズに関する情報 (例: `log2__tran`

50

`s f o r m _ s k i p _ m a x _ s i z e _ m i n u s 2`)がSPSに含まれ、前記情報(例:`l o g 2 _ t r a n s f o r m _ s k i p _ m a x _ s i z e _ m i n u s 2`)がSPSからパーシング/シグナリングされる。または、変換スキップ可用フラグ情報(例:`s p s _ t r a n s f o r m _ s k i p _ e n a b l e d _ f l a g`)の値が0である場合、変換スキップモードにおいて使用される最大ブロックサイズに関する情報(例:`l o g 2 _ t r a n s f o r m _ s k i p _ m a x _ s i z e _ m i n u s 2`)がSPSからパーシング/シグナリングされない。

【0280】

または、一例として、SPSにおいて定義された変換スキップ可用フラグ情報(例:`s p s _ t r a n s f o r m _ s k i p _ e n a b l e d _ f l a g`)の値に基づいて、現在ブロックに対して変換スキップを適用するか否かに関する変換スキップフラグ情報(例:`t r a n s f o r m _ s k i p _ f l a g`)が変換ユニットシンタックスを介してパーシング/シグナリングされることができる。この場合、変換スキップ可用フラグ情報(例:`s p s _ t r a n s f o r m _ s k i p _ e n a b l e d _ f l a g`)の値が1である場合、変換スキップフラグ情報(例:`t r a n s f o r m _ s k i p _ f l a g`)が変換ユニットシンタックスに含まれ、前記情報(例:`t r a n s f o r m _ s k i p _ f l a g`)が変換ユニットシンタックスからパーシング/シグナリングされる。または、変換スキップ可用フラグ情報(例:`s p s _ t r a n s f o r m _ s k i p _ e n a b l e d _ f l a g`)の値が0である場合、変換スキップフラグ情報(例:`t r a n s f o r m _ s k i p _ f l a g`)が変換ユニットシンタックスからパーシング/シグナリングされない。

【0281】

また、例えば、前述のように、SPSに含まれる変換スキップ及び/又はパレットコーディングに関する情報のうち、パレットコーディングが可用である否かに関するパレットコーディング可用情報(例:`s p s _ p a l e t t e _ e n a b l e d _ f l a g`)に対して従属性がある情報を定義することができる。一例として、SPSにおいて定義されたパレットコーディング可用情報(例:`s p s _ p a l e t t e _ e n a b l e d _ f l a g`)の値に基づいて、現在ブロックに対してパレットコーディング(パレット予測モード)を適用するか否かに関するパレット予測モードフラグ情報(例:`p r e d _ m o d e _ p l t _ f l a g`)がコーディングユニットシンタックスを介してパーシング/シグナリングされることができる。この場合、パレットコーディング可用情報(例:`s p s _ p a l e t t e _ e n a b l e d _ f l a g`)の値が1である場合、パレット予測モードのフラグ情報(例:`p r e d _ m o d e _ p l t _ f l a g`)がコーディングユニットシンタックスに含まれ、前記情報(例:`p r e d _ m o d e _ p l t _ f l a g`)がコーディングユニットシンタックスからパーシング/シグナリングされる。または、パレットコーディング可用情報(例:`s p s _ p a l e t t e _ e n a b l e d _ f l a g`)の値が0である場合、パレット予測モードのフラグ情報(例:`p r e d _ m o d e _ p l t _ f l a g`)がコーディングユニットシンタックスからパーシング/シグナリングされない。

【0282】

また、例えば、前述のように、SPSに含まれる変換スキップ及び/又はパレットコーディングに関する情報のうち、変換スキップ可用情報(例:`s p s _ t r a n s f o r m _ s k i p _ e n a b l e d _ f l a g`)及び/又はパレットコーディング可用情報(例:`s p s _ p a l e t t e _ e n a b l e d _ f l a g`)に対して従属性がある情報を定義することができる。

【0283】

一例として、SPSにおいて変換スキップ可用情報(例:`s p s _ t r a n s f o r m _ s k i p _ e n a b l e d _ f l a g`)及び/又はパレットコーディング可用情報(例:`s p s _ p a l e t t e _ e n a b l e d _ f l a g`)のうち少なくとも1つに基づいて、変換スキップモードに対する最小許容量子化パラメータに関する最小量子化パラメータ情報(例:`m i n _ q p _ p r i m e _ t s _ m i n u s 4`)をパーシング/シグナリングすることができる。すなわち、変換スキップ可用情報(例:`s p s _ t r a n s f o`

10

20

30

40

50

`rm__skip__enabled__flag`)の値が1又はパレットコーディング可用情報(例:`sps__palette__enabled__flag`)の値が1である条件に基づいて、最小量子化パラメータ情報(例:`min__qp__prime__ts__minus 4`)がSPSに含まれ、前記条件を満足させる場合に限り、最小量子化パラメータ情報(例:`min__qp__prime__ts__minus 4`)がパーシング/シグナリングされることができる。

【0284】

ここで、最小量子化パラメータ情報(例:`min__qp__prime__ts__minus 4`)は、前述のように変換スキップモードに対する最小許容量子化パラメータに関する情報であり、これに基づいて現在ブロックに対する量子化パラメータを導出することができる。

10

【0285】

例えば、現在ブロックに対して変換スキップモードが適用される場合、最小量子化パラメータ情報(例:`min__qp__prime__ts__minus 4`)に基づいて現在ブロックに対する量子化パラメータを導出し、前記量子化パラメータに基づいて逆量子化過程(スケーリング過程)を行って逆量子化された変換係数(スケーリングされた変換係数)を導出する。前記逆量子化された変換係数に基づいて現在ブロックのレジデュアルサンプルを導出することができる。

【0286】

または、例えば、現在ブロックに対してパレットコーディングモードが適用される場合、最小量子化パラメータ情報(例:`min__qp__prime__ts__minus 4`)に基づいて現在ブロックのエスケープ値に対する量子化パラメータを導出することができる。この場合、前記量子化パラメータに基づいて逆量子化(スケーリング過程)を行って前記現在ブロックのエスケープ値を導出することができる。前記エスケープ値に基づいて現在ブロックの復元サンプルを生成することができる。パレットコーディングモードが適用される過程は、前記表4及び表5に開示されているように行われる。

20

【0287】

デコーディング装置は変換スキップ可用情報に基づいて現在ブロックに変換を適用するか否かを決定する(`S1210`)。

【0288】

一実施例として、変換スキップ可用情報を含む映像情報を受信した場合、デコーディング装置は変換スキップ可用情報に基づいて現在ブロックに変換を適用するか又は変換スキップモードを適用するかを決定する。

30

【0289】

変換スキップ可用情報は、前述のように、変換スキップが可用であるか否かに関する情報であり、前記表6ないし表18に開示されているように、`sps__transform__skip__enabled__flag`シンタックス要素で示すことができる。例えば、`sps__transform__skip__enabled__flag`の値が1である場合は変換スキップが可用であることを示し、この場合、`transform__skip__flag`が変換ユニットシンタックス(`transform unit syntax`)を介してパーシング/シグナリングされる。ここで、`transform__skip__flag`シンタックス要素は、当該変換ブロック(`associated transform block`)に変換が適用できるか否かを示す。`sps__transform__skip__enabled__flag`の値が0である場合は変換スキップが可用ではないことを示し、この場合、`transform__skip__flag`が変換ユニットシンタックスにおいてパーシング/シグナリングされない。変換スキップ可用情報(例:`sps__transform__skip__enabled__flag`)はSPSに含まれたエンコーディング装置からデコーディング装置にシグナリングされる。すなわち、SPSに含まれた変換スキップ可用情報(例:`sps__transform__skip__enabled__flag`)の値が1であることに基づいて、変換ユニットシンタックスが変換スキップフラ

40

50

グ（例：`transform__skip__flag`）を含む。この時、変換ユニットシンタックスに含まれた変換スキップフラグ（例：`transform__skip__flag`）の値が1である場合、現在ブロックに対して変換が適用されていないモード（変換スキップモード）が行われる。または、変換ユニットシンタックスに含まれた変換スキップフラグ（例：`transform__skip__flag`）の値が0である場合、現在ブロックに対して変換が適用される。

【0290】

例えば、変換スキップ可用情報の値が1である場合（すなわち、変換スキップが可用であることを示す変換スキップ可用情報について）、デコーディング装置は現在ブロックに変換を適用するか否かを決定することができる。

10

【0291】

デコーディング装置は、変換適用可否及びレジデュアル情報に基づいてレジデュアルサンプルを導出する（S1220）。

【0292】

一例として、デコーディング装置はレジデュアル情報を含む映像情報を受信する。レジデュアル情報は、前述のように、量子化された変換係数の値情報、位置情報、変換技法、変換カーネル、量子化パラメータなどの情報を含む。デコーディング装置は、レジデュアル情報に含まれている量子化された変換係数情報に基づいて現在ブロックに対する量子化された変換係数を導出し、前記量子化された変換係数に基づいて変換係数を導出することができる。そして、前記変換係数に基づいてレジデュアルサンプルを導出することができる。

20

【0293】

例えば、変換スキップ可用情報の値が1である場合（すなわち、変換スキップが可用であることを示す変換スキップ可用情報に対して）、デコーディング装置は現在ブロックに変換を適用するか否かに関する情報（変換スキップフラグ）を変換ユニットシンタックスから取得することができる。この時、デコーディング装置は前記変換スキップフラグ情報に基づいてレジデュアルサンプルを導出することができる。例えば、変換スキップフラグの値が1である現在ブロックに対しては、変換スキップモードを適用することができ、この場合、デコーディング装置は変換係数を現在ブロックのレジデュアルサンプルとして導出することができる。または、変換スキップフラグの値が0である現在ブロックに対しては変換を適用することができ、この場合、デコーディング装置は変換係数に対して逆変換して現在ブロックのレジデュアルサンプルを導出することができる。

30

【0294】

また、変換スキップフラグの値が1（すなわち、変換スキップモード）である現在ブロックに対して、デコーディング装置は最小量子化パラメータ情報に基づいて逆量子化過程で使われる量子化パラメータを導出することができる。そして、デコーディング装置は、前記量子化パラメータに基づいて逆量子化過程を行って逆量子化された変換係数を導出し、前記逆量子化された変換係数に基づいてレジデュアルサンプルを導出することができる。

【0295】

ここで、最小量子化パラメータ情報は、前述のように、変換スキップモードに対する最小許容量子化パラメータに関する情報であり、変換スキップ可用情報（例：`sps__transform__skip__enabled__flag`）及び/又はパレットコーディング可用情報（例：`sps__palette__enabled__flag`）のうち少なくとも1つに基づいて映像情報（例：`SPS`）に含まれる。例えば、変換スキップ可用情報（例：`sps__transform__skip__enabled__flag`）の値が1又はパレットコーディング可用情報（例：`sps__palette__enabled__flag`）の値が1である条件に基づいて、最小量子化パラメータ情報（例：`min__qp__prime__ts__minus4`）がSPSに含まれる。すなわち、前記条件を満足させる場合に限って、最小量子化パラメータ情報（例：`min__qp__prime__ts__minus4`）がパーシング/シグナリングされることができる。

40

50

【0296】

デコーディング装置はレジデュアルサンプルに基づいて復元のサンプルを生成する（S1230）。

【0297】

一実施例として、デコーディング装置は、映像情報に含まれた予測情報（例えば、予測モード情報）に基づいて現在ブロックに対するインター予測を行うか又はイントラ予測を行うかを決定し、前記決定によって予測を行って現在ブロックに対する予測サンプルを導出することができる。そして、デコーディング装置は予測サンプルとレジデュアルサンプルに基づいて復元サンプルを生成する。この時、デコーディング装置は、予測モードに応じて予測サンプルを直ちに復元サンプルとして利用してもよく、又は予測サンプルにレジデュアルサンプルを加えて復元サンプルを生成してもよい。また、復元サンプルに基づいて復元ブロック又は復元ピクチャを導出することができる。以後、デコーディング装置は、必要に応じて主観的／客観的画質を向上させるためにデブロッキングフィルタリング及び／又はSAO手順のようなインループフィルタリングの手順を前記復元ピクチャに適用できることは前述の通りである。

10

【0298】

前述した実施例において、方法は、一連のステップまたはブロックで流れ図に基づいて説明されているが、該当実施例は、ステップの順序に限定されるものではなく、あるステップは、前述と異なるステップと、異なる順序にまたは同時に発生できる。また、当業者であれば、流れ図に示すステップが排他的でなく、他のステップが含まれ、または、流れ図の一つまたはそれ以上のステップが本文書の実施例の範囲に影響を及ぼさずに削除可能であることを理解することができる。

20

【0299】

前述した本文書の実施例による方法は、ソフトウェア形態で具現されることができ、本文書によるエンコーディング装置及び／またはデコーディング装置は、例えば、TV、コンピュータ、スマートフォン、セットトップボックス、ディスプレイ装置などの映像処理を実行する装置に含まれることができる。

【0300】

本文書において、実施例がソフトウェアで具現される時、前述した方法は、前述した機能を遂行するモジュール（過程、機能など）で具現されることができ、モジュールは、メモリに格納され、プロセッサにより実行されることができ、メモリは、プロセッサの内部または外部にあり、よく知られた多様な手段でプロセッサと連結されることができ、プロセッサは、ASIC（application-specific integrated circuit）、他のチップセット、論理回路及び／またはデータ処理装置を含むことができる。メモリは、ROM（read-only memory）、RAM（random access memory）、フラッシュメモリ、メモリカード、格納媒体及び／または他の格納装置を含むことができる。即ち、本文書で説明した実施例は、プロセッサ、マイクロプロセッサ、コントローラまたはチップ上で具現されて実行されることができ、例えば、各図面で示す機能ユニットは、コンピュータ、プロセッサ、マイクロプロセッサ、コントローラまたはチップ上で具現されて実行されることができ、この場合、具現のための情報（例えば、information on instructions）またはアルゴリズムがデジタル格納媒体に格納されることができ、

30

40

【0301】

また、本文書が適用されるデコーディング装置及びエンコーディング装置は、マルチメディア放送送受信装置、モバイル通信端末、ホームシネマビデオ装置、デジタルシネマビデオ装置、監視用カメラ、ビデオ対話装置、ビデオ通信のようなりアルタイム通信装置、モバイルストリーミング装置、格納媒体、カムコーダ、注文型ビデオ（VoD）サービス提供装置、OTTビデオ（Over the top video）装置、インターネットストリーミングサービス提供装置、3次元（3D）ビデオ装置、VR（virtual reality）装置、AR（argumente reality）装置、画像電話ピ

50

デオ装置、運送手段端末（例えば、車両（自律走行車両を含む）端末、飛行機端末、船舶端末等）、及び医療用ビデオ装置などに含まれることができ、ビデオ信号またはデータ信号を処理するために使われることができる。例えば、OTTビデオ（Over the top video）装置として、コンソール、ブルーレイプレーヤ、インターネット接続TV、ホームシアターシステム、スマートフォン、タブレットPC、DVR（Digital Video Recorder）などを含むことができる。

【0302】

また、本文書の実施例（ら）が適用される処理方法は、コンピュータで実行されるプログラムの形態で生産されることができ、コンピュータが読み取り可能な記録媒体に格納されることができ、本文書の実施例（ら）によるデータ構造を有するマルチメディアデータもコンピュータが読み取り可能な記録媒体に格納されることができ、前記コンピュータが読み取り可能な記録媒体は、コンピュータで読みだすことができるデータが格納される全ての種類の格納装置及び分散格納装置を含む。前記コンピュータが読み取り可能な記録媒体は、例えば、ブルーレイディスク（BD）、汎用直列バス（USB）、ROM、PROM、EPROM、EEPROM、RAM、CD-ROM、磁気テープ、フロッピーディスク、及び光学的データ格納装置を含むことができる。また、前記コンピュータが読み取り可能な記録媒体は、搬送波（例えば、インターネットを介した送信）の形態で具現されたメディアを含む。また、エンコーディング方法で生成されたビットストリームが、コンピュータが読み取り可能な記録媒体に格納され、または、有無線通信ネットワークを介して送信されることができ、

【0303】

また、本文書の実施例（ら）は、プログラムコードによるコンピュータプログラム製品で具現されることができ、前記プログラムコードは、本文書の実施例（ら）によりコンピュータで実行されることができ、前記プログラムコードは、コンピュータにより読み取り可能なキャリア上に格納されることができ、

【0304】

図14は、本文書において開示された実施例が適用できるコンテンツストリーミングシステムの例を示す。

【0305】

図14に示すように、本文書の実施例に適用されるコンテンツストリーミングシステムは、大別してエンコーディングサーバ、ストリーミングサーバ、ウェブサーバ、メディア格納所、使用者装置及びマルチメディア入力装置を含む。

【0306】

前記エンコーディングサーバは、スマートフォン、カメラ、カムコーダなどのようなマルチメディア入力装置から入力されたコンテンツをデジタルデータで圧縮してビットストリームを生成し、これを前記ストリーミングサーバに送信する役割をする。他の例として、スマートフォン、カメラ、カムコーダなどのようなマルチメディア入力装置がビットストリームを直接生成する場合、前記エンコーディングサーバは省略されることができ、

【0307】

前記ビットストリームは、本文書の実施例に適用されるエンコーディング方法またはビットストリーム生成方法により生成されることができ、前記ストリーミングサーバは、前記ビットストリームを送信または受信する過程で一時的に前記ビットストリームを格納することができる。

【0308】

前記ストリーミングサーバは、ウェブサーバを介したユーザ要請に基づいてマルチメディアデータをユーザ装置に送信し、前記ウェブサーバは、ユーザにどのようなサービスがあるかを知らせる媒介体役割をする。ユーザが前記ウェブサーバに所望のサービスを要請すると、前記ウェブサーバは、これをストリーミングサーバに伝達し、前記ストリーミングサーバは、ユーザにマルチメディアデータを送信する。このとき、前記コンテンツストリーミングシステムは、別途の制御サーバを含むことができ、この場合、前記制御サーバ

10

20

30

40

50

は、前記コンテンツストリーミングシステム内の各装置間の命令/応答を制御する役割をする。

【0309】

前記ストリーミングサーバは、メディア格納所及び/またはエンコーディングサーバからコンテンツを受信することができる。例えば、前記エンコーディングサーバからコンテンツを受信ようになる場合、前記コンテンツをリアルタイムで受信することができる。この場合、円滑なストリーミングサービスを提供するために、前記ストリーミングサーバは、前記ビットストリームを一定時間の間格納することができる。

【0310】

前記ユーザ装置の例として、携帯電話、スマートフォン (smart phone)、ノートブックコンピュータ (laptop computer)、デジタル放送用端末、PDA (personal digital assistants)、PMP (portable multimedia player)、ナビゲーション、スレートPC (slate PC)、タブレットPC (tablet PC)、ウルトラブック (ultrabook)、ウェアラブルデバイス (wearable device、例えば、ウォッチ型端末 (smartwatch)、グラス型端末 (smart glass)、HMD (head mounted display)、デジタルTV、デスクトップコンピュータ、デジタルサイニジがある。

10

【0311】

前記コンテンツストリーミングシステム内の各サーバは、分散サーバとして運営されることができ、この場合、各サーバで受信するデータは、分散処理されることができ。

20

【0312】

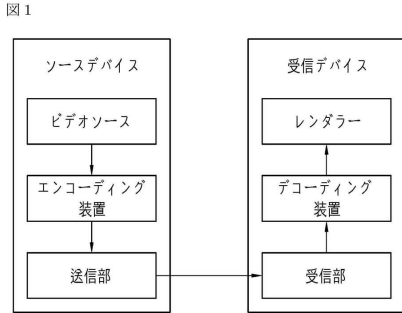
本明細書に記載された請求項は、多様な方式で組み合わせることができる。例えば、本明細書の方法請求項の技術的特徴が組み合わせて装置で具現されることができ、本明細書の装置請求項の技術的特徴が組み合わせて方法で具現されることができ、また、本明細書の方法請求項の技術的特徴と装置請求項の技術的特徴が組み合わせて装置で具現されることができ、本明細書の方法請求項の技術的特徴と装置請求項の技術的特徴が組み合わせて方法で具現されることができ。

30

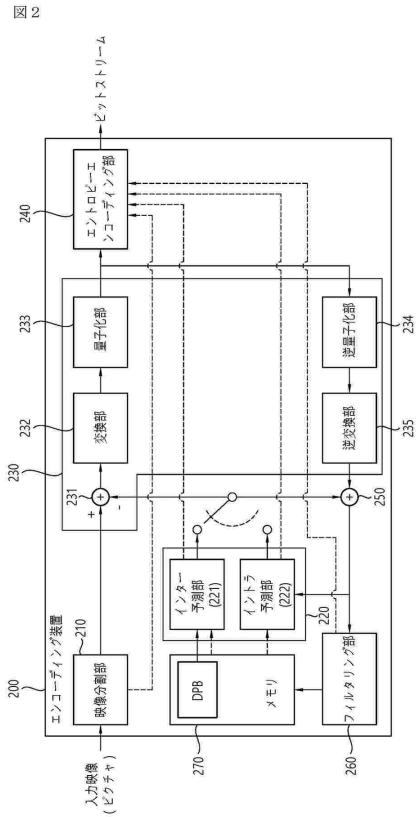
40

50

【 図面 】
【 図 1 】



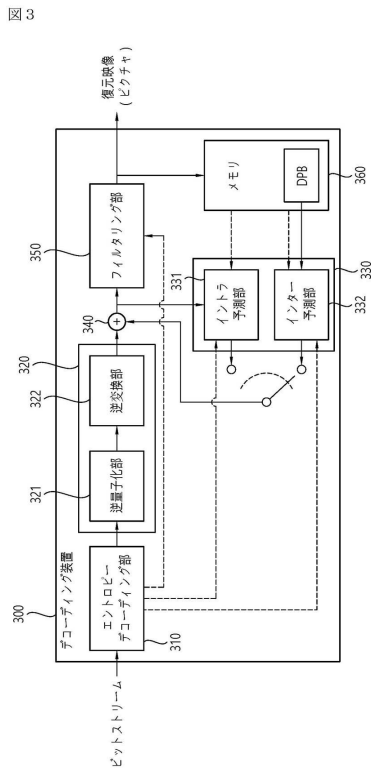
【 図 2 】



10

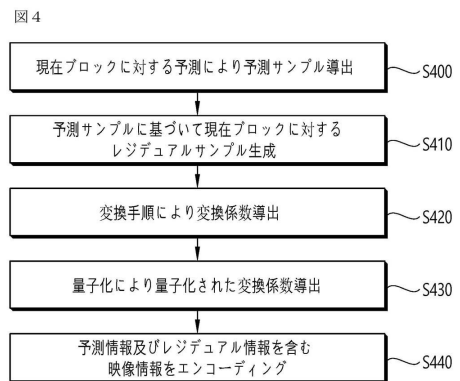
20

【 図 3 】



30

【 図 4 】

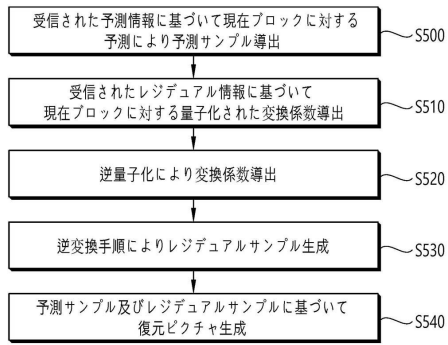


40

50

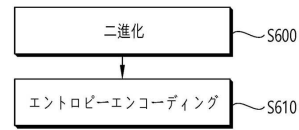
【図 5】

図 5



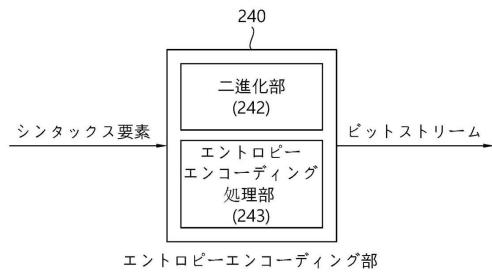
【図 6】

図 6



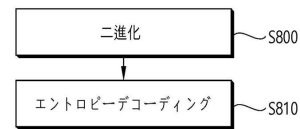
【図 7】

図 7



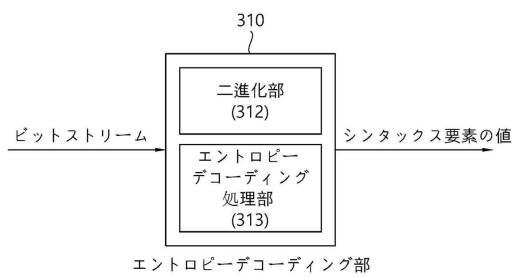
【図 8】

図 8



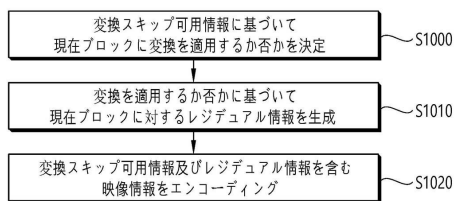
【図 9】

図 9



【図 10】

図 10



10

20

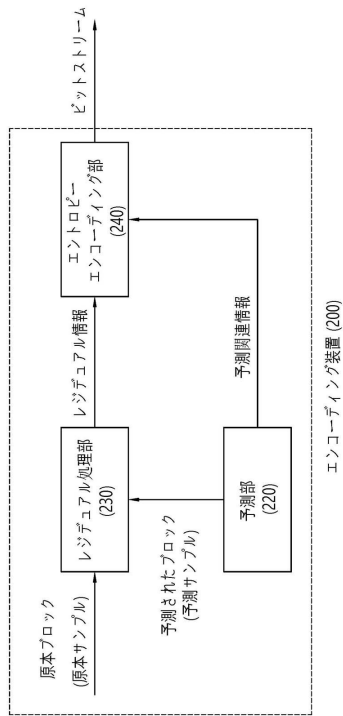
30

40

50

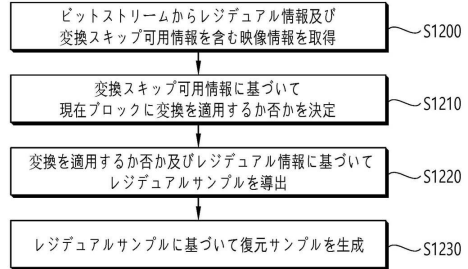
【 図 1 1 】

図 1 1



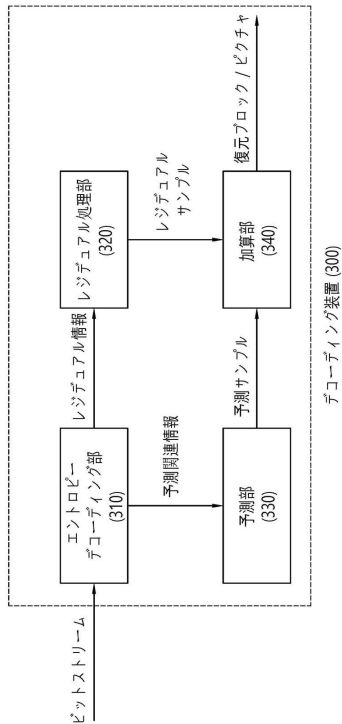
【 図 1 2 】

図 1 2



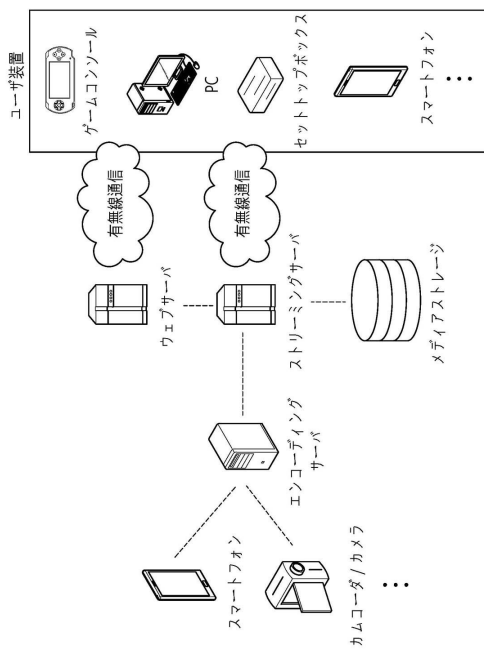
【 図 1 3 】

図 1 3



【 図 1 4 】

図 1 4



10

20

30

40

50

フロントページの続き

- 弁理士 河合 章
 (74)代理人 100114018
 弁理士 南山 知広
 (74)代理人 100159259
 弁理士 竹本 実
 (72)発明者 ユ ソンミ
 大韓民国，ソウル 0 6 7 7 2，ソチョ - ク，ヤンジェ - デロ 1 1 - ギル，1 9，エルジー エレ
 クトロニクス インコーポレイティド，アイピー センター
 (72)発明者 チェ チョンア
 大韓民国，ソウル 0 6 7 7 2，ソチョ - ク，ヤンジェ - デロ 1 1 - ギル，1 9，エルジー エレ
 クトロニクス インコーポレイティド，アイピー センター
 (72)発明者 チェ チャンウォン
 大韓民国，ソウル 0 6 7 7 2，ソチョ - ク，ヤンジェ - デロ 1 1 - ギル，1 9，エルジー エレ
 クトロニクス インコーポレイティド，アイピー センター
 審査官 白川 瑞樹
 (56)参考文献 国際公開第 2 0 2 1 / 0 0 4 8 8 7 (W O ， A 1)
 特開 2 0 1 4 - 0 6 4 2 4 9 (J P ， A)
 米国特許出願公開第 2 0 1 6 / 0 2 6 9 7 3 0 (U S ， A 1)
 HSIANG, Shih-Ta et al. , AHG9: High-Level Syntax Related to Transform Skip Mode , JVET-
 Q0183 (version 1) , ITU , 2019年12月31日 , pp.1-6 , [online] , [retrieved on 2023-02-28]
 , Retrieved from the Internet: URL: https://jvet-experts.org/doc_end_user/documents/17_Brussels/wg11/JVET-Q0183-v1.zip
 CHANG, Yao-Jen et al. , AHG17: High Level Syntax Cleanup on the Syntax Elements of Tran
 sform Skip , JVET-P0430 (version 2) , ITU , 2019年10月04日 , pp.1-4 , [online] , [retriev
 ed on 2023-02-28] , Retrieved from the Internet: URL: [https://jvet-experts.org/doc_end_](https://jvet-experts.org/doc_end_user/documents/16_Geneva/wg11/JVET-P0430-v2.zip)
[user/documents/16_Geneva/wg11/JVET-P0430-v2.zip](https://jvet-experts.org/doc_end_user/documents/16_Geneva/wg11/JVET-P0430-v2.zip)
 BROSS, Benjamin et al. , Versatile Video Coding (Draft 6) , JVET-O2001 (version 14) , ITU
 , 2019年07月31日 , pp.1, 37-43, 73-81, 89, 151, 282, 283 , [online] , [retrieved on 202
 3-02-28] , Retrieved from the Internet: URL: [https://jvet-experts.org/doc_end_user/docu](https://jvet-experts.org/doc_end_user/documents/15_Gothenburg/wg11/JVET-O2001-v14.zip)
[ments/15_Gothenburg/wg11/JVET-O2001-v14.zip](https://jvet-experts.org/doc_end_user/documents/15_Gothenburg/wg11/JVET-O2001-v14.zip)
 CHAO, Yung-Hsuan et al. , Non-CE8: Minimum QP for Escape Mode in Palette , JVET-P047
 4 (version 1) , ITU , 2019年09月25日 , pp.1-3 , [online] , [retrieved on 2023-02-28] , Ret
 rieved from the Internet: URL: [https://jvet-experts.org/doc_end_user/documents/16_Gen](https://jvet-experts.org/doc_end_user/documents/16_Geneva/wg11/JVET-P0474-v1.zip)
[eva/wg11/JVET-P0474-v1.zip](https://jvet-experts.org/doc_end_user/documents/16_Geneva/wg11/JVET-P0474-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