

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5893747号
(P5893747)

(45) 発行日 平成28年3月23日(2016.3.23)

(24) 登録日 平成28年3月4日(2016.3.4)

(51) Int.Cl.

F I

HO 4 N 19/70	(2014.01)	HO 4 N 19/70
HO 4 N 19/119	(2014.01)	HO 4 N 19/119
HO 4 N 19/174	(2014.01)	HO 4 N 19/174
HO 4 N 19/96	(2014.01)	HO 4 N 19/96
HO 4 N 19/46	(2014.01)	HO 4 N 19/46

請求項の数 31 (全 42 頁)

(21) 出願番号 特願2014-540073 (P2014-540073)
 (86) (22) 出願日 平成24年11月1日(2012.11.1)
 (65) 公表番号 特表2014-535220 (P2014-535220A)
 (43) 公表日 平成26年12月25日(2014.12.25)
 (86) 国際出願番号 PCT/US2012/063027
 (87) 国際公開番号 W02013/067158
 (87) 国際公開日 平成25年5月10日(2013.5.10)
 審査請求日 平成26年7月2日(2014.7.2)
 (31) 優先権主張番号 61/555,932
 (32) 優先日 平成23年11月4日(2011.11.4)
 (33) 優先権主張国 米国(US)
 (31) 優先権主張番号 61/557,259
 (32) 優先日 平成23年11月8日(2011.11.8)
 (33) 優先権主張国 米国(US)

(73) 特許権者 595020643
 クォアルコム・インコーポレイテッド
 QUALCOMM INCORPORATED
 アメリカ合衆国、カリフォルニア州 92
 121-1714、サン・ディエゴ、モア
 ハウス・ドライブ 5775
 (74) 代理人 100108855
 弁理士 蔵田 昌俊
 (74) 代理人 100109830
 弁理士 福原 淑弘
 (74) 代理人 100103034
 弁理士 野河 信久
 (74) 代理人 100075672
 弁理士 峰 隆司

最終頁に続く

(54) 【発明の名称】 コード化スライスNALユニット中のセグメントのパディング

(57) 【特許請求の範囲】

【請求項1】

ビデオデータを符号化するための方法であって、前記方法は、

ピクチャーを複数のピクチャーパーティションに分割することと、ここで、前記ピクチャーは複数のツリーブロックを有し、前記ピクチャーパーティションは前記ピクチャーの前記ツリーブロックのオーバーラップしないサブセットに関連付けられる、

前記ピクチャーのスライスに関連付けられた前記ツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ(NAL)ユニットを生成することと、ここで、前記ツリーブロックの前記符号化表現は、前記コード化スライスNALユニット内で前記ツリーブロックが属する前記ピクチャーパーティションに従って、セグメントにグループ化され、

ここにおいて、前記セグメントのそれぞれがバイト境界上で開始するように前記セグメントのうちの1つまたは複数がパディングされ、

前記セグメントは、所与のセグメントを含み、

前記コード化スライスNALユニットを生成することは、コーディング順序の次のツリーブロックが、前記スライス内にあり、前記所与のセグメントに関連付けられるピクチャーパーティションとは異なるピクチャーパーティションに関連付けられる場合に、前記所与のセグメントにビットを付加するパディング演算を実行することを含む、方法。

【請求項2】

前記コード化スライスNALユニットを生成することは、前記セグメントのうちの1つまたは複数のためのエン트리ポイントを示すスライスヘッダを生成することを含む、請求項1に記載の方法。

【請求項3】

前記セグメントのための前記エン트리ポイントは、前記セグメントのバイトオフセットを示す、請求項2に記載の方法。

【請求項4】

前記ピクチャーパーティションは、タイルまたはウェーブフロント並列処理(WPP)ウェーブである、請求項1に記載の方法。

【請求項5】

前記ピクチャーに関連付けられたパラメータセットを生成することをさらに含み、前記パラメータセットは、第1の値を有するフラグを含み、前記第1の値は、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの前記1つまたは複数がパディングされることを示し、

前記フラグが第2の値を有するとき、前記セグメントはバイト境界において開始することとも開始しないこともある、請求項1に記載の方法。

【請求項6】

前記ピクチャーパーティションは、タイルであり、

前記コード化スライスNALユニットを生成することは、前記タイルが互いから独立していると判定した後にのみ、前記セグメントがバイト境界において開始することを保証するパディング演算を実行することを含む、請求項1に記載の方法。

【請求項7】

ビデオデータを復号する方法であって、前記方法は、

ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ(NAL)ユニットを記憶することと、ここで、前記ピクチャーは複数のピクチャーパーティションに区分され、前記ツリーブロックの前記符号化表現は、前記ツリーブロックが属する前記ピクチャーパーティションに従って、セグメントにグループ化され、ここにおいて、

前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの1つまたは複数がパディングされ、

前記セグメントは、所与のセグメントを含み、

前記コード化スライスNALユニットは、コーディング順序の次のツリーブロックが、前記スライス内にあり、前記所与のセグメントに関連付けられるピクチャーパーティションとは異なるピクチャーパーティションに関連付けられる場合に、前記所与のセグメントに付加されるビットを含む、

前記ツリーブロックの前記符号化表現を復号することと、を含む、方法。

【請求項8】

前記コード化スライスNALユニットは、前記セグメントのうちの1つまたは複数のためのエン트리ポイントを示すスライスヘッダを含む、請求項7に記載の方法。

【請求項9】

前記セグメントのための前記エン트리ポイントは、前記セグメントのバイトオフセットを示す、請求項8に記載の方法。

【請求項10】

前記ピクチャーパーティションは、タイルまたはウェーブフロント並列処理(WPP)ウェーブである、請求項7に記載の方法。

【請求項11】

前記ピクチャーに関連付けられたパラメータセットを記憶することをさらに含み、前記パラメータセットは、第1の値を有するフラグを含み、前記第1の値は、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの前記1つまたは

10

20

30

40

50

複数のパディングされることを示し、

前記フラグが第2の値を有するとき、前記セグメントはバイト境界において開始することとも開始しないこともある、請求項7に記載の方法。

【請求項12】

前記ピクチャーパーティションは、タイルであり、

前記タイルが互いから独立しているとき、前記セグメントがバイト境界において開始するように前記セグメントのうちの1つまたは複数のパディングされる、請求項7に記載の方法。

【請求項13】

前記ツリーブロックの前記符号化表現を復号することは、前記セグメントのうちの2つ以上の中の前記ツリーブロックの前記符号化表現を並列に復号することを含む、請求項7に記載の方法。

【請求項14】

ビデオデータを符号化するビデオ符号化デバイスであって、前記ビデオ符号化デバイスは、下記を含む、

前記ビデオデータを記憶するように構成されたデータ記憶媒体と、

1つまたは複数のプロセッサであって、

前記ビデオデータのピクチャーを複数のピクチャーパーティションに分割することと、ここで、前記ピクチャーは複数のツリーブロックを有し、前記ピクチャーパーティションは前記ピクチャーの前記ツリーブロックのオーバーラップしないサブセットに関連付けられる、

前記ピクチャーのスライスに関連付けられた前記ツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ(NAL)ユニットを生成することと、ここで、前記ツリーブロックの前記符号化表現は、前記コード化スライスNALユニット内で前記ツリーブロックが属する前記ピクチャーパーティションに従って、セグメントにグループ化され、ここにおいて、

前記セグメントのそれぞれがバイト境界上で開始するように前記セグメントのうちの1つまたは複数のパディングされ、

前記セグメントは、所与のセグメントを含み、

前記1つまたは複数のプロセッサが、コーディング順序の次のツリーブロックが、前記スライス内にあり、前記所与のセグメントに関連付けられるピクチャーパーティションとは異なるピクチャーパーティションに関連付けられる場合に、前記所与のセグメントにビットを付加するパディング演算を実行するように構成される、

を行うように構成された前記1つまたは複数のプロセッサ。

【請求項15】

前記1つまたは複数のプロセッサは、前記コード化スライスNALユニットが、前記セグメントのうちの1つまたは複数のためのエン트리ポイントを示すスライスヘッダを含むように、前記コード化スライスNALユニットを生成するように構成された、請求項14に記載のビデオ符号化デバイス。

【請求項16】

前記セグメントのための前記エン트리ポイントは、前記セグメントのバイトオフセットを示す、請求項15に記載のビデオ符号化デバイス。

【請求項17】

前記ピクチャーパーティションは、タイルまたはウェーブフロント並列処理(WPP)ウェーブである、請求項14に記載のビデオ符号化デバイス。

【請求項18】

前記1つまたは複数のプロセッサは、前記ピクチャーに関連付けられたパラメータセットを生成するように構成され、前記パラメータセットは、第1の値を有するフラグを含み、前記第1の値は、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの前記1つまたは複数のパディングされることを示し、

前記フラグが第2の値を有するとき、前記セグメントはバイト境界において開始することとも開始しないこともある、請求項14に記載のビデオ符号化デバイス。

【請求項19】

前記ピクチャーパーティションは、タイルであり、

1つまたは複数のプロセッサは、前記タイルが互いから独立していると判定した後のみ、前記セグメントがバイト境界において開始することを保証するパディング演算を実行するように構成された、請求項14に記載のビデオ符号化デバイス。

【請求項20】

前記ビデオ符号化デバイスは、モバイルコンピューティングデバイスである、請求項14に記載のビデオ符号化デバイス。

【請求項21】

ビデオデータを復号するビデオ復号デバイスであって、前記ビデオ復号デバイスは、下記を含む、

ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ（NAL）ユニットを記憶するメモリであって、ここで、前記ピクチャーは複数のピクチャーパーティションに分割され、前記ツリーブロックの前記符号化表現は前記ツリーブロックが属する前記ピクチャーパーティションに従って、セグメントにグループ化され、ここにおいて、

前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの1つまたは複数のパディングされ、

前記セグメントは、所与のセグメントを含み、

前記コード化スライスNALユニットは、コーディング順序の次のツリーブロックが、前記スライス内にあり、前記所与のセグメントに関連付けられるピクチャーパーティションとは異なるピクチャーパーティションに関連付けられる場合に、前記所与のセグメントに付加されるビットを含む、

メモリと、

前記ツリーブロックの前記符号化表現を復号するように構成された1つまたは複数のプロセッサ。

【請求項22】

前記コード化スライスNALユニットは、前記セグメントのうちの1つまたは複数のためのエン트리ポイントを示すスライスヘッダを含む、請求項21に記載のビデオ復号デバイス。

【請求項23】

前記セグメントのための前記エン트리ポイントは、前記セグメントのバイトオフセットを示す、請求項22に記載のビデオ復号デバイス。

【請求項24】

前記ピクチャーパーティションは、タイルまたはウェーブフロント並列処理（WPP）ウェーブである、請求項21に記載のビデオ復号デバイス。

【請求項25】

前記メモリは、前記ピクチャーに関連付けられたパラメータセットを記憶し、前記パラメータセットは、第1の値を有するフラグを含み、前記第1の値は、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの前記1つまたは複数のパディングされることを示し、

前記フラグが第2の値を有するとき、前記セグメントはバイト境界において開始することとも開始しないこともある、請求項21に記載のビデオ復号デバイス。

【請求項26】

前記ビデオ復号デバイスは、モバイルコンピューティングデバイスである、請求項21に記載のビデオ復号デバイス。

【請求項27】

前記1つまたは複数のプロセッサが、前記セグメントのうちの2つ以上の中の前記ツリ

10

20

30

40

50

ーブロックの前記符号化表現を並列に復号するように構成された、請求項 2 1 に記載のビデオ復号デバイス。

【請求項 2 8】

1 つまたは複数のプロセッサによって実行されたとき、

ピクチャーを複数のピクチャーパーティションに分割することと、ここで、前記ピクチャーは複数のツリーブロックを有し、前記ピクチャーパーティションは前記ピクチャーの前記ツリーブロックのオーバーラップしないサブセットに関連付けられる、

前記ピクチャーのスライスに関連付けられた前記ツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ (NAL) ユニットを生成することと、ここで、前記ツリーブロックの前記符号化表現は、前記コード化スライスNALユニット内で前記ツリーブロックが属する前記ピクチャーパーティションに従って、セグメントにグループ化され、ここにおいて、

10

前記セグメントのそれぞれがバイト境界上で開始するように前記セグメントのうちの1 つまたは複数がパディングされ、

前記セグメントは、所与のセグメントを含み、

前記 1 つまたは複数のプロセッサは、コーディング順序の次のツリーブロックが、前記スライス内にあり、前記所与のセグメントに関連付けられるピクチャーパーティションとは異なるピクチャーパーティションに関連付けられる場合に、前記所与のセグメントにビットを付加するパディング演算を実行するように構成される、

を行うようにビデオ符号化デバイスを構成する、命令を記憶する非一時的なコンピュータ可読記憶媒体。

20

【請求項 2 9】

1 つまたは複数のプロセッサによって実行されたとき、

ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ (NAL) ユニットを記憶することと、ここで、前記ピクチャーは複数のピクチャーパーティションに分割され、前記ツリーブロックの前記符号化表現は前記ツリーブロックが属する前記ピクチャーパーティションに従って、セグメントにグループ化され、ここにおいて、

前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの1 つまたは複数がパディングされ、

30

前記セグメントは、所与のセグメントを含み、

前記コード化スライスNALユニットは、コーディング順序の次のツリーブロックが、前記スライス内にあり、前記所与のセグメントに関連付けられるピクチャーパーティションとは異なるピクチャーパーティションに関連付けられる場合に、前記所与のセグメントに付加されたビットを含む、

前記ツリーブロックの前記符号化表現を復号することと、
を行うようにビデオ復号デバイスを構成する、命令を記憶する非一時的なコンピュータ可読記憶媒体。

【請求項 3 0】

ビデオデータを符号化するビデオ符号化デバイスであって、前記ビデオ符号化デバイスは、

40

ピクチャーを複数のピクチャーパーティションに分割するための手段と、ここで、前記ピクチャーは複数のツリーブロックを有し、前記ピクチャーパーティションは前記ピクチャーの前記ツリーブロックのオーバーラップしないサブセットに関連付けられる、

前記ピクチャーのスライスに関連付けられた前記ツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ (NAL) ユニットを生成するための手段と、ここで、前記ツリーブロックの前記符号化表現は、前記コード化スライスNALユニット内で前記ツリーブロックが属する前記ピクチャーパーティションに従って、セグメントにグループ化され、ここにおいて、

前記セグメントのそれぞれがバイト境界上で開始するように前記セグメントのうちの

50

1 つまたは複数がパディングされ、

前記セグメントは、所与のセグメントを含み、

前記コード化スライスNALユニットを生成することは、コーディング順序の次のツリーブロックが、前記スライス内にあり、前記所与のセグメントに関連付けられるピクチャーパーティションとは異なるピクチャーパーティションに関連付けられる場合に、前記所与のセグメントにビットを付加するパディング演算を実行することを含む、
を含む、ビデオ符号化デバイス。

【請求項 31】

ビデオデータを復号するビデオ復号デバイスであって、前記ビデオ復号デバイスは、下記を含む、

10

ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ(NAL)ユニットを記憶するための手段であって、ここで、前記ピクチャーは複数のピクチャーパーティションに分割され、前記ツリーブロックの前記符号化表現は前記ツリーブロックが属する前記ピクチャーパーティションに従って、セグメントにグループ化され、ここにおいて、

前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの1つまたは複数がパディングされ、

前記セグメントは、所与のセグメントを含み、

前記コード化スライスNALユニットは、コーディング順序の次のツリーブロックが、前記スライス内にあり、前記所与のセグメントに関連付けられるピクチャーパーティションとは異なるピクチャーパーティションに関連付けられる場合に、前記所与のセグメントに付加されるビットを含む、

20

手段と、

前記ツリーブロックの前記符号化表現を復号するための手段。

【発明の詳細な説明】

【技術分野】

【0001】

本出願は、その内容全体が参照により本明細書に組み込まれる、2011年11月8日に出願された米国仮出願第61/557,259号の利益を主張する。本出願はまた、2011年11月4日に出願された米国仮出願第61/555,932号の利益を主張する。

30

【0002】

本開示はビデオコーディング(video coding)(すなわち、ビデオデータの符号化(encoding)または復号(decoding))に関する。

【背景技術】

【0003】

デジタルビデオ機能は、デジタルテレビジョン、デジタルダイレクトブロードキャストシステム、ワイヤレスブロードキャストシステム、携帯情報端末(PDA)、ラップトップまたはデスクトップコンピュータ、デジタルカメラ、デジタル記録デバイス、デジタルメディアプレーヤ、ビデオゲームデバイス、ビデオゲームコンソール、セルラー電話または衛星無線電話、ビデオ遠隔会議デバイスなどを含む、広範囲にわたるデバイスに組み込まれ得る。デジタルビデオデバイスは、デジタルビデオ情報をより効率的に送信、受信および記憶するための、たとえばMPEG-2、MPEG-4、ITU-T H.263、ITU-T H.264/MPEG-4, Part 10, Advanced Video Coding(AVC)によって定義された規格、現在開発中の高効率ビデオコーディング(HEVC: High Efficiency Video Coding)規格、およびそのような規格の拡張に記載されているビデオ圧縮技法などのような、ビデオ圧縮技法を実装する。

40

【0004】

ビデオ圧縮技法は、ビデオシーケンスに固有の冗長性を低減または除去するために空間的(イントラピクチャー)予測および/または時間的(インターピクチャー)予測を実行

50

する。ブロックベースのビデオコーディングの場合、ビデオスライスが、ツリーブロック、コーディングユニット(CU)および/またはコーディングノードと呼ばれることもあるビデオブロックに区分(partitioned)され得る。ピクチャーのイントラコード化(I)スライス(intra-coded(I) slice)中のビデオブロックは、同じピクチャー中の隣接ブロック中の参照サンプルに対する空間的予測を使用して符号化される。ピクチャーのインターコード化(PまたはB)スライス(inter-coded(P or B) slice)中のビデオブロックは、同じピクチャー中の隣接ブロック中の参照サンプルに対する空間的予測、または他の参照ピクチャー中の参照サンプルに対する時間的予測を使用し得る。ピクチャーはフレームと呼ばれることがあり、参照ピクチャーは参照フレームと呼ばれることがある。

【発明の概要】

10

【0005】

概して、本開示では、ビデオデータを符号化および復号するための技法について説明する。ビデオエンコーダは、ピクチャーを複数のピクチャーパーティションに分割(divide)し得る。ピクチャーパーティションは、ピクチャーのツリーブロックのオーバーラップしないサブセットを含む。例示的なタイプのピクチャーパーティションは、タイル(tiles)およびウェイブフロント並列処理(WPP)ウェイブ(wavefront parallel processing waves)を含む。ビデオエンコーダは、ピクチャーのスライスに関連付けられたツリーブロックの符号化表現(encoded representations)を含むコード化スライスネットワーク抽象化レイヤ(NAL)ユニット(coded slice network abstraction layer unit)を生成し得る。ビデオエンコーダは、コード化ツリーブロック(coded treeblocks)が、それらツリーブロックが属するピクチャーパーティションによってコード化スライスNALユニット内でグループ化されるように、コード化スライスNALユニットを生成する。ビデオエンコーダは、セグメントのそれぞれがバイト境界上で開始するようにセグメントのうちの1つまたは複数をパディングし得る。ビデオデコーダは、コード化スライスNALユニットのコード化ツリーブロックを復号し得る。

20

【0006】

一態様では、本開示では、ビデオデータを符号化するための方法について説明する。本方法は、ピクチャーを複数のピクチャーパーティションに分割することを含む。ピクチャーは、複数のツリーブロックを有する。ピクチャーパーティションは、ピクチャーのツリーブロックのオーバーラップしないサブセットに関連付けられる。本方法は、ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスNALユニットを生成することをも含み、ここで、ツリーブロックの符号化表現は、コード化スライスNALユニット内で上記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化され、セグメントのそれぞれがバイト境界上で開始するように、セグメントのうちの1つまたは複数がパディングされる。

30

【0007】

別の態様では、本開示では、ビデオデータを復号する方法について説明する。本方法は、ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスNALユニットを記憶することを含む。ピクチャーは、複数のピクチャーパーティションに区分される。ツリーブロックの符号化表現は、上記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化される。セグメントのそれぞれがバイト境界において開始するように、セグメントのうちの1つまたは複数がパディングされる。本方法は、ツリーブロックの符号化表現を復号することをも含む。

40

【0008】

別の態様では、本開示では、ビデオデータを符号化するビデオ符号化デバイスについて説明する。本ビデオ符号化デバイスは、ピクチャーを複数のピクチャーパーティションに分割するように構成された1つまたは複数のプロセッサを含む。ピクチャーは、複数のツリーブロックを有する。ピクチャーパーティションは、ピクチャーのツリーブロックのオーバーラップしないサブセットに関連付けられる。1つまたは複数のプロセッサは、ピク

50

チャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスNALユニットを生成するようにも構成される。ツリーブロックの符号化表現は、コード化スライスNALユニット内で上記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化される。セグメントのそれぞれがバイト境界上で開始するように、セグメントのうちの1つまたは複数がパディングされる。

【0009】

別の態様では、本開示では、ビデオデータを復号するビデオ復号デバイスについて説明する。本ビデオ復号デバイスは、ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスNALユニットを記憶するメモリを含む。ピクチャーは、複数のピクチャーパーティションに分割される。ツリーブロックの符号化表現は、上記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化される。セグメントのそれぞれがバイト境界において開始するように、セグメントのうちの1つまたは複数がパディングされる。本ビデオ復号デバイスは、ツリーブロックの符号化表現を復号するように構成された1つまたは複数のプロセッサをも含む。

10

【0010】

別の態様では、本開示では、1つまたは複数のプロセッサによって実行されたとき、ピクチャーを複数のピクチャーパーティションに分割するようにビデオ符号化デバイスを構成する、命令を記憶する1つまたは複数のコンピュータ可読記憶媒体を含むコンピュータプログラム製品について説明する。ピクチャーは、複数のツリーブロックを有する。ピクチャーパーティションは、ピクチャーのツリーブロックのオーバーラップしないサブセットに関連付けられる。上記命令は、ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスNALユニットを生成するようにもビデオ符号化デバイスを構成する。ツリーブロックの符号化表現は、コード化スライスNALユニット内で上記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化される。セグメントのそれぞれがバイト境界上で開始するように、セグメントのうちの1つまたは複数がパディングされる。

20

【0011】

別の態様では、本開示では、1つまたは複数のプロセッサによって実行されたとき、ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスNALユニットを記憶するようにビデオ復号デバイスを構成する、命令を記憶する1つまたは複数のコンピュータ可読記憶媒体を含むコンピュータプログラム製品について説明する。ピクチャーは、複数のピクチャーパーティションに分割される。ツリーブロックの符号化表現は、上記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化される。セグメントのそれぞれがバイト境界において開始するように、セグメントのうちの1つまたは複数がパディングされる。上記命令は、ツリーブロックの符号化表現を復号するようにもビデオ復号デバイスを構成する。

30

【0012】

別の態様では、本開示では、ビデオデータを符号化するビデオ符号化デバイスについて説明する。本ビデオ符号化デバイスは、ピクチャーを複数のピクチャーパーティションに分割するための手段を含む。ピクチャーは、複数のツリーブロックを有する。ピクチャーパーティションは、ピクチャーのツリーブロックのオーバーラップしないサブセットに関連付けられる。本ビデオ符号化デバイスは、ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスNALユニットを生成するための手段をも含む。ツリーブロックの符号化表現は、コード化スライスNALユニット内で上記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化される。セグメントのそれぞれがバイト境界上で開始するように、セグメントのうちの1つまたは複数がパディングされる。

40

【0013】

50

別の態様では、本開示では、ビデオデータを復号するビデオ復号デバイスについて説明する。本ビデオ復号デバイスは、ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスNALユニットを記憶するための手段を含む。ピクチャーは、複数のピクチャーパーティションに分割される。ツリーブロックの符号化表現は、上記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化される。セグメントのそれぞれがバイト境界において開始するように、セグメントのうちの1つまたは複数がパディングされる。本ビデオ復号デバイスは、ツリーブロックの符号化表現を復号するための手段を含む。

【0014】

1つまたは複数の例の詳細を添付の図面および以下の説明に記載する。他の特徴、目的、および利点は、その説明および図面、ならびに特許請求の範囲から明らかになる。

【図面の簡単な説明】

【0015】

【図1】図1は、本開示の技法を利用し得る例示的なビデオコーディングシステムを示すブロック図である。

【図2】図2は、本開示の技法を実装するように構成された例示的なビデオエンコーダを示すブロック図である。

【図3】図3は、本開示の技法を実装するように構成された例示的なビデオデコーダを示すブロック図である。

【図4】図4は、ピクチャーのスライスのためのスライスデータを生成するための例示的な動作を示すフローチャートである。

【図5】図5は、コード化スライスNALユニットを復号するための例示的な動作を示すフローチャートである。

【図6】図6は、ウェイブフロント並列処理を示す概念図である。

【図7】図7は、ピクチャーが複数のタイルに区分されるときに例示的なコーディング順序を示す概念図である。

【図8】図8は、例示的なコード化スライスNALユニットを示す概念図である。

【発明を実施するための形態】

【0016】

ピクチャーは、複数のツリーブロックを含む。ツリーブロックは、ピクチャー内の2次元ビデオブロックに関連付けられる。ビデオエンコーダは、ピクチャーを複数のピクチャーパーティションに分割する。たとえば、ビデオエンコーダは、ピクチャーをタイルまたはウェイブフロント並列処理(WPP)ウェイブに分割し得る。言い換えれば、本開示では、タイルまたはWPPウェイブを総称的に指すために「ピクチャーパーティション」(picture partition)という用語を使用し得る。ピクチャーパーティションは、ピクチャーのツリーブロックのオーバーラップしないサブセットに関連付けられる。たとえば、ピクチャーの各ツリーブロックがピクチャーパーティションのうちの厳密に1つに関連付けられ得る。

【0017】

ビデオエンコーダは、コード化スライスネットワークアブストラクションレイヤ(NAL)ユニットを生成し得る。コード化スライスNALユニットは、ピクチャーのスライスに関連付けられた各ツリーブロックの符号化表現を含み得る。本開示では、ツリーブロックの符号化表現をコード化ツリーブロックと呼ぶことがある。コード化ツリーブロックは、ツリーブロックに関連付けられたビデオブロックを表すビットのシーケンスを含み得る。コード化ツリーブロック中のビットのシーケンスは、シンタックス要素のシーケンスを表し得る。

【0018】

ビデオエンコーダは、コード化スライスNALユニット内のコード化ツリーブロックをセグメントにグループ化し得る。セグメントは、上記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられる。セグメントのそれぞれは、たとえば

一連の1つまたは複数のコード化ツリーブロックおよび関連データを表すビットなどのような、連続する一連のビットであり得る。したがって、コード化スライスNALユニットは、第1のピクチャーパーティションに関連付けられた各コード化ツリーブロック、その後続く第2のピクチャーパーティションに関連付けられた各コード化ツリーブロック、その後続く第3のピクチャーパーティションに関連付けられた各コード化ツリーブロックなどを含み得る。

【0019】

本開示の技法によれば、ビデオエンコーダは、セグメントのそれぞれがバイト境界上で開始するように、セグメントのうちの1つまたは複数をパディングし得る。ビデオエンコーダがセグメントをパディングするとき、ビデオエンコーダはセグメントにパディングビットを付加し得る。パディングビットは、いかなる意味論的意味 (semantic meaning) をも有しないことがあるが、次のセグメントがバイト境界において開始することを保証するのに役立ち得る。このようにして、ビデオエンコーダは、並列処理目的でタイルまたはWPPウェーブが1つのコード化スライスNALユニット中に含まれるとき、タイルまたはWPPウェーブのバイト・アラインメント (byte alignment) を提供し得る。

【0020】

ビデオデコーダは、コード化スライスNALユニットをバイトアドレス指定のメモリに記憶し得る。ビデオデコーダは、次いで、セグメントのうちの2つ以上を、並列に動作する異なる復号スレッド (decoding threads) に割り当て得る。各復号スレッドは、復号スレッドに割り当てられたセグメントのコード化ツリーブロックを復号する。セグメントのそれぞれがバイト境界において開始するので、ビデオデコーダは、復号スレッドにセグメントを割り当てるとき、復号スレッドにセグメントのメモリアドレスを与え得る。このようにして、セグメントのそれぞれがバイト境界において開始することを保証することにより、ビデオデコーダは、セグメントが非バイト境界位置において開始し得るときよりも単純な方式でセグメントを並列に復号することが可能になり得る。

【0021】

これは、セグメントがバイト境界において開始することを保証しない従来のビデオエンコーダおよび従来のビデオデコーダとは対照をなし得る。セグメントがバイト境界において開始しないことがあるので、バイト単位のメモリアドレス指定を使用する従来のビデオデコーダは、セグメント中のコード化ツリーブロックを並列に復号することができないことがある。従来のビデオデコーダは、セグメント中のコード化ツリーブロックを並列に復号することを可能にするためにビット単位のメモリアドレス指定またはバイト単位+ビット単位のアドレス指定を使用し得るが、実装および計算の複雑さの増加を伴う。

【0022】

添付の図面は例を示している。添付の図面中の参照番号によって示される要素は、以下の説明における同様の参照番号によって示される要素に対応する。本開示では、序数語 (たとえば、「第1の」、「第2の」、「第3の」など) で始まる名前を有する要素は、必ずしもそれらの要素が特定の順序を有することを暗示するとは限らない。むしろ、そのような序数語は、同じまたは同様のタイプの異なる要素を指すために使用されるにすぎない。さらに、以下の説明では、「現在ピクチャー (current picture)」は、「現在、符号化または復号されているピクチャー」を指すことがある。

【0023】

図1は、本開示の技法を利用し得る例示的なビデオコーディングシステム10を示すブロック図である。本明細書で使用し説明する「ビデオコーダ (video coder)」という用語は、ビデオエンコーダとビデオデコーダの両方を総称的に指す。本開示では、「ビデオコーディング (video coding)」または「コーディング (coding)」という用語は、ビデオ符号化 (video encoding) およびビデオ復号 (video decoding) を総称的に指すことがある。

【0024】

図1に示すように、ビデオコーディングシステム10は、ソースデバイス12と宛先デ

10

20

30

40

50

バイス 14 とを含む。ソースデバイス 12 は、符号化ビデオデータを生成する。したがって、ソースデバイス 12 は、ビデオ符号化デバイスと呼ばれることがある。宛先デバイス 14 は、ソースデバイス 12 によって生成された符号化ビデオデータを復号し得る。したがって、宛先デバイス 14 は、ビデオ復号デバイスと呼ばれることがある。ソースデバイス 12 および宛先デバイス 14 はビデオコーディングデバイスの例であり得る。

【0025】

ソースデバイス 12 および宛先デバイス 14 は、デスクトップコンピュータ、モバイルコンピューティングデバイス、ノートブック（たとえば、ラップトップ）コンピュータ、タブレットコンピュータ、セットトップボックス、いわゆる「スマート」フォンなどのような電話ハンドセット、テレビジョン、カメラ、ディスプレイデバイス、デジタルメディアプレーヤ、ビデオゲームコンソール、車内コンピュータなどを含む、広範囲にわたるデバイスを含み得る。いくつかの例では、ソースデバイス 12 および宛先デバイス 14 は、ワイヤレス通信が可能であり得る。

【0026】

宛先デバイス 14 は、チャンネル 16 を介してソースデバイス 12 から符号化ビデオデータを受信し得る。チャンネル 16 は、ソースデバイス 12 から宛先デバイス 14 に符号化ビデオデータを移動することが可能なタイプの媒体またはデバイスを含み得る。一例では、チャンネル 16 は、ソースデバイス 12 が符号化ビデオデータをリアルタイムで宛先デバイス 14 に直接送信することを可能にする通信媒体を含み得る。この例では、ソースデバイス 12 は、ワイヤレス通信プロトコルなどのような通信規格に従って符号化ビデオデータを変調し得、変調されたビデオデータを宛先デバイス 14 に送信し得る。通信媒体は、無線周波数（RF）スペクトルまたは 1 つまたは複数の物理伝送線路などのような、ワイヤレス通信媒体またはワイヤード通信媒体を含み得る。通信媒体は、たとえばローカルエリアネットワーク、ワイドエリアネットワーク、またはインターネットなどのグローバルネットワークなどのような、パケットベースネットワークの一部を形成し得る。通信媒体は、ソースデバイス 12 から宛先デバイス 14 への通信を容易にするルータ、スイッチ、基地局、または他の機器を含み得る。

【0027】

別の例では、チャンネル 16 は、ソースデバイス 12 によって生成された符号化ビデオデータを記憶する記憶媒体に対応し得る。この例では、宛先デバイス 14 は、ディスクアクセスまたはカードアクセスを介して記憶媒体にアクセスし得る。記憶媒体は、たとえば Blu-ray（登録商標）ディスク、DVD、CD-ROM、フラッシュメモリ、または符号化ビデオデータを記憶するための他の好適なデジタル記憶媒体などのような、様々なローカルにアクセスされたデータ記憶媒体を含み得る。さらなる例では、チャンネル 16 は、ソースデバイス 12 によって生成された符号化ビデオを記憶するファイルサーバまたは別の中間ストレージデバイスを含み得る。この例では、宛先デバイス 14 は、ストリーミングまたはダウンロードを介して、ファイルサーバまたは別の中間ストレージデバイスに記憶された符号化ビデオデータにアクセスし得る。ファイルサーバは、符号化ビデオデータを記憶することと、符号化ビデオデータを宛先デバイス 14 に送信することとが可能なタイプのサーバであり得る。例示的なファイルサーバは、（たとえば、ウェブサイトのための）ウェブサーバ、ファイル転送プロトコル（FTP）サーバ、ネットワーク接続ストレージ（NAS）デバイス、およびローカルディスクドライブを含む。宛先デバイス 14 は、インターネット接続を含む、標準のデータ接続を介して符号化ビデオデータにアクセスし得る。例示的なタイプのデータ接続は、ファイルサーバに記憶された符号化ビデオデータにアクセスするのに好適である、ワイヤレスチャンネル（たとえば、Wi-Fi（登録商標）接続）、ワイヤード接続（たとえば、DSL、ケーブルモデムなど）、または両方の組み合わせを含み得る。ファイルサーバからの符号化ビデオデータの送信は、ストリーミング送信、ダウンロード送信、または両方の組み合わせであり得る。

【0028】

本開示の技法は、ワイヤレスのアプリケーションまたはセッティングに限定されない。

本技法は、たとえば無線の（over-the-air）テレビジョン放送、ケーブルテレビジョン送信、衛星テレビジョン送信、たとえばインターネットを介したストリーミングビデオ送信、データ記憶媒体に記憶するためのデジタルビデオの符号化、データ記憶媒体に記憶されたデジタルビデオの復号、または他のアプリケーションなどのような、様々なマルチメディア・アプリケーションのいずれかをサポートするビデオコーディングに適用され得る。いくつかの例では、ビデオコーディングシステム 10 は、たとえばビデオストリーミング、ビデオ再生、ビデオブロードキャスト、および / またはビデオテレフォニーなどのようなアプリケーションをサポートするために、一方向または双方向のビデオ送信をサポートするように構成され得る。

【0029】

10

図 1 の例では、ソースデバイス 12 は、ビデオソース 18 と、ビデオエンコーダ 20 と、出力インターフェース 22 とを含む。場合によっては、出力インターフェース 22 は、変調器 / 復調器（モデム）および / または送信機を含み得る。ソースデバイス 12 において、ビデオソース 18 は、たとえばビデオカメラなどのビデオキャプチャデバイス、以前にキャプチャされたビデオデータを含んでいるビデオアーカイブ、ビデオコンテンツプロバイダからビデオデータを受信するためのビデオフィードインターフェース、および / またはビデオデータを生成するためのコンピュータグラフィックスシステムなどのようなソース、あるいはそのようなソースの組み合わせを含み得る。

【0030】

ビデオエンコーダ 20 は、キャプチャされたビデオデータ、以前にキャプチャされたビデオデータ、またはコンピュータ生成されたビデオデータを符号化し得る。符号化ビデオデータは、ソースデバイス 12 の出力インターフェース 22 を介して宛先デバイス 14 に直接送信され得る。符号化ビデオデータはまた、復号および / または再生のための宛先デバイス 14 による後のアクセスのために記憶媒体またはファイルサーバ上に記憶され得る。

20

【0031】

図 1 の例では、宛先デバイス 14 は、入力インターフェース 28 と、ビデオデコーダ 30 と、ディスプレイデバイス 32 とを含む。場合によっては、入力インターフェース 28 は、受信機および / またはモデムを含み得る。宛先デバイス 14 の入力インターフェース 28 は、チャンネル 16 を介して符号化ビデオデータを受信する。符号化ビデオデータは、ビデオデータを表す、ビデオエンコーダ 20 によって生成された様々なシンタックス要素を含み得る。そのようなシンタックス要素は、通信媒体上で送信されるか、記憶媒体上に記憶されるか、またはファイルサーバ上に記憶される、符号化ビデオデータとともに含まれ得る。

30

【0032】

ディスプレイデバイス 32 は、宛先デバイス 14 と一体化され得るかまたはその外部にあり得る。いくつかの例では、宛先デバイス 14 は、一体型ディスプレイデバイスを含み得、また、外部ディスプレイデバイスとインターフェースするように構成され得る。他の例では、宛先デバイス 14 は、ディスプレイデバイスであり得る。一般に、ディスプレイデバイス 32 は、復号ビデオデータをユーザに表示する。ディスプレイデバイス 32 は、たとえば液晶ディスプレイ（LCD）、プラズマディスプレイ、有機発光ダイオード（OLED）ディスプレイ、または別のタイプのディスプレイデバイスなどのような、様々なディスプレイデバイスのいずれかを含み得る。

40

【0033】

ビデオエンコーダ 20 およびビデオデコーダ 30 は、現在開発中の高効率ビデオコーディング（HEVC）規格などのような、ビデオ圧縮規格に従って動作し得、HEVC テストモデル（HM）に準拠し得る。「HEVC Working Draft 6」または「WD6」と呼ばれる、今度の HEVC 規格の最近のドラフトは、2012 年 5 月 1 日現在、http://phenix.int-evry.fr/jct/doc_end_user/documents/8_San%20Jose/wg11/JCTVC-H1003-v22.zip からダウンロード可能である、ドキュメント JCTVC - H1003、B

50

rossら、「High efficiency video coding (HEVC) text specification draft 6」、Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、8th Meeting: San Jose, California, USA, February, 2012に記載されており、その内容全体が参照により本明細書に組み込まれる。代わりに、ビデオエンコーダ20およびビデオデコーダ30は、タイルまたはウェイブフロント並列処理に類似するピクチャー区分技法が含まれるならば、たとえばITU-T H.264規格(代わりにMPEG-4, Part 10, Advanced Video Coding (AVC)と呼ばれる)などのような、他のプロプライエタリ規格または業界規格、あるいはそのような規格の拡張に従って、動作し得る。しかしながら、本開示の技法は、いかなる特定のコーディング規格またはコーディング技法にも限定されない。ビデオ圧縮の規格および技法の他の例としては、タイルまたはウェイブフロント並列処理のようなピクチャー区分技法が含まれるとき、MPEG-2、ITU-T H.263、ならびにVP8および関係するフォーマットなどのようなプロプライエタリまたはオープンソースの圧縮フォーマットがある。

10

【0034】

図1の例には示されていないが、ビデオエンコーダ20およびビデオデコーダ30は、それぞれオーディオエンコーダおよびデコーダと統合され得、適切なMUX-DEMUXユニット、または他のハードウェアおよびソフトウェアを含んで、共通のデータストリームまたは別個のデータストリーム中のオーディオとビデオの両方の符号化を処理し得る。適用可能な場合、いくつかの例では、MUX-DEMUXユニットはITU H.223マルチプレクサプロトコル、またはユーザデータグラムプロトコル(UDP)などのような他のプロトコルに準拠し得る。

20

【0035】

この場合も、図1は一例にすぎず、本開示の技法は、符号化デバイスと復号デバイスとの間のデータ通信を必ずしも含むとは限らないビデオコーディング・セッティング(たとえば、ビデオ符号化またはビデオ復号)に適用され得る。他の例では、データがローカルメモリから取り出されること、ネットワークを介してストリーミングされることなどが行われ得る。符号化デバイスがデータを符号化し、メモリに記憶し得、および/または復号デバイスがメモリからデータを取り出し、復号し得る。多くの例では、符号化および復号は、互いに通信しないが、メモリにデータを符号化し、および/またはメモリからデータを取り出し、復号するだけである、デバイスによって実行される。

30

【0036】

ビデオエンコーダ20およびビデオデコーダ30は、それぞれ、1つまたは複数のマイクロプロセッサ、デジタル信号プロセッサ(DSP)、特定用途向け集積回路(ASIC)、フィールドプログラマブルゲートアレイ(FPGA)、ディスクリート論理、ハードウェアなどのような、様々な好適な回路のいずれか、あるいはそれらの任意の組み合わせとして実装され得る。本技法が部分的にソフトウェアで実装されるとき、デバイスは、好適な非一時的コンピュータ可読記憶媒体にソフトウェアの命令を記憶し得、1つまたは複数のプロセッサを使用してその命令をハードウェアで実行して、本開示の技法を実行し得る。ビデオエンコーダ20およびビデオデコーダ30のそれぞれは、1つまたは複数のエンコーダまたはデコーダ中に含まれ得、そのいずれも、それぞれのデバイスにおいて複合エンコーダ/デコーダ(コーデック)の一部として統合され得る。

40

【0037】

上記で手短かに述べたように、ビデオエンコーダ20は、ビデオデータを符号化する。ビデオデータは、1つまたは複数のピクチャーを含み得る。ピクチャーのそれぞれは、ビデオの一部を形成する静止画像である。いくつかの事例では、ピクチャーは、ビデオ「フレーム」(frame)またはビデオ「フィールド」(field)と呼ばれることがある。ビデオエンコーダ20がビデオデータを符号化するとき、ビデオエンコーダ20は、ビットストリームを生成し得る。ビットストリームは、ビデオデータのコード化表現(coded represen

50

tation)を形成するビットのシーケンスを含み得る。ビットストリームは、コード化ピクチャー(coded pictures)と関連データとを含み得る。コード化ピクチャーは、ピクチャーのコード化表現である。

【0038】

ビットストリームを生成するために、ビデオエンコーダ20は、ビデオデータ中の各ピクチャーに対して符号化演算(encoding operations)を実行し得る。ビデオエンコーダ20がピクチャーに対して符号化演算を実行するとき、ビデオエンコーダ20は、一連のコード化ピクチャーと関連データとを生成し得る。関連データは、シーケンスパラメータセット、ピクチャーパラメータセット、適応パラメータセット、および他のシンタックス構造を含み得る。シーケンスパラメータセット(SPS:sequence parameter set)は、ピクチャーの0個以上のシーケンスに適用可能なパラメータを含んでいることがある。ピクチャーのシーケンスは、H.264/AVCおよびHEVCの場合のように、コード化ビデオシーケンスと呼ばれることもある。ピクチャーパラメータセット(PPS:picture parameter set)は、0個以上のピクチャーに適用可能なパラメータを含んでいることがある。適応パラメータセット(APS:adaptation parameter set)は、0個以上のピクチャーに適用可能なパラメータを含んでいることがある。APS内のパラメータは、PPS内のパラメータよりも変化する可能性が高いパラメータであり得る。

【0039】

コード化ピクチャーを生成するために、ビデオエンコーダ20は、ピクチャーを等しいサイズのビデオブロックに区分し得る。ビデオブロックは、サンプルの2次元アレイであり得る。ビデオブロックのそれぞれは、ツリーブロックに関連付けられる。いくつかの事例では、ツリーブロックは、最大コーディングユニット(LCU:largest coding unit)またはコーディングツリーブロックと呼ばれることがある。HEVCのツリーブロックは、H.264/AVCなどのような、以前の規格のマクロブロックに広い意味で類似し得る。しかしながら、ツリーブロックは、必ずしも特定のサイズに限定されるとは限らず、1つまたは複数のコーディングユニット(CU:coding unit)を含み得る。ビデオエンコーダ20は、4分木区分を使用して、ツリーブロックのビデオブロックを、CUに関連付けられたビデオブロックに区分し得、したがって「ツリーブロック」(treelocks)という名前がある。

【0040】

いくつかの例では、ビデオエンコーダ20は、ピクチャーを複数のスライスに区分し得る。スライスのそれぞれは、整数個の連続するコード化ツリーブロックを含み得る。いくつかの事例では、スライスのそれぞれは、整数個の連続するコード化CUを含み得る。ピクチャーに対して符号化演算を実行することの一部として、ビデオエンコーダ20は、ピクチャーの各スライスに対して符号化演算を実行し得る。ビデオエンコーダ20がスライスに対して符号化演算を実行するとき、ビデオエンコーダ20は、スライスに関連付けられた符号化データを生成し得る。スライスに関連付けられた符号化データは、「コード化スライス」(coded slice)と呼ばれることがある。

【0041】

コード化スライスを生成するために、ビデオエンコーダ20は、スライス中の各ツリーブロックに対して符号化演算を実行し得る。ビデオエンコーダ20がツリーブロックに対して符号化演算を実行するとき、ビデオエンコーダ20は、コード化ツリーブロックを生成し得る。コード化ツリーブロックは、ツリーブロックの符号化バージョンを表すデータを含み得る。

【0042】

ビデオエンコーダ20がコード化スライスを生成するとき、ビデオエンコーダ20は、ラスタスキャン順序に従ってスライス中のツリーブロックに対して符号化演算を実行し得る(すなわち、そのツリーブロックを符号化し得る)。言い換えれば、ビデオエンコーダ20は、スライス中のツリーブロックの一番上の行にわたって左から右に進み、次いでツリーブロックの次の下の行にわたって左から右に進み、以下同様に進む、順序で、ビデオ

10

20

30

40

50

エンコーダ 20 がスライス中のツリーブロックのそれぞれを符号化するまで、スライスのツリーブロックを符号化し得る。

【0043】

ラストスキャン順序に従ってツリーブロックを符号化した結果として、所与のツリーブロックの上および左のツリーブロックは符号化されていることがあるが、所与のツリーブロックの下および右のツリーブロックはまだ符号化されていない。したがって、ビデオエンコーダ 20 は、所与のツリーブロックを符号化するとき、所与のツリーブロックの上および左のツリーブロックを符号化することによって生成された情報にアクセスすることが可能であり得る。しかしながら、ビデオエンコーダ 20 は、所与のツリーブロックを符号化するとき、所与のツリーブロックの下および右のツリーブロックを符号化することによって生成された情報にアクセスすることができないことがある。

10

【0044】

コード化ツリーブロックを生成するために、ビデオエンコーダ 20 は、ツリーブロックのビデオブロックに対して 4 分木区分を再帰的に実行して、ビデオブロックを徐々により小さいビデオブロックに分割し得る。より小さいビデオブロックのそれぞれは、異なる CU に関連付けられ得る。たとえば、ビデオエンコーダ 20 は、ツリーブロックのビデオブロックを 4 つの等しいサイズのサブブロック (sub-blocks) に区分し、サブブロックのうちの 1 つまたは複数を、4 つの等しいサイズのサブサブブロック (sub-sub-blocks) に区分し得、以下同様である。区分された CU (partitioned CU) は、そのビデオブロックが、他の CU に関連付けられたビデオブロックに区分された、CU であり得る。区分されていない CU (non-partitioned CU) は、そのビデオブロックが、他の CU に関連付けられたビデオブロックに区分されていない、CU であり得る。

20

【0045】

ビットストリーム中の 1 つまたは複数のシンタックス要素は、ビデオエンコーダ 20 がツリーブロックのビデオブロックを区分し得る最大の回数を示し得る。CU のビデオブロックは、形状が正方形であり得る。CU のビデオブロックのサイズ (すなわち、CU のサイズ) は、8 × 8 ピクセルから、ツリーブロックのビデオブロックのサイズ (すなわち、ツリーブロックのサイズ) までにわたり、最大で 64 × 64 ピクセル以上であり得る。

【0046】

ビデオエンコーダ 20 は、z スキャン順序に従って、ツリーブロックの各 CU に対して符号化演算を実行し得る (すなわち、各 CU を符号化し得る)。言い換えれば、ビデオエンコーダ 20 は、左上の CU と、右上の CU と、左下の CU と、次いで右下の CU とを、その順序で符号化し得る。ビデオエンコーダ 20 が区分された CU に対して符号化演算を実行するとき、ビデオエンコーダ 20 は、z スキャン順序に従って、区分された CU のビデオブロックのサブブロックに関連付けられた CU を符号化し得る。言い換えれば、ビデオエンコーダ 20 は、左上のサブブロックに関連付けられた CU と、右上のサブブロックに関連付けられた CU と、左下のサブブロックに関連付けられた CU と、次いで右下のサブブロックに関連付けられた CU とを、その順序で符号化し得る。

30

【0047】

z スキャン順序に従ってツリーブロックの CU を符号化した結果として、所与の CU の上、左上、右上、左、および左下の CU は符号化されていることがある。所与の CU の下および右の CU はまだ符号化されていない。したがって、ビデオエンコーダ 20 は、所与の CU を符号化するとき、所与の CU に隣接するいくつかの CU を符号化することによって生成された情報にアクセスすることが可能であり得る。しかしながら、ビデオエンコーダ 20 は、所与の CU を符号化するとき、所与の CU に隣接する他の CU を符号化することによって生成された情報にアクセスすることができないことがある。

40

【0048】

ビデオエンコーダ 20 が区分されていない CU を符号化するとき、ビデオエンコーダ 20 は、CU のために 1 つまたは複数の予測ユニット (PU: prediction unit) を生成し得る。CU の PU のそれぞれは、CU のビデオブロック内の異なるビデオブロックに関連

50

付けられ得る。ビデオエンコーダ 20 は、CU の各 PU について予測されたビデオブロックを生成し得る。PU の予測されたビデオブロックは、サンプルのブロックであり得る。ビデオエンコーダ 20 は、イントラ予測またはインター予測を使用して、PU の予測されたビデオブロックを生成し得る。

【0049】

ビデオエンコーダ 20 がイントラ予測を使用して PU の予測されたビデオブロックを生成するとき、ビデオエンコーダ 20 は、PU に関連付けられたピクチャーの復号化サンプル (decoded samples) に基づいて、PU の予測されたビデオブロックを生成し得る。ビデオエンコーダ 20 がイントラ予測を使用して CU の PU の予測されたビデオブロックを生成する場合、CU はイントラ予測された CU である。ビデオエンコーダ 20 がインター予測を使用して PU の予測されたビデオブロックを生成するとき、ビデオエンコーダ 20 は、PU に関連付けられたピクチャー以外の 1 つまたは複数のピクチャーの復号化サンプルに基づいて、PU の予測されたビデオブロックを生成し得る。ビデオエンコーダ 20 がインター予測を使用して CU の PU の予測されたビデオブロックを生成する場合、CU はインター予測された CU である。

【0050】

さらに、ビデオエンコーダ 20 がインター予測を使用して PU の予測されたビデオブロックを生成するとき、ビデオエンコーダ 20 は、PU の動き情報を生成し得る。PU の動き情報は、PU の 1 つまたは複数の参照ブロックを示し得る。PU の各参照ブロックは、参照ピクチャー内のビデオブロックであり得る。参照ピクチャーは、PU に関連付けられたピクチャー以外のピクチャーであり得る。いくつかの事例では、PU の参照ブロックは、PU の「参照サンプル」 (reference sample) と呼ばれることもある。ビデオエンコーダ 20 は、PU の参照ブロックに基づいて PU の予測されたビデオブロックを生成し得る。

【0051】

ビデオエンコーダ 20 が CU の 1 つまたは複数の PU の予測されたビデオブロックを生成した後、ビデオエンコーダ 20 は、CU の PU の予測されたビデオブロックに基づいて、CU の残差データ (residual data) を生成し得る。CU の残差データは、CU の PU の予測されたビデオブロック中のサンプルと、CU の元のビデオブロック中のサンプルとの間の差 (differences) を示し得る。

【0052】

さらに、区分されていない CU に対して符号化演算を実行することの一部として、ビデオエンコーダ 20 は、CU の残差データに対して再帰的な 4 分木区分を実行して、CU の残差データを、CU の変換ユニット (TU : transform unit) に関連付けられた残差データの 1 つまたは複数のブロック (すなわち、残差ビデオブロック) に区分し得る。CU の各 TU は、異なる残差ビデオブロックに関連付けられ得る。

【0053】

ビデオエンコーダ 20 は、TU に関連付けられた残差ビデオブロックに 1 つまたは複数の変換を適用して、TU に関連付けられた変換係数ブロック (すなわち、変換係数のブロック) を生成し得る。概念的に、変換係数ブロックは、変換係数の 2 次元 (2D) 行列であり得る。

【0054】

変換係数ブロックを生成した後、ビデオエンコーダ 20 は、変換係数ブロックに対して量子化プロセスを実行し得る。量子化は、一般に、変換係数を表すために使用されるデータの量をできるだけ低減するために変換係数が量子化され、さらなる圧縮を実現する処理を指す。量子化プロセスは、変換係数の一部または全部に関連付けられたビット深度 (bit depth) を低減し得る。たとえば、量子化中に n ビットの変換係数が m ビットの変換係数に切り捨てられることがあり、ここで、n は m よりも大きい。

【0055】

ビデオエンコーダ 20 は、各 CU を量子化パラメータ (QP : quantization parameter

10

20

30

40

50

）値に関連付け得る。C Uに関連付けられたQ P値は、ビデオエンコーダ20が、C Uに関連付けられた変換係数ブロックをどのように量子化するかを決定し得る。ビデオエンコーダ20は、C Uに関連付けられたQ P値を調整することによって、C Uに関連付けられた変換係数ブロックに適用される量子化のディグリー（degree of quantization）を調整し得る。

【0056】

ビデオエンコーダ20が変換係数ブロックを量子化した後、ビデオエンコーダ20は、量子化された変換係数ブロック内で変換係数を表すシンタックス要素のセットを生成し得る。ビデオエンコーダ20は、これらのシンタックス要素のうちのいくつかに、コンテキスト適応型バイナリ算術コーディング（C A B A C）演算（Context Adaptive Binary Arithmetic Coding (CABAC) operations）などのような、エントロピー符号化演算を適用し得る。

10

【0057】

ビデオエンコーダ20によって生成されるビットストリームは、一連のネットワークアブストラクションレイヤ（N A L）ユニットを含み得る。N A Lユニットのそれぞれは、N A Lユニット中のデータのタイプの指示と、データを含んでいるバイトとを含んでいるシンタックス構造であり得る。たとえば、N A Lユニットは、シーケンスパラメータセット、ピクチャーパラメータセット、コード化スライス、1つまたは複数の補足エンハンスメント情報（S E I : supplemental enhancement information）メッセージ、アクセスユニットデリミター（access unit delimiter）、フィラーデータ（filler data）、または別のタイプのデータを表すデータを含んでいることがある。N A Lユニット中のデータは、様々なシンタックス構造を含み得る。

20

【0058】

ビデオデコーダ30は、ビデオエンコーダ20によって生成されたビットストリームを受信し得る。ビットストリームは、ビデオエンコーダ20によって符号化されたビデオデータのコード化表現を含み得る。ビデオデコーダ30がビットストリームを受信するとき、ビデオデコーダ30は、ビットストリームに対してパース演算（parsing operation）を実行し得る。ビデオデコーダ30がパース演算を実行するとき、ビデオデコーダ30は、ビットストリームからシンタックス要素を抽出し得る。ビデオデコーダ30は、ビットストリームから抽出されたシンタックス要素に基づいて、ビデオデータのピクチャーを再構成し得る。シンタックス要素に基づいてビデオデータを再構成するためのプロセスは、一般に、シンタックス要素を生成するためにビデオエンコーダ20によって実行されるプロセスの逆であり得る。

30

【0059】

ビデオデコーダ30がC Uに関連付けられたシンタックス要素を抽出した後、ビデオデコーダ30は、シンタックス要素に基づいて、C UのP Uの予測されたビデオブロックを生成し得る。加えて、ビデオデコーダ30は、C UのT Uに関連付けられた変換係数ブロックを逆量子化し得る。ビデオデコーダ30は、変換係数ブロックに対して逆変換を実行して、C UのT Uに関連付けられた残差ビデオブロックを再構成し得る。予測されたビデオブロックを生成し、残差ビデオブロックを再構成した後、ビデオデコーダ30は、予測されたビデオブロックおよび残差ビデオブロックに基づいて、C Uのビデオブロックを再構成し得る。このようにして、ビデオデコーダ30は、ビットストリーム中のシンタックス要素に基づいて、C Uのビデオブロックを再構成し得る。

40

【0060】

ビデオエンコーダ20は、現在ピクチャーを複数のピクチャーパーティションに分割し得る。ピクチャーパーティションは、現在ピクチャーのツリーブロックのオーバーラップしないサブセットに関連付けられ得る。ビデオエンコーダ20は、様々な方法で現在ピクチャーを複数のピクチャーパーティションに分割し得る。以下で説明するように、ビデオエンコーダ20は、現在ピクチャーを複数のタイルにまたは複数のウェイブフロント並列処理（W P P）ウェイブに分割し得る。本開示では、タイルとW P Pウェイブの両方を総

50

称的に指すために「ピクチャーパーティション」(picture partition)という用語を使用し得る。現在ピクチャーをピクチャーパーティションに分割するプロセスは、現在ピクチャーをピクチャーパーティションに「区分」すること(partitioning)と呼ばれることがある。

【0061】

上述のように、ビデオエンコーダ20は、現在ピクチャーを1つまたは複数のタイルに分割し得る。タイルのそれぞれは、現在ピクチャー中の整数個のツリーブロックを含み得る。ビデオエンコーダ20は、2つ以上の垂直タイル境界と2つ以上の水平タイル境界とを画定することによって現在ピクチャーをタイルに分割し得る。現在ピクチャーの各垂直辺は、垂直タイル境界であると見なされ得る。現在ピクチャーの各水平辺は、水平タイル境界であると見なされ得る。たとえば、ビデオエンコーダ20が現在ピクチャーのための4つの垂直タイル境界と3つの水平タイル境界とを画定する場合、現在ピクチャーは、6つのタイルに分割される。

10

【0062】

ビデオエンコーダ20またはビデオデコーダ30などのビデオコードは、ラスタスキャン順序に従って、現在ピクチャーのタイルをコーディングし得る。さらに、ビデオコードがタイルをコーディングするとき、ビデオコードは、ラスタスキャン順序に従ってタイル内の各ツリーブロックをコーディングし得る。このようにして、ビデオコードは、現在ピクチャーの別のタイルのいずれかのツリーブロックをコーディングする前に、現在ピクチャーの所与のタイルの各ツリーブロックをコーディングし得る。したがって、ビデオコードが現在ピクチャーを複数のタイルに区分するとき、ビデオコードが現在ピクチャーのツリーブロックをコーディングする順序は、ビデオコードが現在ピクチャーを複数のタイルに区分しないと異なり得る。

20

【0063】

さらに、いくつかの事例では、ビデオコードは、所与のCUと空間的に隣接するCUとが同じタイルに属する限り、現在ピクチャー中の所与のCUに対してイントラ予測を実行するために、空間的に隣接するCUに関連付けられた情報を使用し得る。空間的に隣接するCUは、現在ピクチャーの現在スライスに属するCUである。いくつかの事例では、ビデオコードは、所与のCUと空間的に隣接するCUとが同じタイル内にある限り、所与のCUのシンタックス要素をCABAC符号化するためのコンテキストを選択するために、空間的に隣接するCUに関連付けられた情報を使用し得る。これらの制限のために、ビデオコードは、複数のタイルのツリーブロックを並列にコーディングすることが可能であり得る。

30

【0064】

他の例では、ビデオコードは、ウェイブフロント並列処理(WPP)を使用して現在ピクチャーをコーディングし得る。ビデオコードがWPPを使用して現在ピクチャーをコーディングするとき、ビデオコードは、現在ピクチャーのツリーブロックを複数の「WPPウェイブ」(WPP waves)に分割し得る。WPPウェイブのそれぞれは、現在ピクチャー中のツリーブロックの異なる行に対応し得る。ビデオコードがWPPを使用して現在ピクチャーをコーディングするとき、ビデオコードは、ツリーブロックの最上行をコーディングし始め得る。ビデオコードが最上行の2つ以上のツリーブロックをコーディングしたとき、ビデオコードは、ツリーブロックの最上行をコーディングすることと並列にツリーブロックの最上行から2番目の行をコーディングし始め得る。ビデオコードが最上行から2番目の行の2つ以上のツリーブロックをコーディングしたとき、ビデオコードは、ツリーブロックの最上行から3番目の行よりも上の行をコーディングすることと並列にツリーブロックの最上行から3番目の行をコーディングし始め得る。このパターンは、現在ピクチャー中のツリーブロックの行を下って続き得る。

40

【0065】

ビデオコードがWPPを使用して現在ピクチャーをコーディングしているとき、ビデオコードは、空間的に隣接するCUが現在ツリーブロックの左、左上、上、または右上にあ

50

る限り、現在ツリーブロック中の所与のCUに対してイントラ予測を実行するために現在ツリーブロック外の空間的に隣接するCUに関連付けられた情報を使用し得る。現在ツリーブロックが一番上の行以外の行中の最左ツリーブロックである場合、ビデオコーダは、現在ツリーブロックのシンタックス要素をCABAC符号化するためのコンテキストを選択するためにすぐ上の行の第2のツリーブロックに関連付けられた情報を使用し得る。そうではなく、現在ツリーブロックが上記行中の最左ツリーブロックでない場合、ビデオコーダは、現在ツリーブロックのシンタックス要素をCABAC符号化するためのコンテキストを選択するために現在ツリーブロックの左のツリーブロックに関連付けられた情報を使用し得る。このようにして、ビデオコーダは、すぐ上の行の2つ以上のツリーブロックを符号化した後に、すぐ上の行のCABAC状態に基づいて、ある行のCABAC状態を初期化し得る。

10

【0066】

いくつかの例では、ビデオコーダがWPPを使用して現在ピクチャーをコーディングしているとき、現在ピクチャーの単独のタイルの境界は、現在ピクチャーの水平境界および垂直境界である。したがって、現在ピクチャーの単独のタイルは、現在ピクチャーと同じサイズであり得る。ビデオコーダは、現在ピクチャー、したがって現在ピクチャーの単一のタイルを複数のWPPウェーブに分割し得る。

【0067】

上述のように、ビデオエンコーダ20は、スライスの符号化表現を含むコード化スライスNALユニットを生成し得る。スライスは、整数個の連続するコード化ツリーブロックに関連付けられ得る。コード化スライスNALユニットは、スライスヘッダとスライスデータとを含み得る。スライスデータは、スライスに関連付けられた各ツリーブロックの符号化表現を含み得る。ビデオエンコーダ20は、ツリーブロックが属するピクチャーパーティションに従って、ツリーブロックの符号化表現がスライスデータ内でセグメントにグループ化されるように、コード化スライスNALユニットを生成し得る。たとえば、コード化スライスNALユニットは、第1のピクチャーパーティションに関連付けられた各コード化ツリーブロック、その後続く第2のピクチャーパーティションに関連付けられた各コード化ツリーブロック、その後続く第3のピクチャーパーティションに関連付けられた各コード化ツリーブロックなどを含み得る。

20

【0068】

本開示の技法によれば、ビデオエンコーダ20は、セグメントのそれぞれがバイト境界上で開始するようにセグメントのうちの1つまたは複数をパディングし得る。コード化スライスNALユニットは一連のバイトに分割され得る。セグメントは、セグメントの第1のビットがコード化スライスNALユニットのバイトのうちの1つの第1のビットであるとき、バイト境界上で開始し得る。さらに、セグメントは、セグメントの第1のビットがコード化スライスNALユニットのバイトのうちの1つの第1のビットである場合、バイト・アライン(byte aligned)され得る。ビデオエンコーダ20がセグメントをパディングするとき、ビデオエンコーダ20は、セグメントにパディングビットを付加し得る。たとえば、ビデオエンコーダ20は、剰余を残すことなしにセグメント中のビット数が8で割り切れるように、セグメントに1つまたは複数のパディングビットを追加し得る。パディングビットは、いかなる意味論的意味をも有しないことがあるが、次のセグメントがバイト境界において開始することを保証するのに役立ち得る。

30

40

【0069】

ビデオデコーダ30がコード化スライスNALユニットを受信するとき、ビデオエンコーダ30は、コード化スライスNALユニットをメモリに記憶し得る。ピクチャーパーティションを並列に復号するために、ビデオデコーダ30は、並列に動作する異なる復号スレッドにセグメントを割り当て得る。異なる復号スレッドにセグメントを割り当てるために、ビデオデコーダ30は、セグメントの始端に関連付けられたメモリアドレスを示す必要があり得る。ビデオデコーダ30は、バイト単位のメモリアドレス指定を使用し得る。したがって、ビデオデコーダ30は、セグメントの始まりがバイト内で発生する場合、セ

50

グメントの始まりに関連付けられたメモリアドレスを示すことができないことがある。したがって、ビデオデコーダ30は、セグメントのうちの1つまたは複数がバイト内で開始する場合、セグメント中のコード化ツリーブロックを並列に復号することができないことがある。代替として、ビデオデコーダ30は、セグメント中のコード化ツリーブロックを並列に復号することを可能にするためにビット単位のメモリアドレス指定またはバイト単位+ビット単位のアドレス指定を使用し得るが、実装および計算の複雑さの増加を伴う。

【0070】

このようにして、ビデオエンコーダ20は、ピクチャーを複数のピクチャーパーティションに分割し得る。ピクチャーは、複数のツリーブロックを有する。ピクチャーパーティションは、ピクチャーのツリーブロックのオーバーラップしないサブセットに関連付けられる。ビデオエンコーダ20は、ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスNALユニットを生成し得る。ツリーブロックの符号化表現は、コード化スライスNALユニット内で上記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化される。セグメントのそれぞれがバイト境界上で開始するようにセグメントのうちの1つまたは複数

10

【0071】

その上、ビデオデコーダ30は、ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスNALユニットを記憶し得る。ピクチャーは、複数のピクチャーパーティションに分割され得る。ツリーブロックの符号化表現は、上記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化され得る。セグメントのそれぞれがバイト境界において開始するようにセグメントのうちの1つまたは複数

20

【0072】

図2は、本開示の技法を実装するように構成された例示的なビデオエンコーダ20を示すブロック図である。図2は、説明のために与えられており、本開示で広く例示および説明する技法を限定するものと見なされるべきではない。説明のために、本開示では、HEVCコーディングのコンテキストにおいてビデオエンコーダ20について説明する。ただし、本開示の技法は、他のコーディング規格または方法に適用可能であり得る。

30

【0073】

図2の例では、ビデオエンコーダ20は、複数の機能構成要素を含む。ビデオエンコーダ20の機能構成要素は、予測モジュール(prediction module)100と、残差生成モジュール(residual generation module)102と、変換モジュール(transform module)104と、量子化モジュール(quantization module)106と、逆量子化モジュール(inverse quantization module)108と、逆変換モジュール(inverse transform module)110と、再構成モジュール(reconstruction module)112と、フィルタモジュール(filter module)113と、復号化ピクチャーバッファ(decoded picture buffer)114と、エントロピー符号化モジュール(entropy encoding module)116とを含む。予測モジュール100は、インター予測モジュール(inter prediction module)121と、動き推定モジュール(motion estimation module)122と、動き補償モジュール(motion compensation module)124と、イントラ予測モジュール(intra prediction module)126とを含む。他の例では、ビデオエンコーダ20は、より多数の、より少数の、または異なる機能構成要素を含み得る。さらに、動き推定モジュール122と動き補償モジュール124は、高度に統合され得るが、図2の例では、説明のために別々に表されている。

40

【0074】

ビデオエンコーダ20は、ビデオデータを受信し得る。ビデオエンコーダ20は、様々なソースからビデオデータを受信し得る。たとえば、ビデオエンコーダ20は、ビデオソ

50

ース 18 (図 1) または別のソースからビデオデータを受信し得る。ビデオデータは一連のピクチャーを表し得る。ビデオデータを符号化するために、ビデオエンコーダ 20 は、ピクチャーのそれぞれに対して符号化演算を実行し得る。ピクチャーに対して符号化演算を実行することの一部として、ビデオエンコーダ 20 は、ピクチャーの各スライスに対して符号化演算を実行し得る。スライスに対する符号化演算を実行することの一部として、ビデオエンコーダ 20 は、スライス中のツリーブロックに対して符号化演算を実行し得る。

【 0 0 7 5 】

ツリーブロックに対して符号化演算を実行することの一部として、予測モジュール 100 は、ツリーブロックのビデオブロックに対して 4 分木区分を実行して、ビデオブロックを徐々により小さいビデオブロックに分割し得る。より小さいビデオブロックのそれぞれは、異なる CU に関連付けられ得る。たとえば、予測モジュール 100 は、ツリーブロックのビデオブロックを 4 つの等しいサイズのサブブロックに区分し、サブブロックのうちの 1 つまたは複数を、4 つの等しいサイズのサブサブブロックに区分し得、以下同様である。

【 0 0 7 6 】

CU に関連付けられたビデオブロックのサイズは、 8×8 サンプルから、ツリーブロックのサイズまでにわたり、最大で 64×64 サンプル以上であり得る。本開示では、「 $N \times N$ ($N \times N$)」および「 $N \times N$ (N by N)」は、垂直ディメンションおよび水平ディメンションに関するビデオブロックのサンプル・ディメンション、たとえば、 16×16 (16×16) サンプルまたは 16×16 (16 by 16) サンプルを指すために互換的に使用され得る。一般に、 16×16 のビデオブロックは、垂直方向に 16 個のサンプルを有し ($y = 16$)、水平方向に 16 個のサンプルを有する ($x = 16$)。同様に、 $N \times N$ のブロックは、一般に、垂直方向に N 個のサンプルを有し、水平方向に N 個のサンプルを有し、ここで、 N は非負整数値を表す。

【 0 0 7 7 】

さらに、ツリーブロックに対して符号化演算を実行することの一部として、予測モジュール 100 は、ツリーブロック用の階層的な 4 分木データ構造を生成し得る。たとえば、ツリーブロックは、4 分木データ構造のルートノードに対応し得る。予測モジュール 100 が、ツリーブロックのビデオブロックを 4 つのサブブロックに区分する場合、ルートノードは、4 分木データ構造中に 4 つの子ノードを有する。子ノードのそれぞれは、サブブロックのうちの 1 つに関連付けられた CU に対応する。予測モジュール 100 が、サブブロックのうちの 1 つを 4 つのサブサブブロックに区分する場合、サブブロックに関連付けられた CU に対応するノードは、サブサブブロックのうちの 1 つに関連付けられた CU にそれぞれが対応する、4 つの子ノードを有し得る。

【 0 0 7 8 】

4 分木データ構造の各ノードは、対応するツリーブロックまたは CU のシンタックスデータ (たとえば、シンタックス要素) を含み得る。たとえば、4 分木中のノードは、そのノードに対応する CU のビデオブロックが 4 つのサブブロックに区分される (すなわち、分割される) かどうかを示す分割フラグを含み得る。CU のシンタックス要素は、再帰的に定義され得、CU のビデオブロックがサブブロックに分割されるかどうか依存し得る。そのビデオブロックが区分されていない CU は、4 分木データ構造におけるリーフノードに対応し得る。コード化ツリーブロックは、対応するツリーブロック用の 4 分木データ構造に基づくデータを含み得る。

【 0 0 7 9 】

ビデオエンコーダ 20 は、ツリーブロックの区分されていない各 CU に対して符号化演算を実行し得る。ビデオエンコーダ 20 が区分されていない CU に対して符号化演算を実行するとき、ビデオエンコーダ 20 は、区分されていない CU の符号化表現を表すデータを生成する。

【 0 0 8 0 】

C Uに対して符号化演算を実行することの一部として、予測モジュール100は、C Uの1つまたは複数のP Uの中で、C Uのビデオブロックを区分し得る。ビデオエンコーダ20およびビデオデコーダ30は、様々なP Uサイズをサポートし得る。特定のC Uのサイズが $2N \times 2N$ であると仮定すると、ビデオエンコーダ20およびビデオデコーダ30は、 $2N \times 2N$ または $N \times N$ のP Uサイズと、 $2N \times 2N$ 、 $2N \times N$ 、 $N \times 2N$ 、 $N \times N$ 、 $2N \times nU$ 、 $nL \times 2N$ 、 $nR \times 2N$ 、または同様の対称P Uサイズでのインター予測とをサポートし得る。ビデオエンコーダ20およびビデオデコーダ30は、 $2N \times nU$ 、 $2N \times nD$ 、 $nL \times 2N$ 、および $nR \times 2N$ のP Uサイズに対する非対称区分 (asymmetric partitioning) をもサポートし得る。いくつかの例では、予測モジュール100は、C Uのビデオブロックの辺に直角で接触 (meet) しない境界に沿ったC Uの複数のP Uの間で、C Uのビデオブロックを区分するために、幾何学的な区分 (geometric partitioning) を実行し得る。

10

【0081】

インター予測モジュール121は、C Uの各P Uに対してインター予測を実行し得る。インター予測は、時間圧縮を実現し得る。P Uに対してインター予測を実行するために、動き推定モジュール122は、P Uの動き情報を生成し得る。動き補償モジュール124は、動き情報と、C Uに関連付けられたピクチャー以外のピクチャー (すなわち、参照ピクチャー) の復号化サンプルとに基づいて、P Uの予測されたビデオブロックを生成し得る。本開示では、動き補償モジュール124によって生成された予測されたビデオブロックは、インター予測されたビデオブロックと呼ばれることがある。

20

【0082】

スライスは、Iスライス、Pスライス、またはBスライスであり得る。動き推定モジュール122および動き補償モジュール124は、P UがIスライス中にあるか、Pスライス中にあるか、Bスライス中にあるかに応じて、C UのP Uに対して異なる演算を実行し得る。Iスライス中では、すべてのP Uがイントラ予測される。したがって、P UがIスライス中にある場合、動き推定モジュール122および動き補償モジュール124は、P Uに対してインター予測を実行しない。

【0083】

P UがPスライス中にある場合、P Uを含んでいるピクチャーは、「リスト0」 (list 0) と呼ばれる参照ピクチャーのリストに関連付けられる。リスト0中の参照ピクチャーのそれぞれは、他のピクチャーのインター予測に使用され得るサンプルを含んでいる。動き推定モジュール122が、Pスライス中のP Uに関して動き推定演算を実行するとき、動き推定モジュール122は、P Uの参照ブロックについて、リスト0中の参照ピクチャーを探索し得る。P Uの参照ブロックは、P Uのビデオブロック中のサンプルに最も密接に (most closely) 対応するサンプルのセット、たとえば、サンプルのブロックであり得る。動き推定モジュール122は、様々なメトリックを使用して、参照ピクチャー中のサンプルのセットがどの程度密接にP Uのビデオブロック中のサンプルに対応するかを判定し得る。たとえば、動き推定モジュール122は、絶対差分和 (sum of absolute difference) (SAD)、2乗差分和 (sum of square difference) (SSD)、または他の差分メトリックによって、参照ピクチャー中のサンプルのセットがどの程度密接にP Uのビデオブロック中のサンプルに対応するかを判定し得る。

30

40

【0084】

Pスライス中のP Uの参照ブロックを識別した後、動き推定モジュール122は、参照ブロックを含んでいる、リスト0中の参照ピクチャーを示す参照インデックスと、P Uと参照ブロックとの間の空間変位 (spatial displacement) を示す動きベクトルとを生成し得る。様々な例において、動き推定モジュール122は、動きベクトルを可変の精度で生成し得る。たとえば、動き推定モジュール122は、 $1/4$ サンプル精度、 $1/8$ サンプル精度、または他の分数のサンプル精度で、動きベクトルを生成し得る。分数のサンプル精度の場合、参照ブロック値は、参照ピクチャー中の整数位置のサンプル値から補間され得る。動き推定モジュール122は、P Uの動き情報として参照インデックスと動きベク

50

トルとを出力し得る。動き補償モジュール124は、PUの動き情報によって識別される参照ブロックに基づいてPUの予測されたビデオブロックを生成し得る。

【0085】

PUがBスライス中にある場合、PUを含んでいるピクチャーは、「リスト0」(list 0)および「リスト1」(list 1)と呼ばれる参照ピクチャーの2つのリストに関連付けられ得る。いくつかの例では、Bスライスを含んでいるピクチャーは、リスト0とリスト1の組み合わせである、リストの組み合わせに関連付けられ得る。

【0086】

さらに、PUがBスライス中にある場合、動き推定モジュール122は、PUについての単方向予測または双方向予測を実行し得る。動き推定モジュール122が、PUについての単方向予測を実行するとき、動き推定モジュール122は、PUの参照ブロックについて、リスト0またはリスト1の参照ピクチャーを探索し得る。動き推定モジュール122は次いで、参照ブロックを含んでいる、リスト0またはリスト1中の参照ピクチャーを示す参照インデックスと、PUと参照ブロックの間の空間変位を示す動きベクトルとを生成し得る。動き推定モジュール122は、PUの動き情報として参照インデックスと、予測方向インジケータと、動きベクトルとを出力し得る。予測方向インジケータは、参照インデックスが、リスト0中の参照ピクチャーを示すか、リスト1中の参照ピクチャーを示すかを示し得る。動き補償モジュール124は、PUの動き情報によって示される参照ブロックに基づいてPUの予測されたビデオブロックを生成し得る。

【0087】

動き推定モジュール122が、PUについての双方向予測を実行するとき、動き推定モジュール122は、PUの参照ブロックについて、リスト0中の参照ピクチャーを探索し得、また、PUの別の参照ブロックについて、リスト1中の参照ピクチャーを探索し得る。動き推定モジュール122は次いで、参照ブロックを含んでいる、リスト0およびリスト1中の参照ピクチャーを示す参照インデックスと、参照ブロックとPUの間の空間変位を示す動きベクトルとを生成し得る。動き推定モジュール122は、PUの動き情報としてPUの参照インデックスと動きベクトルとを出力し得る。動き補償モジュール124は、PUの動き情報によって示される参照ブロックに基づいてPUの予測されたビデオブロックを生成し得る。

【0088】

いくつかの例では、動き推定モジュール122は、PUの動き情報のフルセットをエン트로ピー符号化モジュール116に出力しない。そうではなく、動き推定モジュール122は、別のPUの動き情報を参照して、PUの動き情報をシグナリングし得る。たとえば、動き推定モジュール122は、PUの動き情報が、隣接PUの動き情報と十分に類似していると判定し得る。この例では、動き推定モジュール122は、PUに関連付けられたシンタックス構造において、PUが隣接PUと同じ動き情報を有することをビデオデコーダ30に示す値を、示し得る。別の例では、動き推定モジュール122は、PUに関連付けられたシンタックス構造において、隣接PUと動きベクトル差分(MVD: motion vector difference)とを識別し得る。動きベクトル差分は、PUの動きベクトルと、示される隣接PUの動きベクトルとの間の差分を示す。ビデオデコーダ30は、示される隣接PUの動きベクトルと、動きベクトル差分とを使用して、PUの動きベクトルを判定し得る。第2のPUの動き情報をシグナリングするときに第1のPUの動き情報を参照することによって、ビデオエンコーダ20は、より少数のビットを使用して、第2のPUの動き情報をシグナリングすることが可能であり得る。

【0089】

CUに対して符号化演算を実行することの一部として、イントラ予測モジュール126は、CUのPUに対してイントラ予測を実行し得る。イントラ予測は空間圧縮を実現し得る。イントラ予測モジュール126がPUに対してイントラ予測を実行するとき、イントラ予測モジュール126は、同じピクチャー中の他のPUの復号化サンプルに基づいて、PUの予測データを生成し得る。PUの予測データは、予測されたビデオブロックと様々

10

20

30

40

50

なシンタックス要素とを含み得る。イントラ予測モジュール126は、Iスライス、Pスライス、およびBスライス中のPUに対してイントラ予測を実行し得る。

【0090】

PUに対してイントラ予測を実行するために、イントラ予測モジュール126は、複数のイントラ予測モードを使用して、PUの予測データの複数のセットを生成し得る。イントラ予測モジュール126が、イントラ予測モードを使用してPUの予測データのセットを生成するとき、イントラ予測モジュール126は、イントラ予測モードに関連付けられた方向(direction)および/または勾配(gradient)で、隣接PUのビデオブロックからPUのビデオブロックにわたって、サンプルを延ばし(extend)得る。隣接PUは、PU、CUおよびツリーブロックについて左から右、上から下の符号化順序を仮定すると、PUの上、右上、左上、または左にあり得る。イントラ予測モジュール126は、PUのサイズに応じて、様々な数のイントラ予測モード、たとえば、33個の方向性イントラ予測モードを使用し得る。

10

【0091】

予測モジュール100は、PUについての、動き補償モジュール124によって生成された予測データ、またはPUについての、イントラ予測モジュール126によって生成された予測データの中から、PUの予測データを選択し得る。いくつかの例では、予測モジュール100は、予測データのセットのレート/ひずみ(distortion)メトリックに基づいて、PUの予測データを選択する。

【0092】

予測モジュール100が、イントラ予測モジュール126によって生成された予測データを選択する場合、予測モジュール100は、PUの予測データを生成するために使用されたイントラ予測モード、すなわち、選択されたイントラ予測モードをシグナリングし得る。予測モジュール100は、選択されたイントラ予測モードを様々な方法でシグナリングし得る。たとえば、選択されたイントラ予測モードは、隣接PUのイントラ予測モードと同じであることがあり得る。言い換えれば、隣接PUのイントラ予測モードは、現在PUに対して最もあり得るモードであり得る。したがって、予測モジュール100は、選択されたイントラ予測モードが隣接PUのイントラ予測モードと同じであることを示すための、シンタックス要素を生成し得る。

20

【0093】

予測モジュール100がCUのPUの予測データを選択した後、残差生成モジュール102は、CUのビデオブロックからCUのPUの予測されたビデオブロックを差し引くことによって、CUの残差データを生成し得る。CUの残差データは、CUのビデオブロック中のサンプルの様々なサンプル成分に対応する、2D残差ビデオブロックを含み得る。たとえば、残差データは、CUのPUの予測されたビデオブロック中のサンプルのルミナンス成分(luminance components)と、CUの元のビデオブロック中のサンプルのルミナンス成分との間の差に対応する、残差ビデオブロックを含み得る。加えて、CUの残差データは、CUのPUの予測されたビデオブロック中のサンプルのクロミナンス成分(chrominance components)と、CUの元のビデオブロック中のサンプルのクロミナンス成分との間の差に対応する、残差ビデオブロックを含み得る。

30

40

【0094】

予測モジュール100は、4分木区分を実行して、CUの残差ビデオブロックをサブブロックに区分し得る。分割されていない各残差ビデオブロックは、CUの異なるTUに関連付けられ得る。CUのTUに関連付けられた残差ビデオブロックのサイズおよび位置は、CUのPUに関連付けられたビデオブロックのサイズおよび位置に基づくことも基づかないこともある。「残差4分木」(RQT: residual quad tree)として知られる4分木構造は、残差ビデオブロックのそれぞれに関連付けられたノードを含み得る。CUのTUは、RQTのリーフノードに対応し得る。

【0095】

変換モジュール104は、TUに関連付けられた残差ビデオブロックに1つまたは複数

50

の変換を適用することによって、CUの各TUについて1つまたは複数の変換係数ブロックを生成し得る。変換係数ブロックのそれぞれは、変換係数の2D行列であり得る。変換モジュール104は、TUに関連付けられた残差ビデオブロックに様々な変換を適用し得る。たとえば、変換モジュール104は、離散コサイン変換(DCT)、方向性変換、または概念的に同様の変換を、TUに関連付けられた残差ビデオブロックに適用し得る。

【0096】

変換モジュール104が、TUに関連付けられた変換係数ブロックを生成した後、量子化モジュール106は、変換係数ブロック中の変換係数を量子化し得る。量子化モジュール106は、CUに関連付けられたQP値に基づいて、CUのTUに関連付けられた変換係数ブロックを量子化し得る。

10

【0097】

ビデオエンコーダ20は、様々な方法でQP値をCUに関連付け得る。たとえば、ビデオエンコーダ20は、CUに関連付けられたツリーブロックに対して、レートひずみ分析を実行し得る。レートひずみ分析では、ビデオエンコーダ20は、ツリーブロックに対して符号化演算を複数回実行することによって、ツリーブロックの複数のコード化表現を生成し得る。ビデオエンコーダ20がツリーブロックの異なる符号化表現を生成するとき、ビデオエンコーダ20は、異なるQP値をCUに関連付け得る。ビデオエンコーダ20は、最小のビットレートおよびひずみメトリックを有するツリーブロックのコード化表現で所与のQP値がCUに関連付けられるとき、所与のQP値がCUに関連付けられることをシグナリングし得る。

20

【0098】

逆量子化モジュール108および逆変換モジュール110は、それぞれ、変換係数ブロックに逆量子化および逆変換を適用して、変換係数ブロックから残差ビデオブロックを再構成し得る。再構成モジュール112は、再構成された残差ビデオブロックを、予測モジュール100によって生成される1つまたは複数の予測されたビデオブロックからの対応するサンプルに追加して、TUに関連付けられた再構成されたビデオブロックを生成し得る。このようにCUの各TUについてビデオブロックを再構成することによって、ビデオエンコーダ20は、CUのビデオブロックを再構成し得る。

【0099】

再構成モジュール112がCUのビデオブロックを再構成した後、フィルタモジュール113は、デブロッキング演算(deblocking operation)を実行して、CUに関連付けられたビデオブロックにおけるブロッキングアーティファクト(blocking artifacts)を低減し得る。1つまたは複数のデブロッキング演算を実行した後、フィルタモジュール113は、復号化ピクチャバッファ114にCUの再構成されたビデオブロックを記憶し得る。動き推定モジュール122および動き補償モジュール124は、再構成されたビデオブロックを含んでいる参照ピクチャを使用して、後続のピクチャのPUに対してインター予測を実行し得る。加えて、イントラ予測モジュール126は、復号化ピクチャバッファ114中の再構成されたビデオブロックを使用して、CUと同じピクチャの中他のPUに対してイントラ予測を実行し得る。

30

【0100】

エントロピー符号化モジュール116は、ビデオエンコーダ20の他の機能構成要素からデータを受信し得る。たとえば、エントロピー符号化モジュール116は、量子化モジュール106から変換係数ブロックを受信し得、予測モジュール100からシンタックス要素を受信し得る。エントロピー符号化モジュール116がデータを受信するとき、エントロピー符号化モジュール116は、1つまたは複数のエントロピー符号化演算を実行して、エントロピー符号化データを生成し得る。たとえば、ビデオエンコーダ20は、コンテキスト適応型可変長コーディング(CAVLC: context adaptive variable length coding)演算、CABAC演算、変数-変数(V2V: variable-to-variable)レンジスコーディング演算、シンタックススペースのコンテキスト適応型バイナリ算術コーディング(SBAC: syntax-based context-adaptive binary arithmetic coding)演算、確率間

40

50

隔区分エントロピー (P I P E : Probability Interval Partitioning Entropy) コーディング演算、または別のタイプのエントロピー符号化演算をデータに対して実行し得る。エントロピー符号化モジュール 116 は、エントロピー符号化データを含むビットストリームを出力し得る。

【 0 1 0 1 】

データに対してエントロピー符号化演算を実行することの一部として、エントロピー符号化モジュール 116 は、コンテキストモデルを選択し得る。エントロピー符号化モジュール 116 が C A B A C 演算を実行している場合、コンテキストモデルは、特定の値を有する特定のピンの確率の推定値を示し得る。C A B A C のコンテキストでは、「ピン」 (bin) という用語は、シンタックス要素の 2 値化されたバージョンのビットを指すために使用される。

10

【 0 1 0 2 】

ビデオエンコーダ 20 は、現在ピクチャーの各スライスについてコード化スライス N A L ユニットを生成し得る。スライスのためのコード化スライス N A L ユニットは、スライスヘッダとスライスデータとを含み得る。スライスデータは、複数のセグメントを含み得る。セグメントのそれぞれは、異なるピクチャーパーティションに関連付けられたコード化ツリーブロックを含む。ビデオエンコーダ 20 は、セグメントのそれぞれがスライスデータ内のバイト境界において開始するようにセグメントをパディングし得る。たとえば、コード化スライス N A L ユニット中のセグメントは所与のセグメントを含み得る。この例では、ビデオエンコーダ 20 は、少なくとも部分的に、次のツリーブロックが、現在スライス内にあり、所与のセグメントとは異なるピクチャーパーティションに関連付けられる場合、所与のセグメントにビットを付加するパディング演算を実行することによって、コード化スライス N A L ユニットを生成し得る。

20

【 0 1 0 3 】

いくつかの例では、ビデオエンコーダ 20 は、コード化スライス N A L ユニットのスライスヘッダがコード化スライス N A L ユニットのスライスデータ中のセグメントのためのエン트리ポイントを示すようにスライスヘッダを生成し得る。エン트리ポイントは、セグメントのスライスデータ内の位置を示し得る。たとえば、エン트리ポイントは、セグメントのバイトオフセットを示し得る。この例では、バイトオフセットは、コード化スライス N A L ユニットの第 1 のビット、スライスデータの第 1 のビット、またはコード化スライス N A L ユニット中の別のビットに対するバイトオフセットであり得る。別の例では、エン트리ポイントは、セグメントのそれぞれ内のビットまたはバイトの数を示し得る。いくつかの例では、スライスヘッダは、スライスデータ中の第 1 のセグメントのためのエン트리ポイントを示さない。

30

【 0 1 0 4 】

いくつかの例では、ビデオエンコーダ 20 は、フラグが第 1 の値 (たとえば、1) を有するかどうかを判定し得る。フラグが第 1 の値を有する場合、ビデオエンコーダ 20 は、各セグメントがバイト境界において開始するようにセグメントのうちの 1 つまたは複数をパディングし得る。フラグが第 2 の値 (たとえば、0) を有するとき、ビデオエンコーダ 20 は、セグメントをパディングしない。その結果、セグメントは、バイト・アラインされた位置 (byte-aligned positions) において開始することも開始しないこともある。そのような例では、シーケンスパラメータセット、ピクチャーパラメータセット、適応パラメータセット、またはスライスヘッダがフラグを含み得る。したがって、いくつかの例では、ビデオエンコーダ 20 は現在ピクチャーに関連付けられたパラメータセットを生成し得、パラメータセットはフラグを含む。フラグが第 1 の値を有するとき、セグメントがバイト境界において開始するようにセグメントのうちの 1 つまたは複数がパディングされる。フラグが第 2 の値を有するとき、セグメントはバイト境界において開始することも開始しないこともある。

40

【 0 1 0 5 】

さらに、いくつかの例では、ビデオエンコーダ 20 は現在ピクチャーを複数のタイルに

50

区分し得る。ビデオエンコーダ 20 がタイル境界を越えるピクチャー内予測を可能にする場合（すなわち、タイルのうちの 2 つ以上が互いに依存するとき）、ビデオエンコーダ 20 はセグメントをパディングしない。その結果、セグメントは、バイト・アラインされた位置において開始することも開始しないこともある。しかしながら、ビデオエンコーダ 20 がタイル境界を越えるピクチャー内予測を可能にしない場合、ビデオエンコーダ 20 は、セグメントのそれぞれがバイト境界において開始するようにセグメントのうちの 1 つまたは複数をパディングし得る。したがって、ビデオエンコーダ 20 は、少なくとも部分的に、タイルが互いから独立していると判定した後のみ、セグメントがバイト境界において開始することを保証するパディング演算を実行することによって、コード化スライス NAL ユニットを生成し得る。

10

【0106】

図 3 は、本開示の技法を実装するように構成された例示的なビデオデコーダ 30 を示すブロック図である。図 3 は、説明のために与えられており、本開示で広く例示および説明する技法に対する限定ではない。説明のために、本開示では、HEVC コーディングのコンテキストにおいてビデオデコーダ 30 について説明する。ただし、本開示の技法は、他のコーディング規格または方法に適用可能であり得る。

【0107】

図 3 の例では、ビデオデコーダ 30 は、複数の機能構成要素を含む。ビデオデコーダ 30 の機能構成要素は、エントロピー復号モジュール (entropy decoding module) 150 と、予測モジュール (prediction module) 152 と、逆量子化モジュール (inverse quantization module) 154 と、逆変換モジュール (inverse transform module) 156 と、再構成モジュール (reconstruction module) 158 と、フィルタモジュール (filter module) 159 と、復号化ピクチャーバッファ (decoded picture buffer) 160 とを含む。予測モジュール 152 は、動き補償モジュール (motion compensation module) 162 と、イントラ予測モジュール (intra prediction module) 164 とを含む。いくつかの例では、ビデオデコーダ 30 は、図 2 のビデオエンコーダ 20 に関して説明した符号化パスとは概して逆の復号パスを実行し得る。他の例では、ビデオデコーダ 30 は、より多数の、より少数の、または異なる機能構成要素を含み得る。

20

【0108】

ビデオデコーダ 30 は、符号化ビデオデータを含むビットストリームを受信し得る。ビットストリームは、複数のシンタックス要素を含み得る。ビデオデコーダ 30 がビットストリームを受信するとき、エントロピー復号モジュール 150 は、ビットストリームに対してパース演算を実行し得る。ビットストリームに対してパース演算を実行した結果として、エントロピー復号モジュール 150 は、ビットストリームからシンタックス要素を抽出し得る。パース演算を実行することの一部として、エントロピー復号モジュール 150 は、ビットストリーム中のエントロピー符号化シンタックス要素をエントロピー復号し得る。予測モジュール 152、逆量子化モジュール 154、逆変換モジュール 156、再構成モジュール 158、およびフィルタモジュール 159 は、ビットストリームから抽出されたシンタックス要素に基づいて復号ビデオデータを生成する再構成演算 (reconstruction operation) を実行し得る。

30

40

【0109】

上記で説明したように、ビットストリームは一連の NAL ユニットを含み得る。ビットストリームの NAL ユニットは、シーケンスパラメータセット NAL ユニット、ピクチャーパラメータセット NAL ユニット、SEI NAL ユニットなどを含み得る。ビットストリームに対してパース演算を実行することの一部として、エントロピー復号モジュール 150 は、シーケンスパラメータセット NAL ユニットからのシーケンスパラメータセット、ピクチャーパラメータセット NAL ユニットからのピクチャーパラメータセット、SEI NAL ユニットからの SEI データなどを抽出しエントロピー復号するパース演算を実行し得る。

【0110】

50

加えて、ビットストリームのNALユニットは、コード化スライスNALユニットを含み得る。ビットストリームに対してパース演算を実行することの一部として、ビデオデコーダ30は、コード化スライスNALユニットからコード化スライスを抽出しエントロピー復号するパース演算を実行し得る。コード化スライスのそれぞれは、スライスヘッダとスライスデータとを含み得る。スライスヘッダは、スライスに関するシンタックス要素を含んでいることがある。スライスヘッダ中のシンタックス要素は、スライスを含んでいるピクチャーに関連付けられたピクチャーパラメータセットを識別するシンタックス要素を含み得る。

【0111】

コード化スライスNALユニットのスライスデータは、複数のセグメントを含み得る。セグメントのそれぞれは、異なるピクチャーパーティション（たとえば、タイルまたはWPPウェーブ）に関連付けられたコード化ツリーブロックを含み得る。スライスデータ中のセグメントのそれぞれがバイト境界において開始するようにセグメントのうちの1つまたは複数がパディングされ得る。コード化スライスNALユニットのスライスヘッダは、セグメントのためのエントリポイントを示し得る。この場合、セグメントが常にバイト境界において開始するので、ビデオデコーダ30は、バイト単位のメモリアドレス指定を使用することによって単純な方式で異なる復号スレッドに上記セグメントのうちの異なるセグメントを割り当てることが可能であり得る。異なる復号スレッドは、セグメントのコード化ツリーブロックをパースし、対応するツリーブロックに関連付けられたビデオデータを並列に再構成し得る。

【0112】

コード化スライスNALユニットからスライスデータを抽出することの一部として、エントロピー復号モジュール150は、コード化CUからシンタックス要素を抽出するパース演算を実行し得る。抽出されたシンタックス要素は、変換係数ブロックに関連付けられたシンタックス要素を含み得る。エントロピー復号モジュール150は、次いでシンタックス要素のうちのいくつかに対してCABAC復号演算を実行し得る。

【0113】

エントロピー復号モジュール150が区分されていないCUに対してパース演算を実行した後、ビデオデコーダ30は、区分されていないCUに対して再構成演算を実行し得る。区分されていないCUに対して再構成演算を実行するために、ビデオデコーダ30は、CUの各TUに対して再構成演算を実行し得る。CUの各TUについて再構成演算を実行することによって、ビデオデコーダ30は、CUに関連付けられた残差ビデオブロックを再構成し得る。

【0114】

TUに対して再構成演算を実行することの一部として、逆量子化モジュール（inverse quantization module）154は、TUに関連付けられた変換係数ブロックを逆量子化（inverse quantize）、すなわち、逆量子化（de-quantize）し得る。逆量子化モジュール154は、HEVCのために提案される、またはH.264復号規格によって定義される逆量子化処理と同様の方式で、変換係数ブロックを逆量子化し得る。逆量子化モジュール154は、量子化のディグリー（degree of quantization）を判定し、同様に、逆量子化モジュール154が適用するべき逆量子化のディグリー（degree of inverse quantization）を判定するために、変換係数ブロックのCUのためにビデオエンコーダ20によって計算される量子化パラメータQPを使用し得る。

【0115】

逆量子化モジュール154が変換係数ブロックを逆量子化した後、逆変換モジュール156は、変換係数ブロックに関連付けられたTUの残差ビデオブロックを生成し得る。逆変換モジュール156は、TUの残差ビデオブロックを生成するために、逆変換を変換係数ブロックに適用し得る。たとえば、逆変換モジュール156は、逆DCT、逆整数変換（inverse integer transform）、逆カルーネンレーベ変換（inverse Karhunen-Loeve transform）（KLT）、逆回転変換（inverse rotational transform）、逆方向性変換（i

10

20

30

40

50

nverse directional transform)、または別の逆変換を変換係数ブロックに適用し得る。

【 0 1 1 6 】

いくつかの例では、逆変換モジュール 1 5 6 は、ビデオエンコーダ 2 0 からのシグナリングに基づいて、変換係数ブロックに適用すべき逆変換を判定し得る。そのような例では、逆変換モジュール 1 5 6 は、変換係数ブロックに関連付けられたツリーブロックの 4 分木のルートノードにおいてシグナリングされた変換に基づいて、逆変換を判定し得る。他の例では、逆変換モジュール 1 5 6 は、たとえばブロックサイズ、コーディングモードなどのような、1 つまたは複数のコーディング特性から、逆変換を推論し得る。いくつかの例では、逆変換モジュール 1 5 6 は、カスケード逆変換 (cascaded inverse transform) を適用し得る。

10

【 0 1 1 7 】

いくつかの例では、動き補償モジュール 1 6 2 は、補間フィルタ (interpolation filters) に基づく補間を実行することによって、PU の予測されたビデオブロックをリファイン (refine) し得る。サブサンプル精度 (sub-sample precision) をもつ動き補償に使用されるべき補間フィルタの識別子は、シンタックス要素中に含まれ得る。動き補償モジュール 1 6 2 は、PU の予測されたビデオブロックの生成中にビデオエンコーダ 2 0 によって使用されたのと同じ補間フィルタを使用して、参照ブロックのサブ整数サンプル (sub-integer samples) に対する補間される値を計算し得る。動き補償モジュール 1 6 2 は、受信したシンタックス情報に従って、ビデオエンコーダ 2 0 によって使用された補間フィルタを判定し、その補間フィルタを使用して、予測されたビデオブロックを生成し得る。

20

【 0 1 1 8 】

イントラ予測を使用して PU が符号化される場合、イントラ予測モジュール 1 6 4 は、イントラ予測を実行して、PU の予測されたビデオブロックを生成し得る。たとえば、イントラ予測モジュール 1 6 4 は、ビットストリーム中のシンタックス要素に基づいて、PU のイントラ予測モードを判定し得る。ビットストリームは、PU のイントラ予測モードを判定するためにイントラ予測モジュール 1 6 4 が使用し得るシンタックス要素を含み得る。

【 0 1 1 9 】

いくつかの事例では、シンタックス要素は、イントラ予測モジュール 1 6 4 が別の PU のイントラ予測モードを使用して現在 PU のイントラ予測モードを判定すべきであることを、示し得る。たとえば、現在 PU のイントラ予測モードが、隣接 PU のイントラ予測モードと同じであることがあり得る。言い換えれば、隣接 PU のイントラ予測モードは、現在 PU に対して最もあり得るモードであり得る。したがって、この例では、ビットストリームは、PU のイントラ予測モードが、隣接 PU のイントラ予測モードと同じであることを示す、小さいシンタックス要素を含み得る。イントラ予測モジュール 1 6 4 は、次いで、イントラ予測モードを使用して、空間的に隣接する PU のビデオブロックに基づいて、PU の予測データ (たとえば、予測されたサンプル) を生成し得る。

30

【 0 1 2 0 】

再構成モジュール 1 5 8 は、適用可能なとき、CU の TU に関連付けられた残差ビデオブロックと CU の PU の予測されたビデオブロックとを使用して、すなわち、イントラ予測データまたはインター予測データのいずれかを使用して、CU のビデオブロックを再構成し得る。したがって、ビデオデコーダ 3 0 は、ビットストリーム中のシンタックス要素に基づいて、予測されたビデオブロックと残差ビデオブロックとを生成し得、予測されたビデオブロックと残差ビデオブロックとに基づいて、ビデオブロックを生成し得る。

40

【 0 1 2 1 】

再構成モジュール 1 5 8 が CU のビデオブロックを再構成した後、フィルタモジュール 1 5 9 は、デブロッキング演算を実行して、CU に関連付けられたブロッキングアーティファクトを低減し得る。フィルタユニット 1 5 9 がデブロッキング演算を実行して CU に関連付けられたブロッキングアーティファクトを低減した後、ビデオデコーダ 3 0 は、復

50

号化ピクチャバッファ160にCUのビデオブロックを記憶し得る。復号化ピクチャバッファ160は、後続の動き補償、イントラ予測、および図1のディスプレイデバイス32などのディスプレイデバイス上での提示のために、参照ピクチャを与え得る。たとえば、ビデオデコーダ30は、復号化ピクチャバッファ160中のビデオブロックに基づいて、他のCUのPUに対して、イントラ予測演算またはインター予測演算を実行し得る。

【0122】

図4は、スライスのためのスライスデータを生成するための例示的な動作200を示すフローチャートである。ビデオエンコーダ20（図1および図2）などの、ビデオエンコーダが動作200を実行し得る。図4の例は一例にすぎない。他の例示的な動作が他の方法でスライスデータを生成し得る。

10

【0123】

ビデオエンコーダが動作200を開始した後、ビデオエンコーダは、ツリーブロックアドレスが現在スライスの初期ツリーブロックを識別するように、ツリーブロックアドレスを初期化し得る（202）。現在スライスは、ビデオエンコーダが現在符号化しているスライスであり得る。現在スライスの初期ツリーブロックは、現在ピクチャのためのツリーブロックコーディング順序に従って現在スライスに関連付けられた第1のツリーブロックであり得る。説明を簡単にするために、本開示では、ツリーブロックアドレスによって識別されるツリーブロックを現在ツリーブロックと呼ぶことがある。

【0124】

20

ビデオエンコーダは、現在スライスのためのコード化スライスNALユニットのスライスデータに、現在ツリーブロックのためのシンタックス要素を付加し得る（204）。現在ツリーブロックのためのシンタックス要素は、現在ツリーブロックの4分木中のシンタックス要素を含み得る。現在ツリーブロックの4分木中のシンタックス要素は、イントラ予測モードを示すシンタックス要素、動き情報、変換係数レベルを示すシンタックス要素などを含み得る。

【0125】

さらに、ビデオエンコーダは、現在スライス中にさらなるデータがあるかどうかを判定し得る（206）。ツリーブロックアドレスによって示されるツリーブロックが現在スライス内にある場合、現在スライス中にさらなるデータがあり得る。現在スライス中にさらなるデータがないと判定したことに応答して（206の「NO」）、ビデオエンコーダは、ビデオエンコーダがスライスデータに必要なシンタックス要素のすべてを追加したので、動作200を終了し得る。

30

【0126】

ビデオエンコーダは、様々な方法で、現在スライス中にさらなるデータがあるかどうかを判定し得る。たとえば、ビデオエンコーダは、ツリーブロックのためのシンタックス要素を出力するために関数「coding_tree（）」を呼び出し得る。この例では、関数「coding_tree（）」は、現在スライス中にさらなるデータがあるかどうかを示す「moreDataFlag」を返し得る。

【0127】

40

現在スライスに関連付けられたさらなるデータがあると判定したことに応答して（206の「YES」）、ビデオエンコーダは、現在ピクチャのタイルが独立しているかどうかと、現在スライスの次のツリーブロックが現在スライスの現在ツリーブロックとは異なるタイル中にあるかどうかとを判定し得る（208）。上記で説明したように、ピクチャのタイルは、ピクチャ内予測（たとえば、現在ピクチャ中のデータを使用するイントラ予測、インター予測、および現在ピクチャの他のタイルからのデータに基づくCABACコンテキスト選択）が禁止される場合、独立していることがある。ビデオエンコーダは、様々な方法で、現在ピクチャのタイルが独立しているかどうかを判定し得る。たとえば、現在ピクチャに関連付けられたシーケンスパラメータセットがシンタックス要素「tile_boundary_independence_idc」を含み得る。こ

50

の例では、「`tile__boundary__independence__idc`」が0に等しい場合、現在ピクチャーのタイルは独立しておらず、タイル境界を越えるピクチャー内予測は可能にされる。「`tile__boundary__independence__idc`」が0に等しい場合、スライス境界を越えるピクチャー内予測は依然として禁止され得る。「`tile__boundary__independence__idc`」が1に等しい場合、現在ピクチャーのタイルは独立しており、タイル境界を越えるピクチャー内予測は可能にされない。

【0128】

ビデオエンコーダは、現在スライスの次のツリーブロックが現在スライスの現在ツリーブロックとは異なるタイル中にあるかどうかを様々な方法で判定し得る。たとえば、ビデオエンコーダは、現在スライスの次のツリーブロックのツリーブロックアドレスを判定し得る。この例では、ビデオエンコーダは、次のツリーブロックのツリーブロックアドレスをパラメータとしてとり、次のツリーブロックが現在ツリーブロックとは異なるタイル中にあるかどうかを示す値「`newTileFlag`」を返す、関数「`NewTile(. . .)`」を呼び出し得る。

【0129】

現在ピクチャーのタイルが独立していないか、または次のツリーブロックが現在ツリーブロックとは異なるタイル中にある場合(208の「NO」)、ビデオエンコーダは、現在ピクチャーがWPPを使用して符号化されており、現在スライスの次のツリーブロックが現在スライスの現在ツリーブロックとは異なるWPPウェーブ中にあるかどうかを判定し得る(210)。ビデオエンコーダは、現在スライスの次のツリーブロックが現在スライスの現在ツリーブロックとは異なるWPPウェーブ中にあるかどうかを様々な方法で判定し得る。たとえば、ビデオエンコーダは、現在スライスの次のツリーブロックのツリーブロックアドレスを判定し得る。この例では、ビデオエンコーダは、次のツリーブロックのツリーブロックアドレスをパラメータとしてとり、次のツリーブロックが現在ツリーブロックとは異なるWPPウェーブ中にあるかどうかを示す値「`newWaveFlag`」を返す、関数「`NewWave(. . .)`」を呼び出し得る。

【0130】

現在ピクチャーがWPPを使用して符号化されており、次のツリーブロックが現在ツリーブロックとは異なるWPPウェーブ中にあると判定したことに応答して(210の「YES」)、または現在ピクチャーのタイルが独立しており、次のツリーブロックが現在ツリーブロックとは異なるタイル中にあると判定したことに応答して(208の「YES」)、ビデオエンコーダは、現在セグメントがバイト・アラインされているかどうかを判定し得る(212)。言い換えれば、ビデオエンコーダは、現在セグメントがバイト境界上で終了するかどうかを判定し得る。現在セグメントは、現在ツリーブロックが関連付けられたピクチャーパーティション(たとえば、タイルまたはWPPウェーブ)に関連付けられたセグメントである。現在セグメントがバイト・アラインされていないと判定したことに応答して(212の「NO」)、ビデオエンコーダは、現在セグメントの終端にパディングビットを付加し得る(214)。パディングビットは様々な値を有し得る。たとえば、パディングビットは、常に、1に等しい値を有し得る。他の例では、パディングビットは、常に、0に等しい値を有し得る。

【0131】

現在セグメントの終端にパディングビットを付加した後に、ビデオエンコーダは、現在セグメントがバイト・アラインされているかどうかを再び判定し得る(212)。このようにして、ビデオエンコーダは、現在セグメントがバイト・アラインされるまで、スライスデータの終端にパディングビットを付加し続け得る。

【0132】

スライスデータがバイト・アラインされていると判定したことに応答して(212の「YES」)、ビデオエンコーダはツリーブロックアドレスを更新し得る(216)。ビデオエンコーダは、ツリーブロックアドレスが現在ピクチャーのツリーブロックコーディン

10

20

30

40

50

グ順序に従って次のツリーブロックを示すようにツリーブロックアドレスを更新し得る。たとえば、ビデオエンコーダがツリーブロックアドレスを更新するとき、ツリーブロックアドレスは、ツリーブロックアドレスによって前に示されたツリーブロックの右のツリーブロックを識別し得る。以下で詳細に説明する、図7は、複数のタイルに区分されたピクチャーのための例示的なツリーブロックコーディング順序を示す概念図である。

【0133】

ツリーブロックアドレスを更新した後に、ビデオエンコーダは、現在スライス中にさらなるデータがあるかどうかを判定し得る(218)。現在スライス中にさらなるデータがあると判定したことに応答して(218の「YES」)、または現在ピクチャーがWPPを使用して符号化されておらず、次のツリーブロックが現在ツリーブロックとは異なるタイル中にないと判定したことに応答して(210の「NO」)、ビデオエンコーダは、スライスデータに現在ツリーブロックのためのシンタックス要素を付加し得る(204)。このようにして、ビデオエンコーダは、スライスデータに現在スライスの各ツリーブロックのためのシンタックス要素を付加し得、異なるピクチャーパーティションに関連付けられたセグメントがバイト境界において開始するようにセグメントがパディングされることを保証し得る。

10

【0134】

現在スライス中にさらなるデータがないと判定したことに応答して(218の「NO」)、ビデオエンコーダは、ビデオエンコーダがスライスデータに現在スライスのシンタックス要素のすべてを付加していることがあるので、動作200を終了し得る。

20

【0135】

図5は、コード化スライスNALユニットを復号するための例示的な動作250を示すフローチャートである。ビデオデコーダ30(図1および図3)などの、ビデオデコーダが動作250を実行し得る。図5の例は一例にすぎない。他の例示的な動作が、コード化スライスNALユニットを復号するために他の動作を実行し得る。

【0136】

図5の例では、ビデオデコーダはコード化スライスNALユニットをバイトアドレス指定のメモリに記憶し得る(252)。コード化スライスNALユニットは、スライスヘッダとスライスデータとを含み得る。スライスデータは、複数のセグメントを含み得る。各セグメントがバイト境界において開始するようにセグメントのうちの1つまたは複数のパディングされ得る。

30

【0137】

コード化スライスNALユニットをメモリに記憶した後に、ビデオデコーダは、コード化スライスNALユニットのスライスデータ内のセグメントの位置を識別し得る(254)。ビデオデコーダは、様々な方法でセグメントの位置を識別し得る。たとえば、ビデオデコーダは、セグメントのバイトオフセットを示すコード化スライスNALユニットのスライスヘッダ中のシンタックス要素に基づいてセグメントの位置を識別し得る。この例では、スライスヘッダは、スライスデータの第1のセグメントの位置がスライスヘッダの終端の直後にくることがあるので、第1のセグメントのためのバイトオフセットを含まないことがある。別の例では、ビデオデコーダは、スライスデータ中のエントリーポイント・マーカーに基づいてセグメントの位置を識別し得る。エントリーポイント・マーカーは、セグメント間に配置(disposed)された値であり得る。

40

【0138】

スライスデータ内のセグメントの位置を識別した後に、ビデオデコーダは、2つ以上の異なる復号スレッドにセグメントのうちの2つ以上を割り当て得る(256)。復号スレッドのそれぞれは、上記で説明したように、復号スレッドに割り当てられたセグメント中のコード化ツリーブロックのシンタックス要素をパースし、対応するツリーブロックのためのビデオブロックを再構成し得る。

【0139】

図6は、ウェイブフロント並列処理を示す概念図である。上記で説明したように、ピク

50

チャーは、ツリーブロックにそれぞれが関連付けられる、ビデオブロックに区分され得る。図6は、ツリーブロックに関連付けられたビデオブロックを白い正方形の格子として示している。ピクチャーは、ツリーブロック行300A~300E(総称して、「ツリーブロック行300」(treeblock rows 300))を含む。

【0140】

第1のスレッドが、ツリーブロック行300A中のツリーブロックをコーディングしていることがある。同時に、他のスレッドが、ツリーブロック行300B、300C、および300D中のツリーブロックをコーディングしていることがある。図6の例では、第1のスレッドはツリーブロック302Aを現在コーディングしており、第2のスレッドはツリーブロック302Bを現在コーディングしており、第3のスレッドはツリーブロック302Cを現在コーディングしており、第4のスレッドはツリーブロック302Dを現在コーディングしている。本開示では、ツリーブロック302A、302B、302C、および302Dを「現在ツリーブロック302」(current treeblocks 302)と総称することがある。ビデオコーダは、すぐ上の行の3つ以上のツリーブロックがコーディングされた後、ツリーブロック行をコーディングし始め得るので、現在ツリーブロック302は、2つのツリーブロックの幅だけ互いから水平方向に変位される。

【0141】

図6の例では、スレッドは、現在ツリーブロック302中のCUのためのイントラ予測またはインター予測を実行するとき、太いグレーの矢印によって示されるツリーブロックからのデータを使用し得る。(スレッドがCUのためのインター予測を実行するとき、スレッドは、1つまたは複数の参照フレームからのデータをも使用し得る。)スレッドが所与のツリーブロックをコーディングするとき、スレッドは、前にコーディングされたツリーブロックに関連付けられた情報に基づいて1つまたは複数のCABACコンテキストを選択し得る。スレッドは、所与のツリーブロックの第1のCUに関連付けられたシンタックス要素に対してCABACコーディングを実行するために1つまたは複数のCABACコンテキストを使用し得る。所与のツリーブロックが行の最左ツリーブロックでない場合、スレッドは、所与のツリーブロックの左のツリーブロックの最後のCUに関連付けられた情報に基づいて1つまたは複数のCABACコンテキストを選択し得る。所与のツリーブロックが行の最左ツリーブロックである場合、スレッドは、所与のツリーブロックの上の2つ右のツリーブロックの最後のCUに関連付けられた情報に基づいて1つまたは複数のCABACコンテキストを選択し得る。スレッドは、現在ツリーブロック302の第1のCUのためのCABACコンテキストを選択するために、細い黒い矢印によって示されるツリーブロックの最後のCUからのデータを使用し得る。

【0142】

図7は、複数のタイル352A、352B、および352Cに区分されたピクチャー350のための例示的なツリーブロックコーディング順序を示す概念図である。ピクチャー350中の各正方形の白いブロックが、ツリーブロックに関連付けられたビデオブロックを表す。太い垂直破線が例示的な垂直タイル境界を示す。太いグレーの線が例示的なスライス境界を示す。

【0143】

ビデオブロック中の番号が、ピクチャー350のためのツリーブロックコーディング順序における対応するツリーブロック(LCU)の位置を示す。図7の例に示すように、最左タイル352A中のツリーブロックのそれぞれは、中間タイル352B中のいずれかのツリーブロックの前のツリーブロックコーディング順序で発生する。中間タイル352B中のツリーブロックのそれぞれは、最右タイル352C中のいずれかのツリーブロックの前のツリーブロックコーディング順序で発生する。タイル352A、352B、および352Cのそれぞれの中で、ツリーブロックはラスタスキャン順序に従ってコーディングされる。

【0144】

ビデオエンコーダが、ピクチャー350のための2つのコード化スライスNALユニッ

10

20

30

40

50

トを生成し得る。第1のコード化スライスNALユニットは、ピクチャー350の左スライスに関連付けられ得る。第1のコード化スライスNALユニットは、ツリーブロック1~23の符号化表現を含み得る。第1のコード化スライスNALユニットのスライスデータは、2つのセグメントを含み得る。第1のセグメントは、ツリーブロック1~15の符号化表現を含み得る。第2のセグメントは、ツリーブロック16~30の符号化表現を含み得る。本開示の技法によれば、第2のセグメントがバイト境界において開始するように、第1のセグメントがパディングされ得る。

【0145】

第2のコード化スライスNALユニットは、ピクチャー350の右スライスに関連付けられ得る。第2のコード化スライスNALユニットは、ツリーブロック24~45の符号化表現を含み得る。第2のコード化スライスNALユニットのスライスデータは、2つのセグメントを含み得る。第1のセグメントは、ツリーブロック24~30の符号化表現を含み得る。第2のセグメントは、ツリーブロック31~45の符号化表現を含み得る。第2のセグメントがバイト境界において開始するように、第1のセグメントがパディングされ得る。

【0146】

図8は、例示的なコード化スライスNALユニット400を示す概念図である。図8の例に示すように、コード化スライスNALユニット400は、スライスヘッダ402とスライスデータ404とを含む。スライスデータ404は、第1のセグメント406と第2のセグメント408とを含む。セグメント406は、コード化ツリーブロック410A~410Nとパディングデータ412とを含む。セグメント408は、コード化ツリーブロック414A~414Nを含む。

【0147】

1つまたは複数の例では、説明した機能は、ハードウェア、ソフトウェア、ファームウェア、またはそれらの任意の組み合わせで実装され得る。ソフトウェアで実装される場合、機能は、1つまたは複数の命令またはコードとしてコンピュータ可読媒体上に記憶されるか、あるいはコンピュータ可読媒体を介して送信され、ハードウェアベースの処理ユニットによって実行され得る。コンピュータ可読媒体は、データ記憶媒体などの有形媒体に対応するコンピュータ可読記憶媒体を含み得、または、たとえば、通信プロトコルに従って、ある場所から別の場所へのコンピュータプログラムの転送を可能にする任意の媒体を含む通信媒体を含み得る。このようにして、コンピュータ可読媒体は、概して、(1)非一時的である有形コンピュータ可読記憶媒体、あるいは(2)信号または搬送波などの通信媒体に対応し得る。データ記憶媒体は、本開示で説明した技法の実装のための命令、コードおよび/またはデータ構造を取り出すために1つまたは複数のコンピュータあるいは1つまたは複数のプロセッサによってアクセスされ得る任意の利用可能な媒体であり得る。コンピュータプログラム製品はコンピュータ可読媒体を含み得る。

【0148】

限定ではなく例として、そのようなコンピュータ可読記憶媒体は、RAM、ROM、EEPROM(登録商標)、CD-ROMまたは他の光ディスクストレージ、磁気ディスクストレージ、または他の磁気ストレージデバイス、フラッシュメモリ、あるいは命令またはデータ構造の形態の所望のプログラムコードを記憶するために使用され得、コンピュータによってアクセスされ得る、任意の他の媒体を含むことができる。また、いかなる接続もコンピュータ可読媒体と適切に呼ばれる。たとえば、命令が、同軸ケーブル、光ファイバーケーブル、ツイストペア、デジタル加入者回線(DSL)、または赤外線、無線、およびマイクロ波などのワイヤレス技術を使用して、ウェブサイト、サーバ、または他のリモートソースから送信される場合、同軸ケーブル、光ファイバーケーブル、ツイストペア、DSL、または赤外線、無線、およびマイクロ波などのワイヤレス技術は、媒体の定義に含まれる。ただし、コンピュータ可読記憶媒体およびデータ記憶媒体は、接続、搬送波、信号、または他の一時媒体を含まないが、代わりに非一時的有形記憶媒体を対象とすることを理解されたい。本明細書で使用するディスク(disk)およびディスク(disc)は、

10

20

30

40

50

コンパクトディスク (disc) (CD)、レーザーディスク (登録商標) (disc)、光ディスク (disc)、デジタル多用途ディスク (disc) (DVD)、フロッピー (登録商標) ディスク (disk) および Blu-ray ディスク (disc) を含み、ディスク (disk) は、通常、データを磁氣的に再生し、ディスク (disc) は、データをレーザーで光学的に再生する。上記の組み合わせもコンピュータ可読媒体の範囲内に含まれるべきである。

【0149】

命令は、1つまたは複数のデジタル信号プロセッサ (DSP) などの1つまたは複数のプロセッサ、汎用マイクロプロセッサ、特定用途向け集積回路 (ASIC)、フィールドプログラマブル論理アレイ (FPGA)、あるいは他の等価な集積回路またはディスクリート論理回路によって実行され得る。したがって、本明細書で使用する「プロセッサ」という用語は、前述の構造、または本明細書で説明した技法の実装に好適な他の構造のいずれかを指すことがある。さらに、いくつかの態様では、本明細書で説明した機能は、符号化および復号のために構成された専用のハードウェアおよび/またはソフトウェアモジュール内に与えられ得、あるいは複合コーデックに組み込まれ得る。また、本技法は、1つまたは複数の回路または論理要素中に十分に実装され得る。

【0150】

本開示の技法は、ワイヤレスハンドセット、集積回路 (IC) または IC のセット (たとえば、チップセット) を含む、多種多様なデバイスまたは装置において実装され得る。本開示では、開示する技法を実行するように構成されたデバイスの機能的態様を強調するために様々な構成要素、モジュール、またはユニットについて説明したが、それらの構成要素、モジュール、またはユニットを、必ずしも異なるハードウェアユニットによって実現する必要があるとは限らない。むしろ、上記で説明したように、様々なユニットが、好適なソフトウェアおよび/またはファームウェアとともに、上記で説明した1つまたは複数のプロセッサを含めて、コーデックハードウェアユニットにおいて組み合わせられるか、または相互動作ハードウェアユニットの集合によって与えられ得る。

【0151】

様々な例について説明した。これらおよび他の例は以下の特許請求の範囲内に入る。

以下に、本願出願の当初の特許請求の範囲に記載された発明を付記する。

[C1]

ビデオデータを符号化するための方法であって、前記方法は、

ピクチャーを複数のピクチャーパーティションに分割することと、ここで、前記ピクチャーは複数のツリーブロックを有し、前記ピクチャーパーティションは前記ピクチャーの前記ツリーブロックのオーバーラップしないサブセットに関連付けられる、

前記ピクチャーのスライスに関連付けられた前記ツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ (NAL) ユニットを生成することと、ここで、前記ツリーブロックの前記符号化表現は、前記コード化スライス NAL ユニット内で前記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化され、前記セグメントのそれぞれがバイト境界上で開始するように前記セグメントのうちの1つまたは複数がパディングされる、を含む、方法。

[C2]

前記コード化スライス NAL ユニットを生成することは、前記セグメントのうちの1つまたは複数のためのエントリーポイントを示すスライスヘッダを生成することを含む、C1に記載の方法。

[C3]

前記セグメントのための前記エントリーポイントは、前記セグメントのバイトオフセットを示す、C2に記載の方法。

[C4]

前記ピクチャーパーティションは、タイルまたはウェイブフロント並列処理 (WPP) ウェイブである、C1に記載の方法。

[C 5]

前記ピクチャーに関連付けられたパラメータセットを生成することをさらに含み、前記パラメータセットは、第1の値を有するフラグを含み、前記第1の値は、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの前記1つまたは複数がパディングされることを示し、

前記フラグが第2の値を有するとき、前記セグメントはバイト境界において開始することも開始しないこともある、C 1 に記載の方法。

[C 6]

前記セグメントは、所与のセグメントを含み、

前記コード化スライスNALユニットを生成することは、次のツリーブロックが、前記スライス内にあり、前記所与のセグメントとは異なるピクチャーパーティションに関連付けられる場合に、前記所与のセグメントにビットを付加するパディング演算を実行することを含む、C 1 に記載の方法。

[C 7]

前記ピクチャーパーティションは、タイルであり、

前記コード化スライスNALユニットを生成することは、前記タイルが互いから独立していると判定した後により、前記セグメントがバイト境界において開始することを保証するパディング演算を実行することを含む、C 1 に記載の方法。

[C 8]

ビデオデータを復号する方法であって、前記方法は、

ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ(NAL)ユニットを記憶することと、ここで、前記ピクチャーは複数のピクチャーパーティションに区分され、前記ツリーブロックの前記符号化表現は前記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化され、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの1つまたは複数がパディングされる、

前記ツリーブロックの前記符号化表現を復号することを含む、方法。

[C 9]

前記コード化スライスNALユニットは、前記セグメントのうちの1つまたは複数のためのエン트리ポイントを示すスライスヘッダを含む、C 8 に記載の方法。

[C 10]

前記セグメントのための前記エン트리ポイントは、前記セグメントのバイトオフセットを示す、C 9 に記載の方法。

[C 11]

前記ピクチャーパーティションは、タイルまたはウェーブフロント並列処理(WPP)ウェーブである、C 8 に記載の方法。

[C 12]

前記ピクチャーに関連付けられたパラメータセットを記憶することをさらに含み、前記パラメータセットは、第1の値を有するフラグを含み、前記第1の値は、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの前記1つまたは複数がパディングされることを示し、

前記フラグが第2の値を有するとき、前記セグメントはバイト境界において開始することも開始しないこともある、C 8 に記載の方法。

[C 13]

前記ピクチャーパーティションは、タイルであり、

前記タイルが互いから独立しているとき、前記セグメントがバイト境界において開始するように前記セグメントのうちの1つまたは複数がパディングされる、C 8 に記載の方法。

[C 14]

10

20

30

40

50

前記ツリーブロックの前記符号化表現を復号することは、前記セグメントのうちの2つ以上の中の前記ツリーブロックの前記符号化表現を並列に復号することを含む、C 8に記載の方法。

[C 1 5]

ビデオデータを符号化するビデオ符号化デバイスであって、前記ビデオ符号化デバイスは、

ピクチャーを複数のピクチャーパーティションに分割することと、ここで、前記ピクチャーは複数のツリーブロックを有し、前記ピクチャーパーティションは前記ピクチャーの前記ツリーブロックのオーバーラップしないサブセットに関連付けられる、

前記ピクチャーのスライスに関連付けられた前記ツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ (NAL) ユニットを生成することと、ここで、前記ツリーブロックの前記符号化表現は、前記コード化スライスNALユニット内で前記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化され、前記セグメントのそれぞれがバイト境界上で開始するように前記セグメントのうちの1つまたは複数のパディングされる、を行うように構成された1つまたは複数のプロセッサを含む、ビデオ符号化デバイス。

[C 1 6]

前記1つまたは複数のプロセッサは、前記コード化スライスNALユニットが、前記セグメントのうちの1つまたは複数のためのエントリーポイントを示すスライスヘッダを含むように、前記コード化スライスNALユニットを生成するように構成された、C 1 5に記載のビデオ符号化デバイス。

[C 1 7]

前記セグメントのための前記エントリーポイントは、前記セグメントのバイトオフセットを示す、C 1 6に記載のビデオ符号化デバイス。

[C 1 8]

前記ピクチャーパーティションは、タイルまたはウェーブフロント並列処理 (WPP) ウェーブである、C 1 5に記載のビデオ符号化デバイス。

[C 1 9]

前記1つまたは複数のプロセッサは、前記ピクチャーに関連付けられたパラメータセットを生成するように構成され、前記パラメータセットは、第1の値を有するフラグを含み、前記第1の値は、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの前記1つまたは複数のパディングされることを示し、

前記フラグが第2の値を有するとき、前記セグメントはバイト境界において開始することも開始しないこともある、C 1 5に記載のビデオ符号化デバイス。

[C 2 0]

前記セグメントは、所与のセグメントを含み、

前記1つまたは複数のプロセッサは、次のツリーブロックが、前記スライス内にあり、前記所与のセグメントとは異なるピクチャーパーティションに関連付けられる場合に、前記所与のセグメントにビットを付加するパディング演算を実行するように構成された、C 1 5に記載のビデオ符号化デバイス。

[C 2 1]

前記ピクチャーパーティションは、タイルであり、

1つまたは複数のプロセッサは、前記タイルが互いから独立していると判定した後にのみ、前記セグメントがバイト境界において開始することを保証するパディング演算を実行するように構成された、C 1 5に記載のビデオ符号化デバイス。

[C 2 2]

前記ビデオ符号化デバイスは、モバイルコンピューティングデバイスである、C 1 5に記載のビデオ符号化デバイス。

[C 2 3]

ビデオデータを復号するビデオ復号デバイスであって、前記ビデオ復号デバイスは、

10

20

30

40

50

ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ (NAL) ユニットを記憶するメモリと、ここで、前記ピクチャーは複数のピクチャーパーティションに分割され、前記ツリーブロックの前記符号化表現は前記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化され、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの1つまたは複数がパディングされる、

前記ツリーブロックの前記符号化表現を復号するように構成された1つまたは複数のプロセッサとを含む、ビデオ復号デバイス。

[C 2 4]

前記コード化スライスNALユニットは、前記セグメントのうちの1つまたは複数のためのエントリーポイントを示すスライスヘッダを含む、C 2 3に記載のビデオ復号デバイス。

[C 2 5]

前記セグメントのための前記エントリーポイントは、前記セグメントのバイトオフセットを示す、C 2 4に記載のビデオ復号デバイス。

[C 2 6]

前記ピクチャーパーティションは、タイルまたはウェイブフロント並列処理 (WPP) ウェイブである、C 2 3に記載のビデオ復号デバイス。

[C 2 7]

前記メモリは、前記ピクチャーに関連付けられたパラメータセットを記憶し、前記パラメータセットは、第1の値を有するフラグを含み、前記第1の値は、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの前記1つまたは複数
がパディングされることを示し、

前記フラグが第2の値を有するとき、前記セグメントはバイト境界において開始することも開始しないこともある、C 2 3に記載のビデオ復号デバイス。

[C 2 8]

前記ビデオ復号デバイスは、モバイルコンピューティングデバイスである、C 2 3に記載のビデオ復号デバイス。

[C 2 9]

前記1つまたは複数のプロセッサが、前記セグメントのうちの2つ以上の中の前記ツリーブロックの前記符号化表現を並列に復号するように構成された、C 2 3に記載のビデオ復号デバイス。

[C 3 0]

1つまたは複数のプロセッサによって実行されたとき、

ピクチャーを複数のピクチャーパーティションに分割することと、ここで、前記ピクチャーは複数のツリーブロックを有し、前記ピクチャーパーティションは前記ピクチャーの前記ツリーブロックのオーバーラップしないサブセットに関連付けられる、

前記ピクチャーのスライスに関連付けられた前記ツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ (NAL) ユニットを生成することと、ここで、前記ツリーブロックの前記符号化表現は、前記コード化スライスNALユニット内で前記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化され、前記セグメントのそれぞれがバイト境界上で開始するように前記セグメントのうちの1つまたは複数
がパディングされる、を行うようにビデオ符号化デバイスを構成する、命令を記憶する1つまたは複数のコンピュータ可読記憶媒体を含む、コンピュータプログラム製品。

[C 3 1]

1つまたは複数のプロセッサによって実行されたとき、

ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ (NAL) ユニットを記憶することと、こ

10

20

30

40

50

ここで、前記ピクチャーは複数のピクチャーパーティションに分割され、前記ツリーブロックの前記符号化表現は前記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化され、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの1つまたは複数がパディングされる、

前記ツリーブロックの前記符号化表現を復号することとを行うようにビデオ復号デバイスを構成する、命令を記憶する1つまたは複数のコンピュータ可読記憶媒体を含む、コンピュータプログラム製品。

[C 3 2]

ビデオデータを符号化するビデオ符号化デバイスであって、前記ビデオ符号化デバイスは、

ピクチャーを複数のピクチャーパーティションに分割するための手段と、ここで、前記ピクチャーは複数のツリーブロックを有し、前記ピクチャーパーティションは前記ピクチャーの前記ツリーブロックのオーバーラップしないサブセットに関連付けられる、

前記ピクチャーのスライスに関連付けられた前記ツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ (NAL) ユニットを生成するための手段と、ここで、前記ツリーブロックの前記符号化表現は、前記コード化スライスNALユニット内で前記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化され、前記セグメントのそれぞれがバイト境界上で開始するように前記セグメントのうちの1つまたは複数がパディングされる、を含む、ビデオ符号化デバイス。

[C 3 3]

ビデオデータを復号するビデオ復号デバイスであって、前記ビデオ復号デバイスは、

ピクチャーのスライスに関連付けられたツリーブロックの符号化表現を含むコード化スライスネットワークアブストラクションレイヤ (NAL) ユニットを記憶するための手段と、ここで、前記ピクチャーは複数のピクチャーパーティションに分割され、前記ツリーブロックの前記符号化表現は前記ピクチャーパーティションのうちの異なるピクチャーパーティションに関連付けられたセグメントにグループ化され、前記セグメントのそれぞれがバイト境界において開始するように前記セグメントのうちの1つまたは複数がパディングされる、

前記ツリーブロックの前記符号化表現を復号するための手段とを含む、ビデオ復号デバイス。

10

20

30

【図 1】

図 1

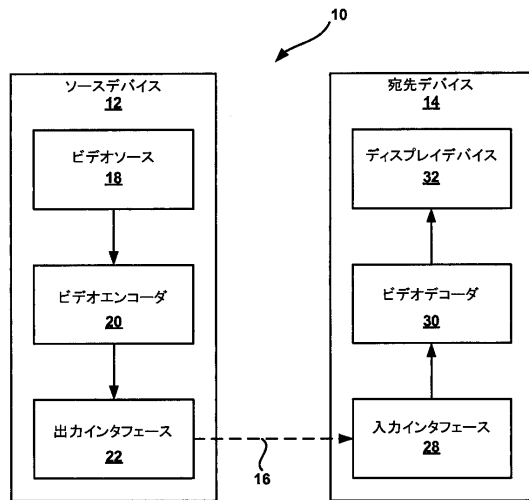


FIG. 1

【図 2】

図 2

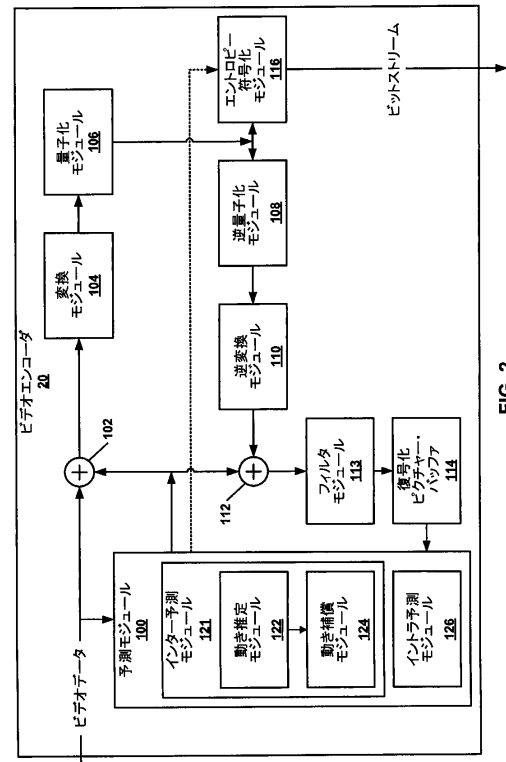


FIG. 2

【図 3】

図 3

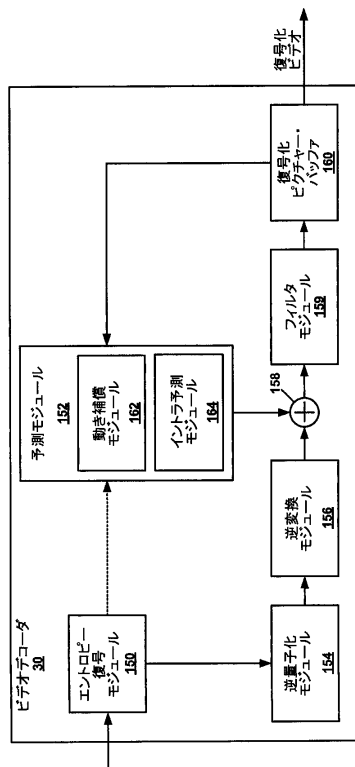


FIG. 3

【図 4】

図 4

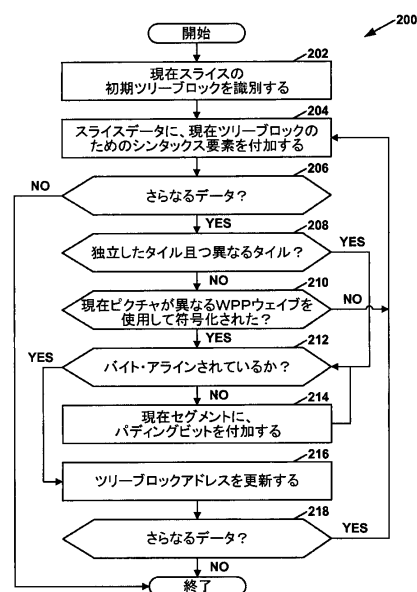


FIG. 4

【図 5】

図 5

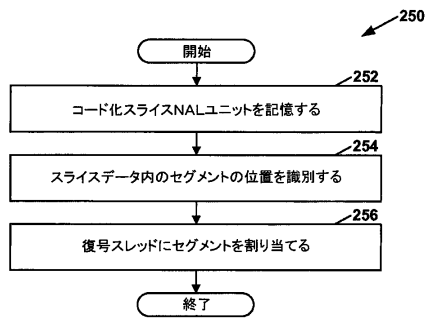


FIG. 5

【図 6】

図 6

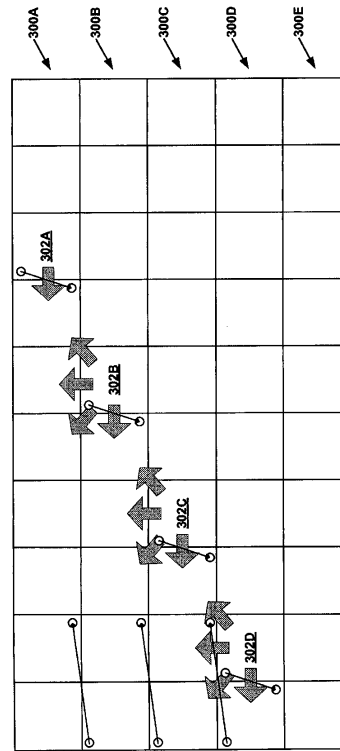


FIG. 6

【図 7】

図 7

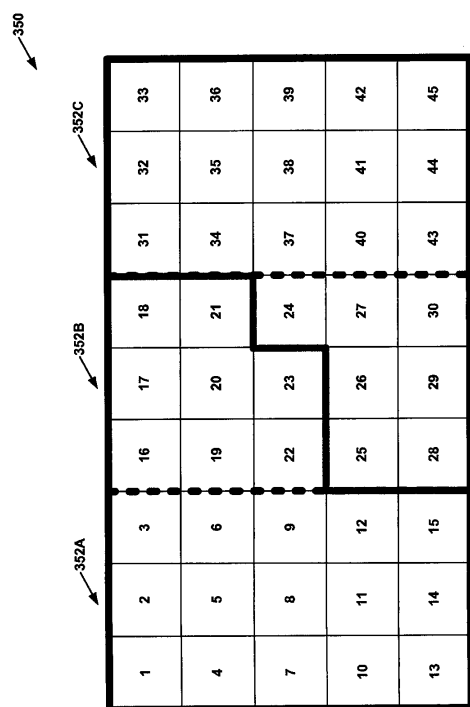


FIG. 7

【図 8】

図 8

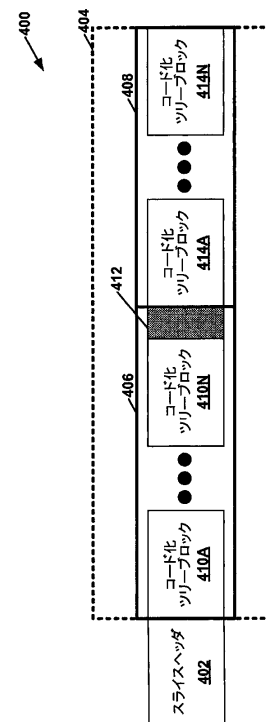


FIG. 8

フロントページの続き

- (31)優先権主張番号 13/548,825
(32)優先日 平成24年7月13日(2012.7.13)
(33)優先権主張国 米国(US)

早期審査対象出願

- (74)代理人 100153051
弁理士 河野 直樹
(74)代理人 100140176
弁理士 砂川 克
(74)代理人 100158805
弁理士 井関 守三
(74)代理人 100179062
弁理士 井上 正
(74)代理人 100124394
弁理士 佐藤 立志
(74)代理人 100112807
弁理士 岡田 貴志
(74)代理人 100111073
弁理士 堀内 美保子
(72)発明者 ワン、イエ クイ
アメリカ合衆国、カリフォルニア州 9 2 1 2 1、サン・ディエゴ、モアハウス・ドライブ 5 7
7 5
(72)発明者 コバン、ムハンメド・ゼイド
アメリカ合衆国、カリフォルニア州 9 2 1 2 1、サン・ディエゴ、モアハウス・ドライブ 5 7
7 5
(72)発明者 カークゼウィックス、マルタ
アメリカ合衆国、カリフォルニア州 9 2 1 2 1、サン・ディエゴ、モアハウス・ドライブ 5 7
7 5

審査官 堀井 啓明

- (56)参考文献 米国特許出願公開第2 0 1 0 / 0 0 6 1 4 5 5 (US, A1)
国際公開第2 0 1 0 / 0 5 0 1 5 7 (WO, A1)
米国特許出願公開第2 0 1 1 / 0 2 0 6 1 2 3 (US, A1)
K.Misra,A.SegalI, Tiles, Joint Collaborative Team on Video Coding(JCT-VC) of ITU-T SG1
6 WP3 and ISO/IEC JTC1/SC29/WG11 5th Meeting:Geneva,CH,16-23 March,2011, 米国, JCTVC,
2 0 1 1 年 3 月 1 2 日, JCTVC-E412, P.1-P.4, URL, [http://phenix.it-sudparis.eu/jct/
index.php](http://phenix.it-sudparis.eu/jct/index.php)
Arild Fuldseth,Michael Horowitz,et.al., Tiles, Joint Collaborative Team on Video Codin
g(JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 5th Meeting:Geneva,CH,16-23 Mar
ch,2011, 米国, JCTVC, 2 0 1 1 年 3 月 1 1 日, JCTVC-E408, P.1-P.14, URL, [http://phe
nix.it-sudparis.eu/jct/index.php](http://phe
nix.it-sudparis.eu/jct/index.php)

- (58)調査した分野(Int.Cl., DB名)
H 0 4 N 1 9 / 0 0 - 1 9 / 9 8