

(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 1
HO4N 19/70 (2014.01)	HO4N 19/70
HO4N 19/119 (2014.01)	HO4N 19/119
HO4N 19/174 (2014.01)	HO4N 19/174
HO4N 19/96 (2014.01)	HO4N 19/96
HO4N 19/46 (2014.01)	HO4N 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) 國際出願番号	PCT/US2012/063027
(87) 國際公開番号	W02013/067158
(87) 國際公開日	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 92121-1714、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つて、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プロックを有し、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プ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プ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プロ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、前記ツリープ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プロックが属する前記ピクチャーパーティションに従つて、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プロックが、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、方法。

【請求項 2】

10

20

前記コード化スライスN A L 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、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、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、タイルまたはウェイブフロント並列処理(W P P)ウェイブである、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N A L 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(N A L)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N A L 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N A L 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、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、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、タイルまたはウェイブフロント並列処理(W P P)ウェイブである、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、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つまたは複数がパディングされる、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。 10

【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。 20

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、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、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、タイルまたはウェイブフロント並列処理(WPP)ウェイブである、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つまたは複数がパディングされることを示し、

10

20

30

40

50

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、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、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。

10

【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ストラクションレイヤ（N A L）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つまたは複数がパディングされ、

20

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N A L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N A L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、請求項21に記載のビデオ復号デバイス。

30

【請求項23】

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、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、タイルまたはウェイブフロント並列処理（W P P）ウェイブである、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つまたは複数がパディングされることを示し、

40

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、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

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プロックを有し、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プ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プ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、前記ツリープ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プ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 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プロックが、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プ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、前記ツリープロックの前記符号化表現は前記ツリープ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プロックが、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、

前記ツリープ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。 40

【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、

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プロックを有し、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プ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プ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、前記ツリープ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プ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N A L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 3 1】

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(N A L)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N A L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】

【0 0 0 1】

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

【0 0 0 2】

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】

【0 0 0 3】

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(P D A)、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M P E G - 2、M P E G - 4、I T U - T H . 2 6 3、I T U - T H . 2 6 4 / M P E G - 4，P a r t 1 0，A d v a n c e d V i d e o

40

C o d i n g (A V C) 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(H E V C : 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。

【0 0 0 4】

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(C U)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フォーム(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(tile)s)およびウェイブフロント並列処理(W P P)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アブストラクションレイヤ(N A L)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N A L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N A L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N A L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N A L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N A L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N A L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。¹⁰ 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。

【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。²⁰ 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。

【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。³⁰ 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。

【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。

【0013】

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ろう。10

【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。20

【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の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。40

【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50

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、連續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、その後に続く第 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、その後に続く第 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。

【 0 0 1 9 】

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 WPPP ウェイプが 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 WPPP ウェイプのバイト・アラインメント (byte alignment) を提供し得る。

【 0 0 2 0 】

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。

【 0 0 2 1 】

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。

【 0 0 2 2 】

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。

【 0 0 2 3 】

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。

【 0 0 2 4 】

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。

10

【0026】

宛先デバイス 14 は、チャネル 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。

20

【0027】

別の例では、チャネル 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せであり得る。

30

【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。

40

50

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せを含み得る。

【0030】

20

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。

【0031】

30

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。

【0032】

40

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。

【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「WD 6」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、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つて、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。20
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マルチブレクサプロトコル、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。

【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。40
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。

【0037】

上記で手短に述べたように、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 represen50

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。

【 0 0 3 8 】

ビットストリームを生成するために、ビデオエンコーダ 2 0 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 2 0 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 2 0 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 (S P S : 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 . 2 6 4 / A V C および H E V C 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 (P P S : 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。A P S 内のパラメータは、P P S 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。10

【 0 0 3 9 】

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 2 0 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 (L C U : 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。H E V C のツリーブロックは、H . 2 6 4 / A V C 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 (C U : coding unit) を含み得る。ビデオエンコーダ 2 0 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、C U 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」 (treeblocks) という名前がある。20

【 0 0 4 0 】

いくつかの例では、ビデオエンコーダ 2 0 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 C U 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 2 0 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 2 0 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 2 0 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。30

【 0 0 4 1 】

コード化スライスを生成するために、ビデオエンコーダ 2 0 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 2 0 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 2 0 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。40

【 0 0 4 2 】

ビデオエンコーダ 2 0 がコード化スライスを生成するとき、ビデオエンコーダ 2 0 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 2 0 は、スライス中のツリーブロックの一一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】

ラスタ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。10

【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)と呼ばれることがある。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。20

【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。30

【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。40

【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 parameter50

) 値に関連付け得る。CUに関連付けられたQP値は、ビデオエンコーダ20が、CU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、CUに関連付けられたQP値を調整することによって、CU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(CABAC)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プロストラクションレイヤ(NAL)ユニットを含み得る。NALユニットのそれぞれは、NAL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、NAL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マント情報(SEI: 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。NAL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CU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、CUのPU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、CUのTU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、CUのTU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、CU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、CU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プ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プフロント並列処理(WPP)ウェイプに分割し得る。本開示では、タイルとWPPウェイプの両方を総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プロックを含み得る。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などのビデオコーダは、ラスタスキヤン順序に従って、現在ピクチャーのタイルをコーディングし得る。さらに、ビデオコーダがタイルをコーディングするとき、ビデオコーダは、ラスタスキヤン順序に従ってタイル内の各ツリープロックをコーディングし得る。このようにして、ビデオコーダは、現在ピクチャーの別のタイルのいずれかのツリープ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プロックをコーディングし得る。したがって、ビデオコーダが現在ピクチャーを複数のタイルに区分するとき、ビデオコーダが現在ピクチャーのツリープロックをコーディングする順序は、ビデオコーダが現在ピクチャーを複数のタイルに区分しないときとは異なり得る。20

【0063】

さらに、いくつかの事例では、ビデオコーダは、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、ビデオコーダは、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CABA C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、ビデオコーダは、複数のタイルのツリープ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、ビデオコーダは、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。ビデオコーダがWPPを使用して現在ピクチャーをコーディングするとき、ビデオコーダは、現在ピクチャーのツリープロックを複数の「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プロックの異なる行に対応し得る。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するとき、ビデオコーダは、ツリープロックの最上行をコーディングし始め得る。ビデオコーダが最上行の2つ以上のツリープロックをコーディングしたとき、ビデオコーダは、ツリープ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プ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。ビデオコーダが最上行から2番目の行の2つ以上のツリープロックをコーディングしたとき、ビデオコーダは、ツリープ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プ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プロックの行を下って継ぎ得る。40

【0065】

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、ビデオコーダは、空間的に隣接するCUが現在ツリープロックの左、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 C U 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 C U 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 C A B A C 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 C A B A C 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 C A B A C 状態に基づいて、ある行の C A B A C 状態を初期化し得る。10

【 0 0 6 6 】

いくつかの例では、ビデオコーダが W P P 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 W P P ウェイプに分割し得る。

【 0 0 6 7 】

上述のように、ビデオエンコーダ 2 0 は、スライスの符号化表現を含むコード化スライス 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 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、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 2 0 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 N A L ユニットを生成し得る。たとえば、コード化スライス N A L 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、その後に続く第 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、その後に続く第 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

【 0 0 6 8 】

本開示の技法によれば、ビデオエンコーダ 2 0 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 N A L 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 N A L 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 N A L 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 2 0 がセグメントをパディングするとき、ビデオエンコーダ 2 0 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 2 0 は、剩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。40

【 0 0 6 9 】

ビデオデコーダ 3 0 がコード化スライス N A L ユニットを受信するとき、ビデオエンコーダ 3 0 は、コード化スライス N A L 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 3 0 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 3 0 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 3 0 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 3 0 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つまたは複数がパディングされる。

【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つまたは複数がパディングされる。ビデオデコーダ30は、ツリーブロックの符号化表現を復号し得る。いくつかの事例では、ビデオデコーダ30は、セグメントのうちの2つ以上の中のツリーブロックの符号化表現を並列に復号し得る。

【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。

【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。

【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

10

20

30

40

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に依存し得る。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 】

10

20

30

40

50

CU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、CUの1つまたは複数のPUの中で、CU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PUサイズをサポートし得る。特定のCU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PU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PU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PU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、CU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CUの複数のPUの間で、CU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。

【0081】

インター予測モジュール121は、CUの各PU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。PU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、PU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、CU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、PU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。

【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、PU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、CUのPU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PUがイントラ予測される。したがって、PU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、PUに対してインター予測を実行しない。

【0083】

PUがP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)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PU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、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る。PUの参照ブロックは、PU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PU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PU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。

【0084】

Pスライス中のPU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、PU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、PUの動き情報として参照インデックスと動きベク

10

20

30

40

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の組み合せである、リストの組み合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トロ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、より少數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 P U に対してイントラ予測を実行し得る。

【0090】

P U に対してイントラ予測を実行するためには、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、P U 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 P U 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 P U のビデオブロックから P U 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 P U は、P U、C U およびツリープ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、P U 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、P U 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、P U 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 P U 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、P U 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、P U の予測データを選択する。20

【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、P U 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 P U 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 P U のイントラ予測モードは、現在 P U 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 P U 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。30

【0093】

予測モジュール 100 が C U の P U の予測データを選択した後、残差生成モジュール 102 は、C U のビデオブロックから C U の P U の予測されたビデオブロックを差し引くことによって、C U の残差データを生成し得る。C U の残差データは、C U 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、C U の P U 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、C U 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、C U の残差データは、C U の P U 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、C U 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。40

【0094】

予測モジュール 100 は、4 分木区分を実行して、C U 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、C U の異なる T U に関連付けられ得る。C U の T U 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、C U の P U 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。C U の T U 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、T U 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、最小のピ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に関連付けさ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、復号化ピクチャーバッファ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ーを使用して、後続のピクチャーの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、復号化ピクチャーバッファ114中の再構成されたビデオブロックを使用して、CUと同じピクチャーの中の他の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、シンタックスベ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。

【 0101 】

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ビ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、「ビン」(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。

【 0102 】

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 NAL 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 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。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 NAL 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 NAL ユニットを生成し得る。

【 0103 】

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 NAL ユニットのスライスヘッダがコード化スライス NAL 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トリーポイントを示すようにスライスヘッダを生成し得る。エントリーポイントは、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、エントリーポイントは、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 NAL 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 NAL 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、エントリーポイントは、セグメントのそれ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トリーポイントを示さない。

【 0104 】

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。

【 0105 】

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

10

20

30

40

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にのみ、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

【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】

40

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。

【 0 1 1 1 】

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 ウェイブ）に関連付けられたコード化ツリープロックを含み得る。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ー・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プロックをパースし、対応するツリープ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。
10

【 0 1 1 2 】

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。
20

【 0 1 1 3 】

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。
30 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。

【 0 1 1 4 】

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。
40

【 0 1 1 5 】

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
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。

【0116】

いくつかの例では、逆変換モジュール156は、ビデオエンコーダ20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る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156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156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156は、カスケード逆変換(cascaded inverse transform)を適用し得る。

10

【0117】

いくつかの例では、動き補償モジュール162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162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20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162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20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

【0118】

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164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164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164が使用し得るシンタックス要素を含み得る。

【0119】

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164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るべきであることを、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164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

【0120】

再構成モジュール158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、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

【0121】

再構成モジュール158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159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159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30は、復

50

号化ピクチャーバッファ 160 に C U のビデオブロックを記憶し得る。復号化ピクチャーバッファ 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、参照ピクチャーを与え得る。たとえば、ビデオエンコーダ 30 は、復号化ピクチャーバッファ 160 中のビデオブロックに基づいて、他の C U の P U 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、現在ピクチャー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。 20

【 0124 】

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。 30

【 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。 40

【 0127 】

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、現在ピクチャー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、ピクチャーのタイルは、ピクチャー内予測 (たとえば、現在ピクチャー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ー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、現在ピクチャー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、現在ピクチャー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。10

【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はない場合(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ウェイプ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ウェイプ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ウェイプ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。20

【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ウェイプ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ウェイプ)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。3040

【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対応して(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はないと判定したことに対応して(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。

【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対応して(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。

【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つまたは複数がパディングされ得る。

【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。

【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、ウェイプ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

10

20

30

40

50

チャーは、ツリーブロックにそ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）と総称することができる。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N A L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N A L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N A L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N A L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N A L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N A L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N A L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N A L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、R A M、R O M、E E P R O M(登録商標)、C D - R O M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(D S L)、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、D S L、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) (C D)、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) (D V D)、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、データを磁気的に再生し、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 (D S P) 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 (A S I C)、フィールドプログラマブル論理アレイ (F P G A)、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。

10

【0150】

本開示の技法は、ワイヤレスハンドセット、集積回路 (I C) またはI C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。

20

【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。

[C 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、

30

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プロックを有し、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プ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プ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ストラクションレイヤ (N A L) ユニットを生成することと、ここで、前記ツリープロックの前記符号化表現は、前記コード化スライスN A L 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。

40

[C 2]

前記コード化スライスN A L 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、C 1に記載の方法。

[C 3]

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に記載の方法。

[C 4]

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 (W P P) ウェイブである、C 1に記載の方法。

50

[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N A L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。

10

[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N A L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、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、

20

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(N A L)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N A L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 8に記載の方法。

30

[C 10]

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 9に記載の方法。

[C 11]

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

40

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]

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 15]

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 (N A L) 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 N A L 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 16]

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 N A L 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 N A L ユニットを生成するように構成された、C 15 に記載のビデオ符号化デバイス。

10

[C 17]

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 16 に記載のビデオ符号化デバイス。

[C 18]

前記ピクチャーパーティションは、タイルまたはウェイプフロント並列処理 (W P P) ウェイプである、C 15 に記載のビデオ符号化デバイス。

[C 19]

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 つまたは複数がパディングされることを示し、

30

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 15 に記載のビデオ符号化デバイス。

[C 20]

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 15 に記載のビデオ符号化デバイス。

40

[C 21]

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 15 に記載のビデオ符号化デバイス。

[C 22]

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 15 に記載のビデオ符号化デバイス。

[C 23]

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、

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 24]

10

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 23に記載のビデオ復号デバイス。

[C 25]

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 24に記載のビデオ復号デバイス。

[C 26]

前記ピクチャーパーティションは、タイルまたはウェイプフロント並列処理（WPP）ウェイプである、C 23に記載のビデオ復号デバイス。

[C 27]

20

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 23に記載のビデオ復号デバイス。

[C 28]

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 23に記載のビデオ復号デバイス。

[C 29]

30

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 23に記載のビデオ復号デバイス。

[C 30]

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 31]

40

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

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を行な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、

10

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。

20

[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。

30

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。

【図1】

図1

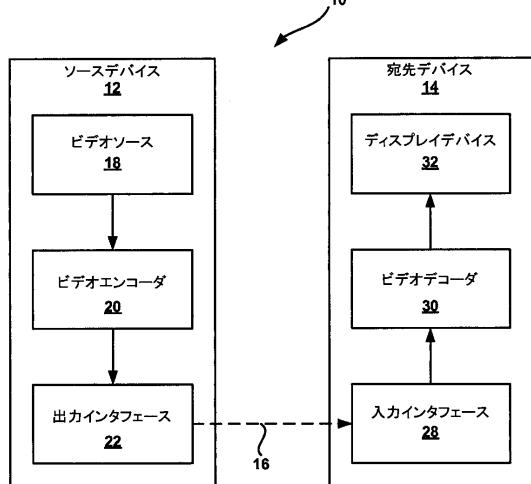


FIG. 1

【図2】

図2

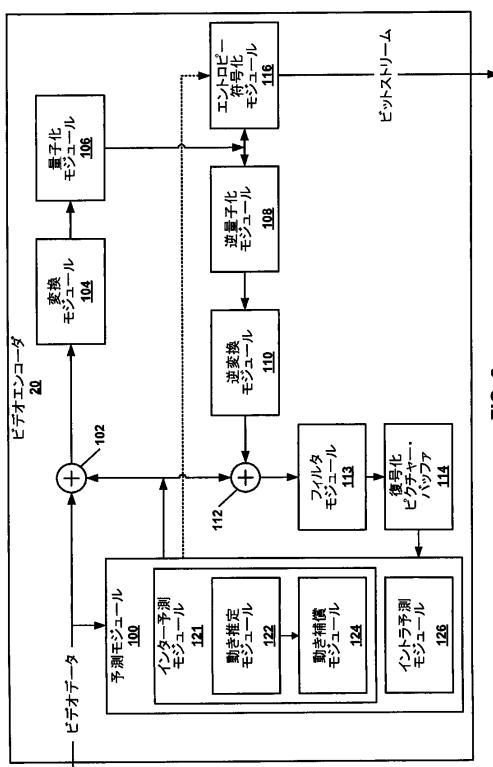


FIG. 2

【図3】

図3

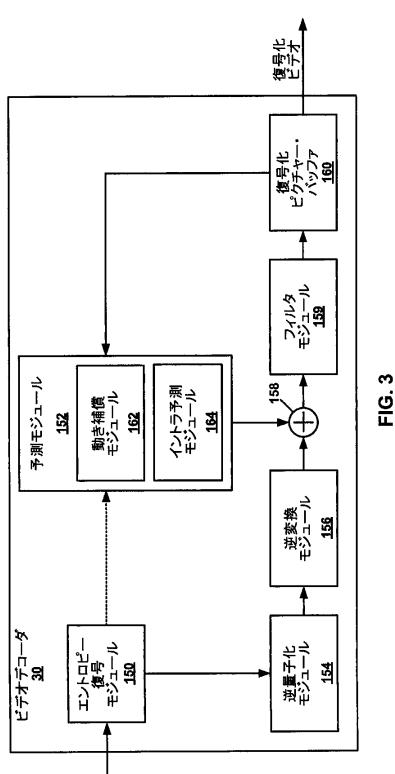


FIG. 3

【図4】

図4

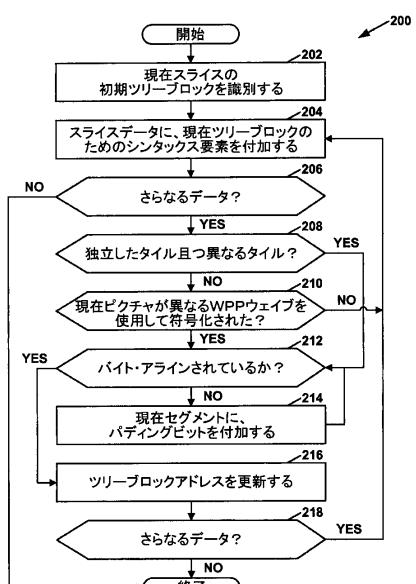


FIG. 4

【図5】

図5

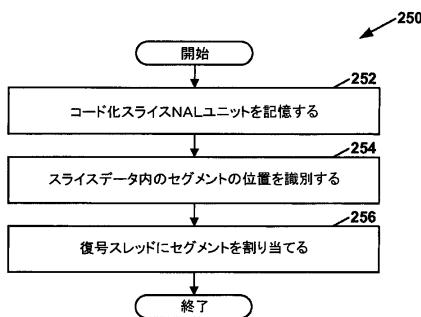


FIG. 5

【図6】

図6

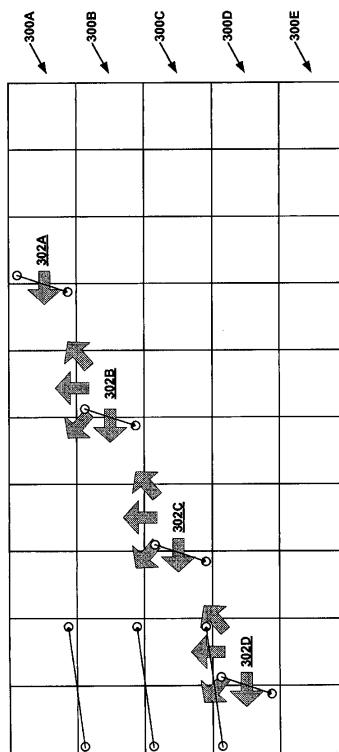


FIG. 6

【図7】

図7

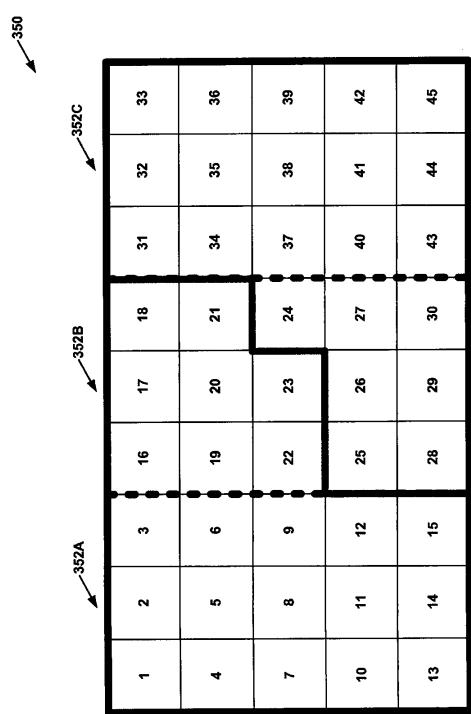


FIG. 7

【図8】

図8

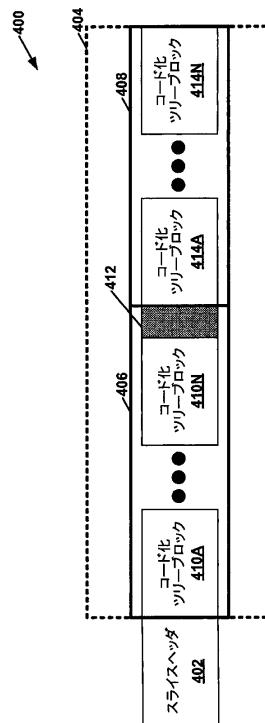


FIG. 8

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 92121、サン・ディエゴ、モアハウス・ドライブ 57
75
(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 92121、サン・ディエゴ、モアハウス・ドライブ 57
75
(72)発明者 カークゼウイックズ、マルタ
アメリカ合衆国、カリフォルニア州 92121、サン・ディエゴ、モアハウス・ドライブ 57
75

審査官 堀井 啓明

(56)参考文献 米国特許出願公開第2010/0061455(US,A1)
国際公開第2010/050157(WO,A1)
米国特許出願公開第2011/0206123(US,A1)
K.Misra,A.Segall,Tiles,Joint Collaborative Team on Video Coding(JCT-VC) of ITU-T SG16
WP3 and ISO/IEC JTC1/SC29/WG11 5th Meeting:Geneva,CH,16-23 March,2011,米国,JCTVC,
2011年 3月12日,JCTVC-E412,P.1-P.4,URL,<http://phenix.it-sudparis.eu/jct/index.php>
Arild Fuldseth,Michael Horowitz,et.al.,Tiles,Joint Collaborative Team on Video Coding(JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 5th Meeting:Geneva,CH,16-23 March,2011,米国,JCTVC,2011年 3月11日,JCTVC-E408,P.1-P.14,URL,<http://phenix.it-sudparis.eu/jct/index.php>

(58)調査した分野(Int.Cl., DB名)

H04N19/00 - 19/98