

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号

特許第7317703号

(P7317703)

(45)発行日 令和5年7月31日(2023.7.31)

(24)登録日 令和5年7月21日(2023.7.21)

(51)国際特許分類

F I

H 0 4 N	19/117 (2014.01)	H 0 4 N	19/117
H 0 4 N	19/159 (2014.01)	H 0 4 N	19/159
H 0 4 N	19/176 (2014.01)	H 0 4 N	19/176
H 0 4 N	19/46 (2014.01)	H 0 4 N	19/46
H 0 4 N	19/59 (2014.01)	H 0 4 N	19/59

請求項の数 20 (全50頁)

(21)出願番号	特願2019-518070(P2019-518070)
(86)(22)出願日	平成29年10月5日(2017.10.5)
(65)公表番号	特表2019-535198(P2019-535198 A)
(43)公表日	令和1年12月5日(2019.12.5)
(86)国際出願番号	PCT/US2017/055390
(87)国際公開番号	WO2018/067848
(87)国際公開日	平成30年4月12日(2018.4.12)
審査請求日	令和2年9月7日(2020.9.7)
(31)優先権主張番号	62/404,749
(32)優先日	平成28年10月5日(2016.10.5)
(33)優先権主張国・地域又は機関	米国(US)
(31)優先権主張番号	15/724,155
(32)優先日	平成29年10月3日(2017.10.3)

最終頁に続く

(73)特許権者	595020643 クゥアルコム・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 9 2121-1714、サン・ディエゴ、 モアハウス・ドライブ 5775
(74)代理人	110003708 弁理士法人鈴榮特許総合事務所
(72)発明者	リ、ソンウォン アメリカ合衆国、カリフォルニア州 9 2121-1714、サン・ディエゴ、 モアハウス・ドライブ 5775
(72)発明者	チェン、ウェイ - ジュン アメリカ合衆国、カリフォルニア州 9

最終頁に続く

(54)【発明の名称】 補間フィルタを切り替えるシステムおよび方法

(57)【特許請求の範囲】

【請求項1】

ビデオデータを符号化する方法であって、
前記ビデオデータを取得することと、
コーディングユニットについて、補間フィルタのセットから補間フィルタのサブセット
を決定することと、前記補間フィルタのサブセットは複数の補間フィルタを含み、ここ
において、前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられ
た、前記ビデオデータ中の情報に基づいて決定され、
前記コーディングユニットを符号化することと、ここにおいて、前記コーディングユニ
ットを符号化することは、前記コーディングユニットについての動き推定および動き補償
のために補間フィルタを選択することを含み、ここにおいて、前記補間フィルタは、前記
補間フィルタのサブセットから以前にコーディングされたブロックのために決定された補
間フィルタから選択され、

符号化されたビデオビットストリームを生成することと、ここにおいて、前記符号化さ
れたビデオビットストリームは、前記符号化されたコーディングユニットを含む、
を備える方法。

【請求項2】

ビデオデータを復号するための方法であって、
符号化されたビデオビットストリームを取得することと、
前記符号化されたビデオビットストリームから現在のフレーム中のコーディングユニッ

10

20

トのための参照フレームを決定することと、

補間フィルタのセットから補間フィルタのサブセットを決定することと、前記補間フィルタのサブセットは複数の補間フィルタを含み、ここにおいて、前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた、前記符号化されたビデオビットストリーム中の情報に基づいて決定され、

前記補間フィルタのサブセットから以前にコーディングされたブロックのために決定された補間フィルタから補間フィルタを決定することと、

前記コーディングユニットを再構成するために、前記参照フレームおよび前記補間フィルタを使用することと

を備える方法。

10

【請求項 3】

前記補間フィルタは、前記コーディングユニットの隣接ブロックのために決定された補間フィルタから選択される、請求項 1 に記載の方法。

【請求項 4】

前記補間フィルタを第 2 の補間フィルタとマージすること、ここにおいて、結果として得られるマージされた補間フィルタが、前記動き推定および前記動き補償のために使用される、

をさらに備える、請求項 1 に記載の方法。

【請求項 5】

前記補間フィルタは、第 2 の補間フィルタに関連付けられ、前記補間フィルタは、整数ピクセル位置のためのものであり、前記第 2 の補間フィルタは、分数ピクセル位置のためのものである、請求項 1 に記載の方法。

20

【請求項 6】

前記補間フィルタと前記第 2 の補間フィルタは、異なるカットオフ周波数を有する、請求項 5 に記載の方法。

【請求項 7】

前記符号化されたビデオビットストリームは、前記補間フィルタを識別する値を含む、請求項 1 に記載の方法。

【請求項 8】

前記補間フィルタのセットは、前記符号化されたビデオビットストリームとともにシグナリングされず、前記補間フィルタの識別は、前記符号化されたビデオビットストリームから暗示され得る、請求項 1 に記載の方法。

30

【請求項 9】

前記符号化されたビデオビットストリームを使用して、前記補間フィルタの識別を導出すること

をさらに備える、請求項 2 に記載の方法。

【請求項 10】

装置であって、

ビデオデータを記憶するように構成されたメモリと、

前記ビデオデータを取得することと、

40

コーディングユニットについて、補間フィルタのセットから補間フィルタのサブセットを決定することと、前記補間フィルタのサブセットは複数の補間フィルタを含み、ここにおいて、前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた、前記ビデオデータ中の情報に基づいて決定され、

前記コーディングユニットを符号化することと、ここにおいて、前記コーディングユニットを符号化することは、動き推定および動き補償のために補間フィルタを選択することを含み、ここにおいて、前記補間フィルタは、前記補間フィルタのサブセットから以前にコーディングされたブロックのために決定された補間フィルタから選択され、

符号化されたビデオビットストリームを生成することと、ここにおいて、前記符号化されたビデオビットストリームは、前記符号化されたコーディングユニットを含む、

50

を行うように構成されたプロセッサと
を備える装置。

【請求項 1 1】

装置であって、

符号化されたビデオデータを記憶するように構成されたメモリと、

符号化されたビデオビットストリームを取得することと、

前記符号化されたビデオビットストリームから現在のフレーム中のコーディングユニットのための参照フレームを決定することと、

補間フィルタのセットから補間フィルタのサブセットを決定することと、前記補間フィルタのサブセットは複数の補間フィルタを含み、ここにおいて、前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた、前記符号化されたビデオビットストリーム中の情報に基づいて決定され、

前記補間フィルタのサブセットから以前にコーディングされたブロックのために決定された補間フィルタから補間フィルタを決定することと、

前記コーディングユニットを再構成するために、前記参照フレームおよび前記補間フィルタを使用することと

を行うように構成されたプロセッサと

を備える装置。

【請求項 1 2】

前記装置は、ピクチャをキャプチャするためのカメラを有するモバイルデバイスを備える、請求項 1 0 に記載の装置。

【請求項 1 3】

命令を記憶したコンピュータ可読媒体であって、前記命令は、プロセッサによって実行されると、前記プロセッサに、請求項 1、請求項 3 ~ 請求項 8 のうちのいずれかの方法を実行させる、コンピュータ可読媒体。

【請求項 1 4】

前記補間フィルタは、前記コーディングユニットの隣接ブロックのために決定された補間フィルタから選択される、請求項 2 に記載の方法。

【請求項 1 5】

前記補間フィルタを第 2 の補間フィルタとマージすること、ここにおいて、結果として得られるマージされた補間フィルタが、動き推定および動き補償のために使用される、をさらに備える、請求項 2 に記載の方法。

【請求項 1 6】

前記補間フィルタは、第 2 の補間フィルタに関連付けられ、前記補間フィルタは、整数ピクセル位置のためのものであり、前記第 2 の補間フィルタは、分数ピクセル位置のためのものである、請求項 2 に記載の方法。

【請求項 1 7】

前記補間フィルタと前記第 2 の補間フィルタは、異なるカットオフ周波数を有する、請求項 1 6 に記載の方法。

【請求項 1 8】

前記符号化されたビデオビットストリームは、前記補間フィルタを識別する値を含む、請求項 2 に記載の方法。

【請求項 1 9】

前記装置は、ピクチャをキャプチャするためのカメラを有するモバイルデバイスを備える、請求項 1 1 に記載の装置。

【請求項 2 0】

命令を記憶したコンピュータ可読媒体であって、前記命令は、プロセッサによって実行されると、前記プロセッサに、請求項 2、請求項 9、請求項 1 4 ~ 請求項 1 8 のうちのいずれかの方法を実行させる、コンピュータ可読媒体。

【発明の詳細な説明】

10

20

30

40

50

【技術分野】

【0001】

[0001]本願は、JEM (Joint Exploration Model) で開発されているコードを含む、アドバンスドビデオコーデックのコンテキストにおいて、動き推定 (ME) および動き補償 (MC) プロセスにおいて使用されることになる補間フィルタを切り替えることに関する。例えば、提案された方法は、適応的なやり方で、複数の補間フィルタを採用することによって、インターレイヤ予測のパフォーマンスを改善する。

【背景技術】

【0002】

[0002]ビデオコーディング規格は、ITU-T H.261、ISO/IEC MPEG-1 Visual、ITU-T H.262またはISO/IEC MPEG-2 Visual、ITU-T H.263、ISO/IEC MPEG-4 Visual、および (ISO/IEC MPEG-4 AVCとしても知られる) ITU-T H.264を含み、そのスケーラブルビデオコーディング (SVC) およびマルチビュービデオコーディング (MVC) 拡張を含む。

【0003】

[0003]加えて、そのレンジおよびスクリーンコンテンツコーディング拡張と、3Dビデオコーディング (3D-HEVC) スケーラブル拡張 (SHVC) とおよびマルチビュー拡張 (MV-HEVC) と、を含む、新しいビデオコーディング規格、すなわち高効率ビデオコーディング (HEVC) またはITU-T H.265が、ビデオコーディング共同研究部会 (JCT-V C)、ならびにITU-Tビデオコーディングエキスパートグループ (VCEG) と、ISO/IECモーションピクチャエキスパートグループ (MPEG) との3Dビデオコーディング拡張開発共同研究部会 (JCT-3V) によって、近年開発されてきた。

【0004】

[0004]2016年に、MPEGとITU-T VCEGは、次世代のビデオコーディング規格についての新しいコーディングツールを調査するために、JVET (joint exploration video team) を設立した。参照ソフトウェアは、JEM (joint exploration model) と呼ばれる。

【発明の概要】

【0005】

[0005]ビデオ圧縮技法は、入力ビデオ信号に固有の冗長性を低減または取り除くために、空間的および時間的予測を実行する。時間的冗長性 (すなわち、隣接フレームにおける視覚的類似性) を低減させるために、動き推定が行われて、ビデオにおけるオブジェクトの動き (movement) を追跡する。動きベクトルは、ピクセル単位の距離で変位を示し得る。いくつかのケースでは、動きベクトルは、1/2ピクセル、1/4ピクセル、または1/16ピクセル距離など、整数ピクセル精度よりも高い精度を有し得る。より細かい精度は、ビデオコードがより正確に動きフィールドを追跡し、したがって、より良い予測を得ることを可能にし得る。

【0006】

[0006]少なくとも1つの例によると、ビデオデータを取得することを含む、ビデオデータを符号化する方法が提供される。方法は、コーディングユニットについて、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに含み、ここにおいて、補間フィルタのサブセットは、コーディングユニットに関連付けられた、ビデオデータ中の情報に基づいて決定される。方法は、コーディングユニットを符号化することをさらに含み、ここにおいて、コーディングユニットを符号化することは、コーディングユニットについての動き推定および動き補償のために補間フィルタを選択することを含み、ここにおいて、補間フィルタは、補間フィルタのサブセットから選択される。方法は、符号化されたビデオビットストリームを生成することをさらに含み、ここにおいて、符号化されたビデオビットストリームは、符号化されたコーディングユニットを含む。

【 0 0 0 7 】

[0007]別の例では、ビデオデータを記憶するように構成されたメモリと、プロセッサとを含む装置が提供される。プロセッサは、ビデオデータを取得し得るように構成される。プロセッサは、コーディングユニットについて、補間フィルタのセットから補間フィルタのサブセットを決定し得るように構成され、ここにおいて、補間フィルタのサブセットは、コーディングユニットに関連付けられた、ビデオデータ中の情報に基づいて決定される。プロセッサは、コーディングユニットを符号化し得るように構成され、ここにおいて、処理を符号化することは、動き推定および動き補償のために補間フィルタを選択することを含み、ここにおいて、補間フィルタは、補間フィルタのサブセットから選択される。プロセッサは、符号化されたビデオビットストリームを生成し得るように構成され、ここにおいて、符号化されたビデオビットストリームは、符号化されたコーディングユニットを含む。

10

【 0 0 0 8 】

[0008]別の例では、プロセッサによって実行されると、ビデオデータを取得することを含む方法を実行する命令を記憶したコンピュータ可読媒体が提供される。方法は、コーディングユニットについて、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに含み、ここにおいて、補間フィルタのサブセットは、コーディングユニットに関連付けられた、ビデオデータ中の情報に基づいて決定される。方法は、コーディングユニットを符号化することをさらに含み、ここにおいて、コーディングユニットを符号化することは、コーディングユニットについての動き推定および動き補償のために補間フィルタを選択することを含み、ここにおいて、補間フィルタは、補間フィルタのセットから選択される。方法は、符号化されたビデオビットストリームを生成することをさらに含み、ここにおいて、符号化されたビデオビットストリームは、符号化されたコーディングユニットを含む。

20

【 0 0 0 9 】

[0009]別の例では、ビデオデータを取得するための手段を含む装置が提供される。装置は、コーディングユニットについて、補間フィルタのセットから補間フィルタのサブセットを決定するための手段をさらに含み、ここにおいて、補間フィルタのサブセットは、コーディングユニットに関連付けられた、ビデオデータ中の情報に基づいて決定される。装置は、コーディングユニットを符号化するための手段をさらに備え、ここにおいて、コーディングユニットを符号化することは、コーディングユニットについての動き推定および動き補償のために補間フィルタを選択することを含み、ここにおいて、補間フィルタは、補間フィルタのサブセットから選択される。装置は、符号化されたビデオビットストリームを生成するための手段をさらに備え、ここにおいて、符号化されたビデオビットストリームは、符号化されたコーディングユニットを含む。

30

【 0 0 1 0 】

[0010]いくつかの態様では、上記で説明された方法、装置、およびコンピュータ可読媒体は、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに備え、ここにおいて、補間フィルタのサブセットは、コーディングユニットのコーディングレベルに基づいて決定され、ここにおいて、補間フィルタは、補間フィルタのサブセットから決定される。

40

【 0 0 1 1 】

[0011]いくつかの態様では、上記で説明された方法、装置、およびコンピュータ可読媒体は、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに備え、ここにおいて、補間フィルタのサブセットは、コーディングユニットのための予測モードに基づいて決定され、ここにおいて、補間フィルタは、補間フィルタのサブセットから決定される。

【 0 0 1 2 】

[0012]いくつかの態様では、上記で説明された方法、装置、およびコンピュータ可読媒体は、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに備え

50

、ここにおいて、補間フィルタのサブセットは、動き情報に基づいて決定され、ここにおいて、補間フィルタは、補間フィルタのサブセットから決定される。

【 0 0 1 3 】

[0013]いくつかの態様では、上記で説明された方法、装置、およびコンピュータ可読媒体は、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに備え、ここにおいて、補間フィルタのサブセットは、コーディングユニットをコーディングするために使用されるコーディングツールに基づいて決定され、ここにおいて、補間フィルタは、補間フィルタのサブセットから決定される。

【 0 0 1 4 】

[0014]いくつかの態様では、補間フィルタは、以前にコーディングされたブロックから選択される。いくつかの態様では、補間フィルタは、コーディングユニットの隣接ブロックから選択される。

10

【 0 0 1 5 】

[0015]いくつかの態様では、上記で説明された方法、装置、およびコンピュータ可読媒体は、補間フィルタを第2の補間フィルタとマージすることをさらに備え、ここにおいて、結果として得られるマージされた補間フィルタは、動き推定のために使用される。

【 0 0 1 6 】

[0016]いくつかの態様では、補間フィルタは、第2の補間フィルタに関連付けられ、ここにおいて、補間フィルタは、整数ピクセル位置のためのものであり、ここにおいて、第2の補間フィルタは、分数ピクセル位置のためのものである。

20

【 0 0 1 7 】

[0017]いくつかの態様では、補間フィルタと第2の補間フィルタは、異なるカットオフ周波数 (cut-off frequencies) を有する。

【 0 0 1 8 】

[0018]いくつかの態様では、符号化されたビデオビットストリームは、補間フィルタを識別する値を含む。

【 0 0 1 9 】

[0019]いくつかの態様では、補間フィルタのセットは、符号化されたビデオビットストリームとともにシグナリングされず、ここにおいて、補間フィルタの識別は、符号化されたビデオビットストリームから暗示され得る。

30

【 0 0 2 0 】

[0020]いくつかの態様では、上記で説明されたような装置は、ピクチャをキャプチャするためのカメラを含み得る。

【 0 0 2 1 】

[0021]少なくとも1つの例によると、符号化されたビデオビットストリームを取得することを含む、ビデオデータを復号する方法が提供される。方法は、符号化されたビデオビットストリームから現在のフレーム中のコーディングユニットのための参照フレームを決定することをさらに含む。方法は、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに含む、ここにおいて、補間フィルタのサブセットは、コーディングユニットに関連付けられた、符号化されたビデオビットストリーム中の情報に基づいて決定される。方法は、補間フィルタのサブセットから補間フィルタを決定することをさらに含む。方法は、コーディングユニットを再構成するために、参照フレームおよび補間フィルタを使用することをさらに含む。

40

【 0 0 2 2 】

[0022]別の例では、ビデオデータを記憶するように構成されたメモリと、プロセッサとを含む装置が提供される。プロセッサは、符号化されたビデオビットストリームを取得し得るように構成される。プロセッサは、符号化されたビデオビットストリームから現在のフレーム中のコーディングユニットのための参照フレームを決定し得るように構成される。プロセッサは、補間フィルタのセットから補間フィルタのサブセットを決定するようにさらに構成され、ここにおいて、補間フィルタのサブセットは、コーディングユニットに

50

関連付けられた、符号化されたビデオビットストリーム中の情報に基づいて決定される。プロセッサは、補間フィルタのサブセットから補間フィルタを決定し得るように構成される。プロセッサは、コーディングユニットを再構成するために、参照フレームおよび補間フィルタを使用し得るように構成される。

【0023】

[0023]別の例では、プロセッサによって実行されると、符号化されたビデオビットストリームを取得することを含む方法を実行する命令を記憶したコンピュータ可読媒体が提供される。方法は、符号化されたビデオビットストリームから現在のフレーム中のコーディングユニットのための参照フレームを決定することをさらに含む。方法は、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに含み、ここにおいて、補間フィルタのサブセットは、コーディングユニットに関連付けられた、符号化されたビデオビットストリーム中の情報に基づいて決定される。方法は、補間フィルタのセットから補間フィルタを決定することをさらに含む。方法は、コーディングユニットを再構成するために、参照フレームおよび補間フィルタを使用することをさらに含む。

10

【0024】

[0024]別の例では、符号化されたビデオビットストリームを取得するための手段を含む装置が提供される。装置は、符号化されたビデオビットストリームから現在のフレーム中のコーディングユニットのための参照フレームを決定するための手段をさらに備える。装置は、補間フィルタのセットから補間フィルタのサブセットを決定するための手段をさらに含み、ここにおいて、補間フィルタのサブセットは、コーディングユニットに関連付けられた、符号化されたビデオビットストリーム中の情報に基づいて決定される。装置は、補間フィルタのサブセットから補間フィルタを決定するための手段をさらに備える。装置は、コーディングユニットを再構成するために、参照フレームおよび補間フィルタを使用するための手段をさらに備える。

20

【0025】

[0025]いくつかの態様では、上記で説明された方法、装置、およびコンピュータ可読媒体は、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに備え、ここにおいて、補間フィルタのサブセットは、コーディングユニットのコーディングレベルに基づいて決定され、ここにおいて、補間フィルタは、補間フィルタのサブセットから決定される。

30

【0026】

[0026]いくつかの態様では、上記で説明された方法、装置、およびコンピュータ可読媒体は、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに備え、ここにおいて、補間フィルタのサブセットは、コーディングユニットのための予測モードに基づいて決定され、ここにおいて、補間フィルタは、補間フィルタのサブセットから決定される。

【0027】

[0027]いくつかの態様では、上記で説明された方法、装置、およびコンピュータ可読媒体は、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに備え、ここにおいて、補間フィルタのサブセットは、動き情報に基づいて決定され、ここにおいて、補間フィルタは、補間フィルタのサブセットから決定される。

40

【0028】

[0028]いくつかの態様では、上記で説明された方法、装置、およびコンピュータ可読媒体は、補間フィルタのセットから補間フィルタのサブセットを決定することをさらに備え、ここにおいて、補間フィルタのサブセットは、コーディングユニットをコーディングするために使用されるコーディングツールに基づいて決定され、ここにおいて、補間フィルタは、補間フィルタのサブセットから決定される。

【0029】

[0029]いくつかの態様では、補間フィルタは、以前にコーディングされたブロックから選択される。いくつかの態様では、補間フィルタは、隣接ブロックから選択される。

50

【 0 0 3 0 】

[0030]いくつかの態様では、上記で説明された方法、装置、およびコンピュータ可読媒体は、補間フィルタを第2の補間フィルタとマージすることをさらに備え、ここにおいて、結果として得られるマージされた補間フィルタは、予測ユニットを再構成するために使用される。

【 0 0 3 1 】

[0031]いくつかの態様では、補間フィルタは、第2の補間フィルタに関連付けられ、ここにおいて、補間フィルタは、整数ピクセル位置のためのものであり、ここにおいて、第2の補間フィルタは、分数ピクセル位置のためのものである。いくつかの態様では、補間フィルタと第2の補間フィルタは、異なるカットオフ周波数を有する。

10

【 0 0 3 2 】

[0032]いくつかの態様では、符号化されたビデオビットストリームは、補間フィルタを識別する値を含む。

【 0 0 3 3 】

[0033]いくつかの態様では、上記で説明された方法、装置、およびコンピュータ可読媒体は、符号化されたビデオビットストリームを使用して、補間フィルタの識別を導出することをさらに備える。

【 0 0 3 4 】

[0034]いくつかの態様では、上記で説明されたような装置は、ピクチャをキャプチャするためのカメラを含み得る。いくつかの態様では、装置は、ビデオデータを表示するためのディスプレイを含み得る。

20

【 図面の簡単な説明 】

【 0 0 3 5 】

[0035]本発明の例示的な実施形態が、以下の図面を参照して、以下で詳細に説明される。

【 図 1 】 [0036]図1は、符号化デバイスおよび復号デバイスの例を例示するブロック図である。

【 図 2 】 [0037]図2は、現在のフレームからの現在のブロックについての符号化プロセスの例を例示する。

【 図 3 】 [0038]図3は、分数ピクセル位置の例を例示する。

30

【 図 4 】 [0039]図4は、符号化プロセスの動き推定ステップの例を例示する。

【 図 5 】 [0040]図5は、復号プロセスの動き推定ステップの例を例示する。

【 図 6 】 [0041]図6は、3つの例となるフィルタ、F0、F1、およびF2の周波数応答を例示するグラフである。

【 図 7 】 [0042]図7は、符号化プロセス中に補間フィルタを切り替えるためのプロセスの例を例示する。

【 図 8 】 [0043]図8は、復号プロセスにおいて補間フィルタを切り替えるためのプロセスの例である。

【 図 9 】 [0044]図9は、例となる符号化デバイスを例示するブロック図である。

【 図 1 0 】 [0045]図10は、例となるビデオ復号デバイスを例示するブロック図である。

40

【 詳細な説明 】

【 0 0 3 6 】

[0046]本開示のある特定の態様および実施形態が、以下で提供される。当業者に明らかであるように、これらの態様および実施形態のうちのいくつかは、独立に適用され得、また、それらのうちのいくつかは、組合せにおいて適用され得る。以下の説明では、説明を目的として、特定の詳細が本発明の実施形態の完全な理解を提供するために示される。しかしながら、様々な実施形態が、これらの特定の詳細なしで実施され得ることは、明らかであろう。図面および説明は、限定的であるようには意図されない。

【 0 0 3 7 】

[0047]次に続く説明は、例示的な実施形態のみを提供しており、本開示の範囲、適用可

50

能性、または構成を限定するようには意図されない。むしろ、例示的な実施形態の次に続く説明は、例示的な実施形態をインプリメントすることを可能にする説明を当業者に提供するであろう。様々な変更が、添付の特許請求の範囲に示されるような本発明の趣旨および範囲から逸脱することなく、要素の配置および機能において行われ得ることが理解されるべきである。

【 0 0 3 8 】

[0048]特定の詳細は、実施形態の完全な理解を提供するために、以下の説明において与えられる。しかしながら、当業者であれば、実施形態が、これらの特定の詳細なしで実施され得ることを理解するであろう。例えば、回路、システム、ネットワーク、プロセス、および他の構成要素が、不要な詳細で実施形態を曖昧にしないために、ブロック図形式の構成要素として示され得る。他の事例では、周知の回路、プロセス、アルゴリズム、構造、および技法が、実施形態を曖昧にすることを避けるために、不要な詳細なしで示され得る。

10

【 0 0 3 9 】

[0049]また、個々の実施形態は、フローチャート、フロー図、データフロー図、構造図、またはブロック図として図示されるプロセスとして説明され得ることに留意されたい。フローチャートは、シーケンシャルプロセスとして動作を説明し得るが、動作の多くは、並行してまたは同時並行に実行され得る。加えて、動作の順序は再配置され得る。プロセスは、その動作が完了したときに終了するが、図面に含まれていない追加のステップを有し得る。プロセスは、方法、関数、プロシージャ、サブルーチン、サブプログラム、等に対応し得る。プロセスが関数に対応するとき、その終了は、この関数が呼び出し関数または主関数に戻ることに対応し得る。

20

【 0 0 4 0 】

[0050]「コンピュータ可読媒体」という用語は、それに限定されるものではないが、ポータブルまたは非ポータブル記憶デバイス、光記憶デバイス、および（１つまたは複数の）命令および／またはデータを記憶、包含、または搬送することが可能な様々な他の媒体を含む。コンピュータ可読媒体は、データが記憶されることができかつワイヤレスにまたはワイヤード接続を介して伝搬する搬送波および／または一時的な電子信号を含まない、非一時的な媒体を含み得る。非一時的な媒体の例は、それに限定されるものではないが、磁気ディスクまたはテープ、コンパクトディスク（ＣＤ）またはデジタル多目的ディスク（ＤＶＤ）のような光記憶媒体、フラッシュメモリ、メモリまたはメモリデバイスを含み得る。コンピュータ可読媒体は、プロシージャ、関数、サブプログラム、プログラム、ルーチン、サブルーチン、モジュール、ソフトウェアパッケージ、クラス、あるいは命令、データ構造、またはプログラムステートメントの任意の組合せを表し得るコードおよび／または機械実行可能命令を記憶し得る。コードセグメントは、情報、データ、引き数、パラメータ、またはメモリコンテンツを渡すことおよび／または受け取ることによって、別のコードセグメントまたはハードウェア回路に結合され得る。情報、引き数、パラメータ、データ、等は、メモリ共有、メッセージパッシング、トークンパッシング、ネットワーク送信、または同様のものを含む、任意の好適な手段を介して渡される、転送される、または送信され得る。

30

40

【 0 0 4 1 】

[0051]さらに、実施形態は、ハードウェア、ソフトウェア、ファームウェア、ミドルウェア、マイクロコード、ハードウェア記述言語、またはこれらの任意の組合せによってインプリメントされ得る。ソフトウェア、ファームウェア、ミドルウェアまたはマイクロコードでインプリメントされるとき、必要なタスクを実行するためのプログラムコードまたはコードセグメント（例えば、コンピュータプログラム製品）は、コンピュータ可読または機械可読媒体中に記憶され得る。（１つまたは複数の）プロセッサが、必要なタスクを実行し得る。

【 0 0 4 2 】

[0052]より多くのデバイスおよびシステムが、デジタルビデオデータを消費する能力を

50

消費者に提供するにつれて、効率的なビデオコーディング技法の必要性がより重要になる。ビデオコーディングは、デジタルビデオデータ中に存在する大量のデータを処理する (handle) のに必要な記憶および送信要件を低減させるために必要である。様々なビデオコーディング技法が、高いビデオ品質を維持しながら、ビデオデータをより低いビットレートを使用する形態へと圧縮するために使用され得る。ここで使用される場合、「コーディング」は、「符号化」または「復号」を指す。

【 0 0 4 3 】

[0053]図 1 は、符号化デバイス 1 0 4 および復号デバイス 1 1 2 を含む、ビデオコーディングシステム 1 0 0 の例を例示するブロック図である。符号化デバイス 1 0 4 は、ソースデバイスの一部であり得、復号デバイス 1 1 2 は、受信デバイスの一部であり得る。ソースデバイスおよび / または受信デバイスは、モバイルまたは固定の電話ハンドセット (例えば、スマートフォン、セルラ電話、または同様のもの)、デスクトップコンピュータ、ラップトップまたはノートブックコンピュータ、タブレットコンピュータ、セットトップボックス、テレビ、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、ビデオゲーム機、ビデオストリーミングデバイス、インターネットプロトコル (IP) カメラなどの電子デバイス、またはその他任意の好適な電子デバイスを含み得る。いくつかの例では、ソースデバイスおよび受信デバイスは、ワイヤレス通信のための 1 つまたは複数のワイヤレストランシーバを含み得る。ここで説明されるコーディング技法は、(例えば、インターネットを介した) ストリーミングビデオ送信、テレビ放送または送信、データ記憶媒体上での記憶のためのデジタルビデオの符号化、データ記憶媒体上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を含む、様々なマルチメディアアプリケーションにおけるビデオコーディングに適用可能である。いくつかの例では、システム 1 0 0 は、ビデオ会議、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、ゲーミング、および / またはビデオ電話通信などの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し得る。

【 0 0 4 4 】

[0054]符号化デバイス 1 0 4 (またはエンコーダ) は、符号化されたビデオビットストリームを生成するために、ビデオコーディング規格またはプロトコルを使用して、ビデオデータを符号化するために使用され得る。ビデオコーディング規格の例は、ITU - T H . 2 6 1、ISO / IEC MPEG - 1 Visual、ITU - T H . 2 6 2 または ISO / IEC MPEG - 2 Visual、ITU - T H . 2 6 3、ISO / IEC MPEG - 4 Visual、そのスケーラブルビデオコーディング (SVC) およびマルチビュービデオコーディング (MVC) 拡張を含む (ISO / IEC MPEG - 4 AVCとしても知られている) ITU - T H . 2 6 4、および高効率ビデオコーディング (HEVC) または ITU - T H . 2 6 5 を含む。マルチレイヤビデオコーディングを扱う HEVC への様々な拡張が存在し、レンジおよびスクリーンコンテンツコーディング拡張、3D ビデオコーディング (3D - HEVC) およびマルチビュー拡張 (MV - HEVC) およびスケーラブル拡張 (SHVC) を含む。HEVC およびその拡張は、ビデオコーディング共同研究部会 (JCT - VC)、ならびに ITU - T ビデオコーディングエキスパートグループ (VCEG) と、ISO / IEC モーションピクチャエキスパートグループ (MPEG) との 3D ビデオコーディング拡張開発共同研究部会 (JCT - 3V) によって開発されてきた。MPEG と ITU - T VCEG はまた、次世代のビデオコーディング規格のための新しいコーディングツールを調査するために、JVET (joint exploration video team) を設立した。参照ソフトウェアは、JEM (joint exploration model) と呼ばれる。

【 0 0 4 5 】

[0055]ここで説明される多くの例は、JEM モデル、HEVC 規格、および / またはそれらの拡張を使用した例を提供する。しかしながら、ここで説明される技法およびシステムはまた、AVC、MPEG、それらの拡張などの他のコーディング規格、または現在存在している他の好適なコーディング規格または将来のコーディング規格に適用可能であり

10

20

30

40

50

得る。したがって、ここで説明される技法およびシステムが、特定のビデオコーディング規格を参照して説明され得る一方で、当業者であれば、説明がその特定の規格にのみ適用されると解釈されるべきではないことを理解するであろう。

【0046】

[0056]図1を参照すると、ビデオソース102が、符号化デバイス104にビデオデータを提供し得る。ビデオソース102は、ソースデバイスの一部であり得、またはソースデバイス以外のデバイスの一部であり得る。ビデオソース102は、ビデオキャプチャデバイス（例えば、ビデオカメラ、カメラフォン、ビデオフォン、または同様のもの）、記憶されたビデオを含むビデオアーカイブ、ビデオデータを提供するビデオサーバまたはコンテンツプロバイダ、ビデオサーバまたはコンテンツプロバイダからのビデオを受信するビデオフィードインターフェース、コンピュータグラフィックス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、このようなソースの組合せ、またはその他任意の好適なビデオソースを含み得る。

【0047】

[0057]ビデオソース102からのビデオデータは、1つまたは複数の入力ピクチャまたはフレームを含み得る。ビデオのピクチャまたはフレームは、シーンの静止画像である。符号化デバイス104のエンコーダエンジン106（またはエンコーダ）は、符号化されたビデオビットストリームを生成するためにビデオデータを符号化する。いくつかの例では、符号化されたビデオビットストリーム（あるいは、「ビデオビットストリーム」または「ビットストリーム」）は、一連の1つまたは複数のコーディングされたビデオシーケンスである。コーディングされたビデオシーケンス（CVS）は、ある特定の特性を有し（with certain properties）かつベースレイヤにおけるランダムアクセスポイントピクチャを有するアクセスユニット（AU）から開始し、ある特定の特性を有しかつベースレイヤにおけるランダムアクセスポイントピクチャを有する次のAUまでの（およびそれを含まない）（up to and not including a next AU）一連のAUを含む。例えば、CVSを開始するランダムアクセスポイントピクチャのある特定の特性は、1に等しいRASLフラグ（例えば、NoRASLOutputFlag）を含み得る。そうでない場合、（0に等しいRASLフラグを有する）ランダムアクセスポイントピクチャは、CVSを開始しない。アクセスユニット（AU）は、1つまたは複数のコーディングされたピクチャと、同じ出力時間を共有するコーディングされたピクチャに対応する制御情報とを含む。ピクチャのコーディングされたスライスは、ネットワーク抽象化レイヤ（NAL）ユニットと呼ばれるデータユニットへとビットストリームレベルでカプセル化される。例えば、HEVCビデオビットストリームは、NALユニットを含む1つまたは複数のCVSを含み得る。NALユニットの各々は、NALユニットヘッダを有する。一例では、ヘッダは、（マルチレイヤ拡張を除き）H.264/AVCの場合は1バイト、およびHEVCの場合は2バイトである。NALユニットヘッダにおけるシンタックス要素は、指定されたビットを占め（take）、したがって、特に、トランスポートストリーム、リアルタイムトランスポート（RTP）プロトコル、ファイル形式などの、全ての種類のシステムおよびトランスポートレイヤに対して可視である。

【0048】

[0058]ビデオコーディングレイヤ（VCL）NALユニットおよび非VCL NALユニットを含む、2つのクラスのNALユニットがHEVC規格に存在する。VCL NALユニットは、コーディングされたピクチャデータの（以下で説明される）1つのスライスまたはスライスセグメントを含み、非VCL NALユニットは、1つまたは複数のコーディングされたピクチャに関連する制御情報を含む。いくつかのケースでは、NALユニットは、パケットと呼ばれ得る。HEVC AUは、コーディングされたピクチャデータを含むVCL NALユニットと、コーディングされたピクチャデータに対応する非VCL NALユニット（もしあれば）とを含む。

【0049】

[0059]NALユニットは、ビデオ中のピクチャのコーディングされた表現のような、ビ

10

20

30

40

50

デオデータのコーディングされた表現を形成するビットのシーケンス（例えば、符号化されたビデオビットストリーム、ビットストリームのC V S、または同様のもの）を含み得る。エンコーダエンジン106は、各ピクチャを複数のスライスに区分化することによって、ピクチャのコーディングされた表現を生成する。スライスは、他のスライスとは独立しており、それによって、スライス中の情報は、同じピクチャ内の他のスライスからのデータに依存することなくコーディングされる。スライスは、独立スライスセグメントと、もしあれば、以前のスライスセグメントに依存する1つまたは複数の従属スライスセグメントとを含む、1つまたは複数のスライスセグメントを含む。その後、スライスは、ルーマサンプルおよびクロマサンプルのコーディングツリーブロック（CTB）に区分化される。ルーマサンプルのCTBおよびクロマサンプルの1つまたは複数のCTBは、サンプルのためのシンタックスと共に、コーディングツリーユニット（CTU）と呼ばれる。CTUは、HEVC符号化のための基本処理ユニットである。CTUは、可変サイズの複数のコーディングユニット（CU）へと分割され得る。CUは、コーディングブロック（CB）と呼ばれるルーマおよびクロマサンプル配列を含む。

【0050】

[0060]ルーマおよびクロマCBは、予測ブロック（PB）へとさらに分割され得る。PBは、（利用可能であるか、または使用のためにイネーブルにされるとき）インター予測またはイントラブロックコピー予測のために同じ動きパラメータを使用するルーマ成分またはクロマ成分のサンプルのブロックである。ルーマPBおよび1つまたは複数のクロマPBは、関連するシンタックスと共に、予測ユニット（PU）を形成する。インター予測の場合、動きパラメータのセット（例えば、1つまたは複数の動きベクトル、参照インデックス、または同様のもの）が、各PUについてビットストリーム中でシグナリングされ、ルーマPBおよび1つまたは複数のクロマPBのインター予測のために使用される。動きパラメータはまた、動き情報と呼ばれ得る。CBはまた、1つまたは複数の変換ブロック（TB）に区分化され得る。TBは、予測残差信号をコーディングするために同じ2次元変換が適用される、色成分のサンプルの正方形ブロックを表す。変換ユニット（TU）は、ルーマおよびクロマサンプルのTBと、対応するシンタックス要素とを表す。

【0051】

[0061]CUのサイズは、コーディングモードのサイズに対応し、形状が正方形であり得る。例えば、CUのサイズは、 8×8 サンプル、 16×16 サンプル、 32×32 サンプル、 64×64 サンプル、または最大で対応するCTUのサイズまでのその他任意の好適なサイズであり得る。「 $N \times N$ 」という表現は、縦および横の寸法の観点から、ビデオブロックのピクセル寸法（例えば、 8 ピクセル \times 8 ピクセル）を指すためにここで使用される。ブロック中のピクセルは、行および列に配置され得る。いくつかの例では、ブロックは、横方向に、縦方向と同じ数のピクセルを有していない場合がある。CUに関連付けられたシンタックスデータは、例えば、1つまたは複数のPUへのCUの区分化を説明し得る。区分化モードは、CUがイントラ予測モード符号化されたか、またはインター予測モード符号化されたかの間で異なり得る。PUは、形状が非正方形になるように区分化され得る。CUに関連付けられたシンタックスデータはまた、例えば、CTUによる、1つまたは複数のTUへのCUの区分化を説明し得る。TUは、形状が正方形または非正方形であり得る。

【0052】

[0062]HEVC規格によると、変換は、変換ユニット（TU）を使用して実行され得る。TUは、異なるCUに対して異なり得る。TUは、所与のCU内のPUのサイズに基づいて、サイズ決定され得る。TUは、PUと同じサイズであるかそれより小さくなり得る。いくつかの例では、CUに対応する残差サンプルは、残差4分木（RQT: residual quad tree）として知られる4分木構造を使用して、より小さいユニットに細分され得る。RQTのリーフノードは、TUに対応し得る。TUに関連付けられたピクセル差分値は、変換係数を作成するために変換され得る。その後、変換係数は、エンコーダエンジン106によって量子化され得る。

10

20

30

40

50

【 0 0 5 3 】

[0063]いったんビデオデータのピクチャがC Uに区分化されると、エンコーダエンジン106は、予測モードを使用して各P Uを予測する。その後、予測ユニットまたは予測ブロックは、(以下で説明される)残差を得るために元のビデオデータから減算される。各C Uについて、予測モードは、シンタックスデータを使用してビットストリームの内部でシグナリングされ得る。予測モードは、イントラ予測(またはイントラピクチャ予測)またはインター予測(またはインターピクチャ予測)を含み得る。イントラ予測は、ピクチャ内の空間的に隣接するサンプル間の相関を利用する。例えば、イントラ予測を使用して、各P Uは、例えば、P Uの平均値を求めるためのDC予測、平面をP Uに合わせるための平面予測、隣接データから外挿するための方向予測、またはその他任意の好適なタイプの予測を使用して、同じピクチャ中の隣接画像データから予測される。インター予測は、画像サンプルのブロックについての動き補償予測を導出するために、ピクチャ間の時間的相関を使用する。例えば、インター予測を使用して、各P Uは、(出力順序で現在のピクチャの前または後の)1つまたは複数の参照ピクチャ中の画像データからの動き補償予測を使用して予測される。ピクチャエリアを、インターピクチャ予測を使用してコーディングするか、またはイントラピクチャ予測を使用してコーディングするかは、例えば、C Uレベルで行われ得る。

10

【 0 0 5 4 】

[0064]いくつかの例では、ピクチャの1つまたは複数のスライス、スライスタイプを割り当てられる。スライスタイプは、Iスライス、Pスライス、およびBスライスを含む。Iスライス(イントラフレーム、独立に復号可能)は、イントラ予測によってのみコーディングされるピクチャのスライスであり、したがって、Iスライスが、スライスの任意の予測ユニットまたは予測ブロックを予測するために、フレーム内のデータのみを必要とすることから、独立に復号可能である。Pスライス(片方向(unidirectional)予測フレーム)は、イントラ予測で、および片方向インター予測でコーディングされ得るピクチャのスライスである。Pスライス内の各予測ユニットまたは予測ブロックは、イントラ予測またはインター予測のいずれかでコーディングされる。インター予測が適用されるとき、予測ユニットまたは予測ブロックは、1つの参照ピクチャによってのみ予測され、したがって、参照サンプルは、1つのフレームの1つの参照領域のみからのものである。Bスライス(双方向予測フレーム)は、イントラ予測で、およびインター予測で(例えば、双予測(bi-prediction)または単予測(uni-prediction)のいずれかで)コーディングされ得るピクチャのスライスである。Bスライスの予測ユニットまたは予測ブロックは、2つの参照ピクチャから双方向に予測され得、ここで、各ピクチャは、1つの参照領域を与え(contributes)、2つの参照領域のサンプルセットは、双方向予測ブロックの予測信号を作成するために、(例えば、等しい重みで、または異なる重みで)重み付けされる。上記で説明されたように、1つのピクチャのスライスは、独立にコーディングされる。いくつかのケースでは、ピクチャは、単に1つのスライスとしてコーディングされ得る。

20

30

【 0 0 5 5 】

[0065]P Uは、予測プロセスに関連するデータ(例えば、動きパラメータまたは他の好適なデータ)を含み得る。例えば、P Uがイントラ予測を使用して符号化されるとき、P Uは、P Uのためのイントラ予測モードを記述するデータを含み得る。別の例として、P Uがインター予測を使用して符号化されるとき、P Uは、P Uのための動きベクトルを定義するデータを含み得る。P Uのための動きベクトルを定義するデータは、例えば、動きベクトルの水平成分(x)、動きベクトルの垂直成分(y)、動きベクトルについての解像度(例えば、整数精度、4分の1ピクセル精度、または8分の1ピクセル精度)、動きベクトルが指し示す参照ピクチャ、参照インデックス、動きベクトルについての参照ピクチャリスト(例えば、リスト0、リスト1、またはリストC)、またはこれらの任意の組合せを記述し得る。

40

【 0 0 5 6 】

[0066]その後、符号化デバイス104は、変換および量子化を実行し得る。例えば、予

50

測に続いて、エンコーダエンジン 106 は、PU に対応する残差値を計算し得る。残差値は、コーディングされているピクセルの現在のブロック (PU) と、現在のブロックを予測するために使用される予測ブロック (例えば、現在のブロックの予測されたバージョン) との間のピクセル差分値を備え得る。例えば、(例えば、インター予測またはイントラ予測を発行して) 予測ブロックを生成した後、エンコーダエンジン 106 は、現在のブロックから予測ユニットによって作成される予測ブロックを減算することによって、残差ブロックを生成し得る。残差ブロックは、現在のブロックのピクセル値と、予測ブロックのピクセル値との間の差分を定量化するピクセル差分値のセットを含む。いくつかの例では、残差ブロックは、2 次元ブロックフォーマット (例えば、ピクセル値の 2 次元行列または配列) で表され得る。このような例では、残差ブロックは、ピクセル値の 2 次元表現である。

10

【0057】

[0067] 予測が実行された後に残り得る任意の残差データは、ブロック変換を使用して変換され、これは、離散コサイン変換、離散サイン変換、整数変換、ウェーブレット変換、他の好適な変換関数 (transform function)、またはこれらの任意の組合せに基づき得る。いくつかのケースでは、1 つまたは複数のブロック変換 (例えば、サイズ 32×32 、 16×16 、 8×8 、 4×4 、または同様のもの) が、各 CU 中の残差データに適用され得る。いくつかの例では、TU が、エンコーダエンジン 106 によってインプリメントされる変換および量子化プロセスのために使用され得る。1 つまたは複数の PU を有する所与の CU がまた、1 つまたは複数の TU を含み得る。以下でさらに詳細に説明されるように、残差値は、ブロック変換を使用して変換係数へと変換され得、その後、エントロピーコーディングのための直列化された変換係数 (serialized transform coefficients) を作成するために、TU を使用して量子化および走査され得る。

20

【0058】

[0068] いくつかの例では、CU の PU を使用するイントラ予測またはインター予測コーディングに続いて、エンコーダエンジン 106 は、CU の TU についての残差データを計算し得る。PU は、空間領域 (またはピクセル領域) 中のピクセルデータを備え得る。TU は、ブロック変換の適用後の変換領域中の係数を備え得る。前述されたように、残差データは、符号化されていないピクチャのピクセルと、PU に対応する予測値との間のピクセル差分値に対応し得る。エンコーダエンジン 106 は、CU についての残差データを含む TU を形成し得、その後、CU のための変換係数を作成するために TU を変換し得る。

30

【0059】

[0069] エンコーダエンジン 106 は、変換係数の量子化を実行し得る。量子化は、係数を表すために使用されるデータの量を低減させるために、変換係数を量子化することによって、さらなる圧縮を提供する。例えば、量子化は、係数の一部または全てに関連付けられたビット深度を低減させ得る。一例では、 n ビットの値を有する係数は、量子化中に m ビットの値に切り捨てられ得るとともに、 n は、 m よりも大きい。

【0060】

[0070] いったん量子化が実行されると、コーディングされたビデオビットストリームは、量子化された変換係数、予測情報 (例えば、予測モード、動きベクトル、ブロックベクトル、または同様のもの)、区分化情報、および他のシンタックスデータのような、その他任意の好適なデータを含む。その後、コーディングされたビデオビットストリームの異なる要素は、エンコーダエンジン 106 によってエントロピー符号化され得る。いくつかの例では、エンコーダエンジン 106 は、エントロピー符号化され得る直列化されたベクトルを作成するために、予め定義された走査順序を利用して、量子化された変換係数を走査し得る。いくつかの例では、エンコーダエンジン 106 は、適応走査を実行し得る。ベクトル (例えば、1 次元ベクトル) を形成するために、量子化された変換係数を走査した後、エンコーダエンジン 106 は、ベクトルをエントロピー符号化し得る。例えば、エンコーダエンジン 106 は、コンテキスト適応可変長コーディング、コンテキスト適応バイナリ算術コーディング、シンタックスベースコンテキスト適応バイナリ算術コーディング

40

50

、確率間隔区分化エントロピーコーディング、または別の好適なエントロピー符号化技法を使用し得る。

【 0 0 6 1 】

[0071] 前述されたように、H E V C ビットストリームは、V C L N A L ユニットおよび非 V C L N A L ユニットを含む、N A L ユニットのグループを含む。V C L N A L ユニットは、コーディングされたビデオビットストリームを形成するコーディングされたピクチャデータを含む。例えば、コーディングされたビデオビットストリームを形成するビットのシーケンスは、V C L N A L ユニット中で再送される。非 V C L N A L ユニットは、他の情報に加えて、符号化されたビデオビットストリームに関連する高レベル情報を有するパラメータセットを含み得る。例えば、パラメータセットは、ビデオパラメータセット (V P S)、シーケンスパラメータセット (S P S)、およびピクチャパラメータセット (P P S) を含み得る。パラメータセットの目的の例は、ビットレート効率、エラー耐性、およびシステムレイヤインターフェースを提供することを含む。各スライスは、復号デバイス 1 1 2 がスライスを復号するために使用し得る情報にアクセスするために、単一のアクティブ P P S、S P S、および V P S を参照する。識別子 (I D) は、V P S I D、S P S I D、および P P S I D を含む各パラメータセットについてコーディングされ得る。S P S は、S P S I D および V P S I D を含む。P P S は、P P S I D および S P S I D を含む。各スライスヘッダは、P P S I D を含む。I D を使用して、アクティブパラメータセットは、所与のスライスのために識別され得る。

【 0 0 6 2 】

[0072] P P S は、所与のピクチャ中の全てのスライスに適用される情報を含む。このことから、ピクチャ中の全てのスライスは、同じ P P S を参照する。異なるピクチャ中のスライスもまた、同じ P P S を参照し得る。S P S は、同じコーディングされたビデオシーケンス (C V S) またはビットストリーム中の全てのピクチャに適用される情報を含む。前述されたように、コーディングされたビデオシーケンスは、(上記で説明された) ある特定の特性を有しかつベースレイヤにおけるランダムアクセスポイントピクチャ (例えば、瞬時復号参照 (I D R : instantaneous decode reference) ピクチャまたはブロークンリンクアクセス (B L A : broken link access) ピクチャ、または他の適切なランダムアクセスポイントピクチャ) から開始し、(ビットストリームの終了または) ある特定の特性を有しかつベースレイヤにおけるランダムアクセスポイントピクチャを有する次のアクセスユニット (A U) までの (およびそれを含まない)、一連の A U である。S P S 中の情報は、コーディングされたビデオシーケンス内のピクチャごとに変化しないことがあり得る。コーディングされたビデオシーケンス中のピクチャは、同じ S P S を使用し得る。V P S は、コーディングされたビデオシーケンスまたはビットストリーム内の全てのレイヤに適用される情報を含む。V P S は、コーディングされたビデオシーケンス全体に適用されるシンタックス要素を有するシンタックス構造を含む。いくつかの例では、V P S、S P S、または P P S は、符号化されたビットストリームとともに帯域内で送信され得る。いくつかの例では、V P S、S P S、または P P S は、コーディングされたビデオデータを含む N A L ユニットとは別個の送信において帯域外で送信され得る。

【 0 0 6 3 】

[0073] ビデオビットストリームはまた、補足エンハンスメント情報 (S E I) メッセージを含み得る。例えば、S E I N A L ユニットは、ビデオビットストリームの一部であり得る。いくつかのケースでは、S E I メッセージは、復号プロセスによって必要とされない情報を含み得る。例えば、S E I メッセージ中の情報は、デコーダがビットストリームのビデオピクチャを復号するのに必須でない場合があるが、デコーダは、ピクチャの表示または処理 (例えば、復号された出力) を改善するために、この情報を使用し得る。S E I メッセージ中の情報は、埋め込まれたメタデータであり得る。1 つの例示的な例では、S E I メッセージ中の情報は、コンテンツの視認性を改善するために、デコーダ側エンティティによって使用され得る。いくつかの事例では、ある特定のアプリケーション規格は、品質の改善が、アプリケーション規格に準拠する全てのデバイスにもたらされ得るよ

10

20

30

40

50

うに、ビットストリーム中のこのような S E I メッセージの存在を規定し得る（例えば、多くの他の例に加えて、フレーム互換平面立体視 3 D T V ビデオフォーマット（frame-compatible plano-stereoscopic 3DTV video format）のためのフレームパッキング S E I メッセージの搬送、ここで、S E I メッセージは、ビデオのフレームごとに搬送され、リカバリポイント S E I メッセージの処理、D V B におけるパンスキャン走査矩形 S E I メッセージの使用）。

【0064】

[0074]符号化デバイス 104 の出力 110 は、受信デバイスの復号デバイス 112 に、通信リンク 120 を介して、符号化されたビデオデータを構成する N A L ユニットを送り得る。復号デバイス 112 の入力 114 は、N A L ユニットを受信し得る。通信リンク 120 は、ワイヤレスネットワーク、ワイヤードネットワーク、またはワイヤードネットワークとワイヤレスネットワークの組合せによって提供されるチャネルを含み得る。ワイヤレスネットワークは、任意のワイヤレスインターフェースまたは複数のワイヤレスインターフェースの組合せを含み得、また、任意の好適なワイヤレスネットワーク（例えば、インターネットまたは他のワイドエリアネットワーク、パケットベースネットワーク、W i F i（登録商標）、無線周波数（R F）、U W B、W i F i - D i r e c t、セルラ、ロングタームエボリューション（L T E（登録商標））、W i M a x（登録商標）、または同様のもの）を含み得る。ワイヤードネットワークは、任意のワイヤードインターフェース（例えば、ファイバー、イーサネット（登録商標）、電力線イーサネット、同軸ケーブルを通じたイーサネット、デジタル信号線（D S L）、または同様のもの）を含み得る。ワイヤードおよび/またはワイヤレスネットワークは、基地局、ルータ、アクセスポイント、ブリッジ、ゲートウェイ、スイッチ、または同様のものなどの、様々な機器を使用してインプリメントされ得る。符号化されたビデオデータは、ワイヤレス通信プロトコルのような通信規格に従って変調され、受信デバイスに送信され得る。

【0065】

[0075]いくつかの例では、符号化デバイス 104 は、符号化されたビデオデータを記憶装置 108 に記憶し得る。出力 110 は、エンコーダエンジン 106 から、または記憶装置 108 から、符号化されたビデオデータを取り出し得る。記憶装置 108 は、様々な分散されたまたはローカルにアクセスされるデータ記憶媒体のうちの任意のものを含み得る。例えば、記憶装置 108 は、ハードドライブ、ストレージディスク、フラッシュメモリ、揮発性または不揮発性メモリ、または符号化されたビデオデータを記憶するためのその他任意の好適なデジタル記憶媒体を含み得る。

【0066】

[0076]復号デバイス 112 の入力 114 は、符号化されたビデオビットストリームデータを受信し、デコーダエンジン 116 に、またはデコーダエンジン 116 による後の使用のために記憶装置 118 に、ビデオビットストリームデータを提供し得る。デコーダエンジン 116 は、（例えば、エントロピーデコーダを使用して）エントロピー復号し、符号化されたビデオデータを構成する 1 つまたは複数のコーディングされたビデオシーケンスの要素を抽出することによって、符号化されたビデオビットストリームデータを復号し得る。その後、デコーダエンジン 116 は、符号化されたビデオビットストリームデータの再スケーリング（rescale）を行い、それに対して逆変換を実行し得る。その後、残差データが、デコーダエンジン 116 の予測ステージに渡される。その後、デコーダエンジン 116 は、ピクセルのブロック（例えば、P U）を予測する。いくつかの例では、予測は、逆変換の出力（残差データ）に加算（added）される。

【0067】

[0077]復号デバイス 112 は、復号されたビデオをビデオ宛先デバイスに出力し得、これは、復号されたビデオデータをコンテンツの消費者に表示するためのディスプレイまたは他の出力デバイスを含み得る。いくつかの態様では、ビデオ宛先デバイス 122 は、復号デバイス 112 を含む受信デバイスの一部であり得る。いくつかの態様では、ビデオ宛先デバイス 122 は、受信デバイス以外の別個のデバイスの一部であり得る。

【 0 0 6 8 】

[0078]いくつかの例では、ビデオ符号化デバイス 1 0 4 および / またはビデオ復号デバイス 1 1 2 は、それぞれ、オーディオ符号化デバイスおよびオーディオ復号デバイスと一体化され得る。ビデオ符号化デバイス 1 0 4 および / またはビデオ復号デバイス 1 1 2 はまた、1 つまたは複数のマイクロプロセッサ、デジタルシグナルプロセッサ (D S P)、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ゲートアレイ (F P G A)、ディスクリートロジック、ソフトウェア、ハードウェア、ファームウェアまたはこれらの任意の組合せなどの、上記で説明されたコーディング技法をインプリメントするのに必要な他のハードウェアまたはソフトウェアを含み得る。ビデオ符号化デバイス 1 0 4 およびビデオ復号デバイス 1 1 2 は、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) の一部として一体化され得る。符号化デバイス 1 0 4 の特定の詳細の例が、図 9 を参照して以下で説明される。復号デバイス 1 1 2 の特定の詳細の例が、図 1 0 を参照して以下で説明される。

10

【 0 0 6 9 】

[0079] H E V C 規格に対する拡張は、M V - H E V C と呼ばれる、マルチビュービデオコーディング拡張と、S H V C と呼ばれる、スケーラブルビデオコーディング拡張とを含む。M V - H E V C および S H V C 拡張は、レイヤードコーディングの概念を共有するとともに、異なるレイヤが、符号化されたビデオビットストリーム中に含まれている。コーディングされたビデオシーケンス中の各レイヤは、一意のレイヤ識別子 (I D) によってアドレス指定 (addressed) される。レイヤ I D は、N A L ユニットが関連付けられているレイヤを識別するために、N A L ユニットのヘッダ中に存在し得る。M V - H E V C では、異なるレイヤは、通常、ビデオビットストリーム中の同じシーンの異なるビューを表す。S H V C では、異なる空間解像度 (またはピクチャ解像度) で、または異なる再構成忠実度でビデオビットストリームを表す異なるスケーラブルレイヤが提供される。スケーラブルレイヤは、(レイヤ I D = 0 を有する) ベースレイヤと、(レイヤ I D = 1、2、... n を有する) 1 つまたは複数のエンハンスメントレイヤとを含み得る。ベースレイヤは、H E V C の第 1 のバージョンのプロファイルに準拠し得、ビットストリーム中の最下位の利用可能なレイヤを表す。エンハンスメントレイヤは、ベースレイヤと比較して、増大された空間解像度、時間解像度またはフレームレート、および / または再構成忠実度 (または品質) を有する。エンハンスメントレイヤは、階層的に編成され、下位レイヤに依存すること (またはしないこと) もあり得る。いくつかの例では、異なるレイヤは、単一標準コーデックを使用してコーディングされ得る (例えば、全てのレイヤが、H E V C、S H V C、または他のコーディング規格を使用して符号化される)。いくつかの例では、異なるレイヤは、多標準コーデックを使用してコーディングされ得る。例えば、ベースレイヤは、A V C を使用してコーディングされ得、一方、1 つまたは複数のエンハンスメントレイヤは、H E V C 規格に対する S H V C および / または M V - H E V C 拡張を使用してコーディングされ得る。

20

30

【 0 0 7 0 】

[0080] ビデオ圧縮技法は、入力ビデオ信号に固有の冗長性を低減または取り除くために、空間的および時間的予測を実行する。時間的冗長性 (すなわち、隣接フレームにおける視覚的類似性) を低減させるために、動き推定が行われて、ビデオにおけるオブジェクトの動きを追跡する。動き推定は、現在のフレームからの所与のピクセルのブロックについて、1 つの参照フレーム中の (または双予測の場合は 2 つの参照フレーム中の) 最も良く一致する (例えば、最も類似した) ピクセルのブロックを識別するプロセスである。1 つまたは複数の参照フレームが、1 つまたは複数の参照インデックスを使用して、現在のブロックのために識別され得る。参照フレーム中のピクセルのブロックは、候補ブロック、予測子 (predictor)、または予測ブロックと呼ばれ得る。ブロックのサイズは異なり得る。現在のフレームからの現在のブロックと、参照フレームからの候補ブロックとの間のオフセットは、動きベクトルによって特徴付けられ得、これは、変位の程度 (degree of displacement) および変位の方向を示し得る。

40

50

【 0 0 7 1 】

[0081]動きベクトルは、ピクセル単位の距離で変位を示し得る。いくつかのケースでは、動きベクトルは、 $1/2$ ピクセル、 $1/4$ ピクセル、または $1/16$ ピクセル距離など、整数ピクセル精度よりも高い精度を有し得る。より細かい精度は、ビデオコードがより正確に動きフィールドを追跡し、したがって、より良い予測を得ることを可能にし得る。

【 0 0 7 2 】

[0082]ビデオコードがフレームを再構成しており、分数ピクセル値を有する動きベクトルに遭遇したとき、ビデオコードは、分数ピクセル値を生成するために補間を行い得る。例えば、および以下でさらに説明されるように、ビデオコードは、 $1/2$ ピクセル、 $1/4$ ピクセル、または何らかの他の分数ピクセル位置についての補間されたピクセルを作成するために、利用可能なピクセルに補間フィルタを適用し得る。

10

【 0 0 7 3 】

[0083]多くのビデオコーデックは、固定補間フィルタを使用する。固定補間フィルタでは、エンコードおよびデコードの両方が、前もってフィルタ係数を提供され得、これら係数は、符号化されたビットストリームにおいて提供される必要がない。

【 0 0 7 4 】

[0084]しかしながら、動き推定および動き補償は、固定フィルタではなく、最適な補間フィルタを使用することによって、最適化され得る。所与の状況についての最良のフィルタは、高度にローカライズされおよびコンテンツ依存である傾向にある。したがって、最適なフィルタを使用することは、フィルタ係数がエンコードからデコードに送信される必要があるという点で、シグナリングオーバーヘッドを著しく増大させ得、および/またはエンコードの複雑さを増大させ得る。

20

【 0 0 7 5 】

[0085]様々なインプリメンテーションでは、異なる補間フィルタが、異なるコーディングユニットに適用され得、かつコーディングされたビットストリームのサイズに対する最小限の追加の増大で (with minimal additional increase) シグナリングされ得るシステムおよび方法が提供される。例えば、HEVCの場合、フィルタ候補の特定のセットは、コーディングユニットが、特に、シーケンスレベル、ピクチャレベル、コーディングツリーユニットレベル、コーディングユニットレベル、変換ユニットレベル、または予測ユニットレベルにあるとき、利用可能になり得る。加えて、JEMの場合、フィルタ候補の特定のセットは、特に、4分木プラス2分木 (Q T B T : quadtree plus binary tree) レベルにおいて利用可能になり得る。ここで、コーディングユニットは、ブロックとも呼ばれ得る。これらおよび他の例では、ビデオコードは、ビデオを符号化またはビットストリームを復号しながら、複数の補間フィルタ間で切り替え得る。

30

【 0 0 7 6 】

[0086]様々なインプリメンテーションでは、異なる特性 (例えば、他の特性の中でも特に、フィルタ長、カットオフ周波数、遷移帯域、またはリップルの量) をそれぞれ有する、N個のフィルタのセットが定義され得る。異なるコーディングレベルのブロックに対して、N個のフィルタのセットのサブセットが定義され得る。様々な基準を使用して、エンコードは、フィルタのサブセットからフィルタを選択し得る。その後、選択されたフィルタは、明示的にまたは暗黙的に、デコードにシグナリングまたは示され得る。所与のレベルで適用されるフィルタのサブセットは、他の基準の中でも特に、以前にコーディングされたブロック中のフィルタのヒット率などの、ある特定の基準に基づいて異なり得る。いくつかのインプリメンテーションでは、整数ピクセル位置またはクロマ成分のためのフィルタ選択がまた、同様のメカニズムを使用して、暗黙的にまたは暗黙的にシグナリングされ得る。

40

【 0 0 7 7 】

[0087]図2は、現在のフレームからの現在のブロック236についての符号化プロセス200の例を例示する。現在のブロック236はまた、コーディングユニットとも呼ばれ得る。現在のブロック236は、現在のフレーム全体よりも小さくなり得る。

50

【 0 0 7 8 】

[0088]この例では、符号化プロセス 2 0 0 は、最初に、現在のブロック 2 3 6 に対して動き推定 2 4 2 を行う。動き推定 2 4 2 は、利用可能な参照フレーム 2 6 4 の中から 1 つまたは複数の参照フレームを識別することを含み得る。参照フレームは、現在のフレームの時間的に前に、または後に発生するフレーム (a frame that occurs before, temporally, or after the current frame) を含み得る。動き推定 2 4 2 を行うことは、現在のブロック 2 3 6 に「最も良く」一致する領域を求めて、(1 つまたは複数の) 参照フレーム中の探索エリアを探索することを含み得る。最良の一致は、例えば、残差エネルギーを決定するために、現在のブロック 2 3 6 から候補領域を減算することによって、決定され得る。この例では、最低の残差エネルギーを有する候補領域が、最良の一致として選ばれ得る。

10

【 0 0 7 9 】

[0089]動き推定 2 4 2 は、動きベクトル 2 4 3 を作成することをさらに含み、これは、(1 つまたは複数の) 参照フレームに関連して (relative to)、現在のブロック 2 3 6 についての動きの程度および方向を推定する。ほとんどのケースでは、1 つまたは複数の参照インデックスが、動きベクトルに関連付けられ、ここで、参照インデックスは、動きベクトルを決定するために使用された (1 つまたは複数の) 参照フレームを識別する。いくつかのケースでは、動き推定の後に、例えば、レート歪みモデルを使用して、最良の動きベクトルが選択され得る。

【 0 0 8 0 】

20

[0090]動き補償 2 4 4 は、最良の動きベクトルを使用して動き補償を実行することによって、予測ブロック 2 4 5 を生成することを含み得る。予測ブロック 2 4 5 は、最良の動きベクトルを使用することからの参照から選択されたブロックである。予測ブロック 2 4 5 は、残差ブロック 2 5 1 を作成するために、現在のブロック 2 3 6 から減算され得る。ブロック変換 2 5 2 が、残差ブロック 2 5 1 に適用され得る。結果として得られる変換係数は、量子化 2 5 4 を経て、量子化された係数 2 5 5 を作成し得る。エントロピーコーディング 2 5 6 が、ビットレートをさらに低減させ、符号化されたビットストリーム 2 1 0 を作成するために、量子化された係数 2 5 5 に適用され得る。

【 0 0 8 1 】

[0091]量子化された係数 2 5 5 はまた、逆量子化 2 5 8 に、そしてその後、逆変換 2 6 0 に入力され得る。逆量子化 2 5 8 および逆変換 2 6 0 は、ブロック変換 2 5 2 および量子化 2 5 4 ステップの逆であり、再構成された残差ブロック 2 6 1 を作成する。再構成された残差ブロック 2 6 1 は、再構成されたブロック 2 5 3 を作成するために、予測されたブロック 2 4 5 に加算され得る。再構成されたブロック 2 5 3 は、現在のブロック 2 3 6 とほぼ同じである。再構成されたビデオブロック 2 5 3 は、追加のブロックを符号化するために使用されることになる参照フレーム中に含まれ得る。

30

【 0 0 8 2 】

[0092]例示的なプロセス 2 0 0 は、主に時間的動き推定を説明する。他の例では、H . 2 6 4 / A V C および H E V C 符号化プロセスなどの符号化プロセスがまた、イントラコーディングされたブロックを作成するために、空間的予測を行い得る。符号化プロセスは、図 9 を参照してより詳細に説明される。

40

【 0 0 8 3 】

[0093]前述されたように、現在のブロックについての最も良く一致する参照ブロックは、現在のブロックからフルピクセルオフセット、または分数ピクセルオフセットにあり得る。動き情報が、最良の一致が分数ピクセル位置にあることを示すとき、ビデオコードは、参照ブロック中のピクセルから分数位置にあるピクセルを補間することによって、ブロックを再構成し得る。

【 0 0 8 4 】

[0094]図 3 は、分数ピクセル位置の例を例示する。図 3 の例では、幅 3 ピクセルおよび高さ 3 ピクセルである、参照フレームからのブロック 3 0 0 が例示される。整数ピクセル

50

位置は、大文字とともに網掛けのブロックを使用して例示される（例えば、 $A_{0,0}$ 、 $A_{1,0}$ 、 $A_{0,1}$ 、 $A_{-1,0}$ 、 $A_{0,-1}$ 、 $A_{1,1}$ 、 $A_{1,-1}$ 、 $A_{-1,1}$ 、および $A_{-1,-1}$ ）。ここでサブピクセル位置とも呼ばれる $1/4$ ピクセル位置が、小文字を使用して例示される（例えば、 $a_{0,0}$ 、 $b_{0,0}$ 、 $c_{0,0}$ 、等）。この例では、 $a \sim r$ とラベル付けされた（明確さのために、「1」および「0」は省略されていることに留意されたい）、ピクセル位置あたり 15 個のサブピクセル位置が存在する。

【0085】

[0095]様々な方法が、サブピクセル位置についてのサンプルを導出するために使用され得る。例えば、 $1/2$ ピクセル位置 $b_{0,0}$ におけるサンプルは、位置 $A_{0,0}$ および $A_{1,0}$ におけるピクセルを平均することによって生成され得る。別の例として、位置 $A_{-1,0}$ 、 $A_{1,0}$ 、 $A_{0,0}$ 、および隣接ブロックからの $A_{2,0}$ におけるピクセルは、位置 $b_{0,0}$ についてのサンプルを作成するために平均され得る。

10

【0086】

[0096]H E V C では、位置 $a_{0,0}$ 、 $b_{0,0}$ 、および $c_{0,0}$ についての横のサブピクセルサンプル、ならびに位置 $d_{0,0}$ 、 $h_{0,0}$ 、および $n_{0,0}$ についての縦のサブピクセルサンプルは、最も近い整数位置サンプルに 8 タップフィルタを適用することによって導出され得る。位置 $e_{0,0}$ 、 $i_{0,0}$ 、 $p_{0,0}$ 、 $f_{0,0}$ 、 $j_{0,0}$ 、 $q_{0,0}$ 、 $g_{0,0}$ 、 $k_{0,0}$ 、および $r_{0,0}$ についてのサンプルは、位置 $a_{0,i}$ 、 $b_{0,i}$ および $c_{0,i}$ におけるサンプルに 8 タップフィルタを適用することによって導出され得、ここで、 $i = -3 \dots 4$ である。8 タップルーマ補間フィルタの例が、表 1 に例示される。

20

【0087】

【表 1】

表1

位相シフト	係数
0	{ 0, 0, 0, 64, 0, 0, 0, 0 },
1	{ -1, 4, -10, 58, 17, -5, 1, 0 },
2	{ -1, 4, -11, 40, 40, -11, 4, -1 },
3	{ 0, 1, -5, 17, 58, -10, 4, -1 },

【0088】

30

[0097]表 1 は、位相シフトあたり 8 つの係数を含み、各タップにつき 1 つである。各位相シフトは、 $1/4$ ピクセル位置に対応する。例えば、位相シフト 1 は、サブピクセル位置 $a_{0,0}$ および $d_{0,0}$ に対応し得、位相シフト 2 は、サブピクセル位置 $b_{0,0}$ および $h_{0,0}$ に対応し得、以下同様である。他の例では、より長いタップを有する（例えば、より多くの係数を有する）フィルタが適用され得る。より長いタップを有するフィルタは、8 タップフィルタとは異なる特性を有し得、適用時により複雑な計算を必要とし得る。

【0089】

[0098]J E M において開発されているコーデックを含む、他のビデオコーデックでは、 $1/16$ サブピクセル動きベクトル解像度がイネーブルにされ得る。 $1/16$ サブピクセル位置では、16 位相を有するフィルタが、補間のために使用され得る。16 位相を有する例となるフィルタが、表 2 に提供される。しかしながら、J E M では、固定された係数を有する 8 タップフィルタが使用される。

40

【0090】

【表 2】

表2

位相シフト	係数
0	{ 0, 0, 0, 64, 0, 0, 0, 0 },
1	{ 0, 1, -3, 63, 4, -2, 1, 0 },
2	{ -1, 2, -5, 62, 8, -3, 1, 0 },
3	{ -1, 3, -8, 60, 13, -4, 1, 0 },
4	{ -1, 4, -10, 58, 17, -5, 1, 0 },
5	{ -1, 4, -11, 52, 26, -8, 3, -1 },
6	{ -1, 3, -9, 47, 31, -10, 4, -1 },
7	{ -1, 4, -11, 45, 34, -10, 4, -1 },
8	{ -1, 4, -11, 40, 40, -11, 4, -1 },
9	{ -1, 4, -10, 34, 45, -11, 4, -1 },
10	{ -1, 4, -10, 31, 47, -9, 3, -1 },
11	{ -1, 3, -8, 26, 52, -11, 4, -1 },
12	{ 0, 1, -5, 17, 58, -10, 4, -1 },
13	{ 0, 1, -4, 13, 60, -8, 3, -1 },
14	{ 0, 1, -3, 8, 62, -5, 2, -1 },
15	{ 0, 1, -2, 4, 63, -3, 1, 0 }

10

【0091】

20

【0099】図2を参照すると、コンテンツ適応バイナリ算術コーディング(CABAC)は、ビットストリーム中のシンタックス要素を符号化および復号するために、エントロピーコーディング256において使用され得る1つのシステムである。CABACは、要素のコンテキストを使用して、各シンタックス要素についての確率モデルを選択することによって；局所統計量(local statistics)に基づいて、確率推定値を適応すること(adapting)によって；および可変長コーディングの代わりに、算術コーディングを使用して、良好な圧縮パフォーマンスを達成し得る。

【0092】

【0100】CABACは、バイナリ算術コーディングを使用し、これは、バイナリ決定(1または0)のみが符号化されることを意味する。非バイナリ値シンボル(例えば、変換係数または動きベクトル、あるいは2つ以上の可能な値を有する任意のシンボル)が、算術コーディングの前に「2値化される」、すなわちバイナリコードに変換される。その後、2値化されたシンボルの各ビットについて、コンテキストモデルが選択される。コンテキストモデルは、2値化されたシンボルの1つまたは複数のピンについての確率モデルである。コンテキストモデルは、直前にコーディングされたデータシンボルの統計量に基づいて、利用可能なモデルから選ばれる。コンテキストモデルは、各ピンが、「1」または「0」である確率を記憶する。その後、算術コードが、選択された確率モデルに従って、各ピンを符号化する。その後、選択されたコンテキストモデルは、実際のコーディングされた値に基づいて更新される(例えば、ピン値が「1」であった場合、「1」の頻度カウンタが増加される)。

30

40

【0093】

【0101】例として、skip_flag_C[0]、skip_flag_C[1]、およびskip_flag_C[2]と名付けられた3つの候補コンテキストモデルが、シンタックス要素cu_skip_flagをコーディングするために使用され得る。3つの候補から適切なコンテキストを選ぶために、選択値xが、次のように計算され得る：

$$x = (cu_skip_flag[xNbL][yNbL] \&\& availableL) + (cu_skip_flag[xNbA][yNbA] \&\& availableA)$$

【0102】上記式について、使用すべきコンテキストモデルは、以下のように決定され得る：

【0103】(x0, y0)は、現在のピクチャの左上のサンプルに対する(relative to

50

）、現在のルーマブロックの左上のルーマサンプルの位置を指定する。

【0094】

[0104] 位置 (xNbL, yNbL) は、(x0 - 1, y0) に設定され得、変数 availableL は、現在のブロックのすぐ左に位置するブロックの利用可能性を示し得る。

【0095】

[0105] 位置 (xNbA, yNbA) は、(x0, y0 - 1) に等しく設定され得、変数 availableA は、現在のブロックの真上に位置するコーディングブロックの利用可能性を指定し得る。

【0096】

[0106] cu__skip__flag[xNbL][yNbL] および cu__skip__flag[xNbA][yNbA] は、それぞれ、第1のブロック (ブロックL) および第2のブロック (ブロックA) についての cu__skip__flag を表し得る。

【0097】

[0107] 上述されたように、HEVC および JEM によってインプリメントされるコーデックは、サブピクセル動きベクトルがイネーブルにされているとき、固定補間フィルタを使用する。補間フィルタは、任意の所与の状況に対して、最適ではないことがあり得る。例えば、より高いカットオフ周波数を有するフィルタは、コーディングユニットがより多くのディテール (detail) を有するときにより適切であり得、それによって、補間が行われるときにディテールが保存され得る。別の例として、より低いカットオフ周波数を有するフィルタは、処理ユニットがより少ないディテールを有するときにより適切であり得、それによって、コーディングユニットは、より少ないビットでコーディングされ得る。

【0098】

[0108] 動き推定および動き補償は、固定フィルタではなく、最適な補間フィルタを使用することによって、最適化され得る。所与の状況についての最良のフィルタは、高度にローカライズされおよびコンテンツ依存である傾向にある。したがって、最適なフィルタを使用することは、フィルタ係数がエンコーダからデコーダに送信される必要があるという点で、シグナリングオーバーヘッドを著しく増大させ得、および/またはエンコーダの複雑さを増大させ得る。

【0099】

[0109] 様々なインプリメンテーションでは、ビデオコーデックが、異なる予測ユニットについての動き推定および動き補償を行うときに、利用可能な複数の補間フィルタ間で切り替えるように構成され得る。いくつかのケースでは、インターレイヤ予測が、適応的なやり方で補間フィルタを選択することによって改善され得る。様々なインプリメンテーションでは、フィルタ長、カットオフ周波数、遷移帯域、リップルの量などの特性、および/または特性の組合せをそれぞれ有する、N個のフィルタのセットが定義され得る。

【0100】

[0110] f400。また、フィルタのセット410が、図4に例示されており、これは、コーデックによる使用のために定義された全てのフィルタを表す。上述されたように、動き推定444ステップにおいて、参照フレーム464からの参照ブロックが決定される。参照ブロックは、外観という点で、コーディングユニット402に最も良く一致する (例えば、それに最も類似している) 参照フレーム464の一部分である。この例では、コーディングユニット402は、シーケンス、ピクチャ、スライス、コーディングツリーユニット、コーディングユニット、変換ユニット、予測ユニット、またはビデオデータの何らかの他のブロックであり得る。参照ブロックを使用して、動き推定444は、コーディングユニット402についての動き情報406を決定し得る。動き情報406は、特に、参照フレームを識別するために使用され得るインデックスと、1つまたは複数の動きベクトルとを含み得る。

【0101】

[0111] 図4の例では、動き推定444ステップはまた、フィルタ選択基準414を決定

10

20

30

40

50

し得る。フィルタ選択基準 4 1 4 は、特定のコーディングユニット 4 0 2 に適用可能であるフィルタのサブセット 4 1 2（ここでは、フィルタ候補とも呼ばれる）を選択するために使用され得る。例えば、いくつかのインプリメンテーションでは、フィルタ選択基準 4 1 4 は、コーディングユニット 4 0 2 のコーディングレベル（例えば、シーケンスレベル、ピクチャレベル、スライスレベル、コーディングツリーユニットレベル、コーディングユニットレベル、変換ユニットレベル、予測ユニットレベル、等）であり得る。この例では、フィルタのサブセット 4 1 2 は、コーディングユニット 4 0 2 のコーディングレベルに基づいて選択され得る。例えば、サブセットにおける特定のフィルタおよびサブセットにおけるフィルタの数は、異なる処理レベルにわたって変化し得る。

【0102】

[0112]様々なインプリメンテーションでは、動き推定 4 4 4 は、フィルタのサブセット 4 1 2の中から最適な補間フィルタを選択し得る。例えば、動き推定 4 4 4 は、フィルタのサブセット 4 1 2におけるフィルタのうちのどれが最良の圧縮を提供するかを決定するために、レート歪み最適化を使用し得る。様々な他の方法が、最適なフィルタを選択するために使用され得る。

【0103】

[0113]様々なインプリメンテーションでは、選択されたフィルタは、フィルタインデックス 4 0 8を使用して識別され得る。フィルタインデックス 4 0 8 は、フィルタのサブセット 4 1 2の中から選ばれたフィルタを識別し得る。例えば、可能なインデックスは、（例示される例では、それぞれ、F 1、F 3、および F 9 に対して）0、1、または2であり得、フィルタ F 1 が選択されたとき、フィルタインデックス 4 0 8 は、「0」になり得る。いくつかのケースでは、フィルタインデックス 4 0 8 は、動き情報 4 0 6とともに、出力ビットストリーム 4 2 0中に含まれ得る。これらのインプリメンテーションでは、フィルタインデックス 4 0 8 は、ビットストリーム 4 2 0中で明示的にシグナリングされると言える。

【0104】

[0114]いくつかのケースでは、フィルタインデックス 4 0 8 は、暗黙的にシグナリングされ得、すなわち、フィルタインデックス 4 0 8 は、ビットストリーム 4 2 0中に含まれず、代わりに、ビットストリーム 4 2 0を復号するときに、デコーダによって導出され得る。例えば、いくつかのケースでは、フィルタ選択基準 4 1 4 は、1つのフィルタのみを含むフィルタのサブセット 4 1 2をもたらし。別の例として、いくつかのケースでは、フィルタのサブセット 4 1 2 は、複数のフィルタを含み得るが、動き推定 4 4 4 は、デフォルトフィルタを使用することを決定する（例えば、時間的マージ候補については、フィルタ F 1 が常に選択される）。他の例では、コーディングレベル、予測モード、および/または使用されているコーディングツールなどの、1つまたは複数のファクタ（factors）が、適切なフィルタを導出するために、デコーダによって使用され得る。

【0105】

[0115]いくつかのインプリメンテーションでは、動き推定 4 4 4 は、以前にコーディングされたブロックから導出された統計量に基づいて、フィルタを選択し得る。例えば、以前にコーディングされたブロックにおいて、フィルタのサブセット 4 1 2からのいくつかのフィルタは、乏しいパフォーマンスを有していることにより、一度も使用されたことがない可能性がある。この例では、動き推定 4 4 4 は、現在の特定のコーディングユニット 4 0 2のためのフィルタを選択するときに、パフォーマンスの悪いフィルタを除去し得る。逆に、特定のフィルタが、以前にコーディングされたブロックにおいて非常に頻繁に使用されているとき、動き推定 4 4 4 は、特定のコーディングユニット 4 0 2のためのフィルタを選択するときに、このフィルタを優先し得る。

【0106】

[0116]いくつかのインプリメンテーションでは、動き推定 4 4 4 は、空間的に隣接するものまたは時間的に隣接するものなどの、隣接ブロックからフィルタを選択し得る。例えば、いくつかのケースでは、コーディングユニット 4 0 2 は、隣接ブロックから動きベク

10

20

30

40

50

トルをコピーし得る。この例では、コーディングユニット 4 0 2 はまた、同じ隣接ブロックからフィルタインデックスを借り得る。別の例として、いくつかの隣接ブロックが同じフィルタを使用するとき、動き推定 4 4 4 は、コーディングユニット 4 0 2 が同じフィルタを使用すべきであると決定し得る。

【 0 1 0 7 】

[0117]いくつかのインプリメンテーションでは、フィルタ選択基準 4 1 4 は、様々なファクタに基づき得る。例えば、フィルタのサブセット 4 1 2 におけるフィルタは、現在のブロックのために使用される予測モード（例えば、インター予測、イントラ予測、等）、動きベクトルの解像度または異なるもの、使用されているコーディングツール（例えば、双方向オプティカルフロー）、および/またはオーバーラップブロック動き補償（overlapped block motion compensation）におけるピクセルに基づいて選択され得る。別の例として、フィルタのサブセット 4 1 2 におけるフィルタは、現在のブロックのサイズおよび/または形状、現在のフレームの解像度、現在のフレームと参照フレームとの間の距離間の距離、および/または参照ピクチャについての量子化パラメータ（QP）値に基づいて選択され得る。

【 0 1 0 8 】

[0118]例として、コーディングユニット 4 0 2 が、フレームまたはスライスなどのデータの大きいセットであるか、またはコーディングユニット 4 0 2 が、ディテールをほとんど有していないとき、フィルタのサブセット 4 1 2 におけるフィルタは、低いカットオフを有するフィルタであり得る。低いカットオフ周波数を有するフィルタは、ディテールを滑らかにする傾向があり、したがって、コーディングユニット 4 0 2 がディテールをほとんど有していないとき、低いカットオフ周波数を有するフィルタを使用することによってデータが失われることはない。別の例として、コーディングユニット 4 0 2 が、コーディングユニットなどのデータの小さいセットであるか、またはコーディングユニット 4 0 2 が、高度のディテール（high degree of detail）を有するとき、フィルタのサブセット 4 1 2 は、高いカットオフ周波数を有するフィルタを含み得る。高いカットオフ周波数を有するフィルタは、コーディングユニット 4 0 2 における高い周波数成分（high frequency components）を維持し得、したがって、ディテールを保存する。

【 0 1 0 9 】

[0119]いくつかのインプリメンテーションでは、追加のフィルタ候補が、フィルタのサブセット 4 1 2 からのフィルタをマージすることによって生成され得る。例えば、2つのフィルタ候補が与えられると、第1の候補フィルタからの偶数の位相シフトが、第2の候補からの奇数の位相シフトとマージされて、第3の候補が生成される。別の例として、第4の候補が、第1の候補からの奇数の位相シフトを、第2の候補からの偶数の位相シフトと組み合わせることによって構成され得る。

【 0 1 1 0 】

[0120]いくつかのインプリメンテーションでは、複数の補間フィルタが結合され得る。例えば、整数サンプルのための補間フィルタが、分数サンプルのための補間フィルタへとカウントされ得る。この例では、整数サンプルのためのフィルタは、分数サンプルのためのフィルタからの最小の非ゼロ位相シフトを有するフィルタのものよりも高いカットオフ周波数を有し得る。代替として、整数サンプルのためのフィルタは、分数サンプルのためのフィルタのものよりも低いカットオフ周波数を有し得る。

【 0 1 1 1 】

[0121]いくつかのインプリメンテーションでは、コーディングツールがまた、フィルタに関連付けられる。例えば、双方向オプティカルフローは、勾配フィルタを使用し得、フレームレートアップコンバージョンは、リファインメントフィルタ（refinement filter）を使用し得る。いくつかのインプリメンテーションでは、ツール固有のフィルタが、フィルタのサブセット 4 1 2 における各フィルタ候補に従って生成され得る。例えば、3つの補間フィルタ候補が与えられると、3つの勾配フィルタが、双方向オプティカルフローのために導出され得る。これらのインプリメンテーションでは、ツール固有のフィルタを

導出することによって、ツール固有のフィルタは、出力ビットストリーム 4 2 0 中で記述される必要がない。

【0 1 1 2】

[0122]いくつかの例では、フィルタが整数ピクセル位置と分数ピクセル位置との両方に適用されるとき、整数ピクセル位置のためのフィルタが、最初に適用され得る。その後、フィルタされた整数サンプルが、さらなる補間のための入力として使用されて、分数ピクセルサンプルを導出し得る。

【0 1 1 3】

[0123]いくつかの例では、フィルタが整数ピクセル位置と分数ピクセル位置との両方に適用されるとき、整数ピクセル位置のためのフィルタは、関連する動きベクトルが整数の精度を有するときのみ適用可能であり得る。これらの例では、分数ピクセルサンプルは、整数位置におけるフィルタされていないピクセルを入力として使用して導出され得る。

【0 1 1 4】

[0124]図 5 は、復号プロセス 5 0 0 の動き推定 5 4 4 ステップの例を例示する。また、フィルタのセット 5 1 0 が、図 5 に例示されており、これは、コーデックによる使用のために定義された全てのフィルタを表す。デコーダプロセス 5 0 0 では、動き推定 5 4 4 は、予測ユニット 5 0 2 を再構成または予測するために使用され得る。現在のフレーム中の予測ユニット 5 0 2 について、符号化されたビットストリーム 5 2 0 は、動き情報 5 0 6 を含み得る。動き情報 5 0 6 は、例えば、参照フレーム 5 6 4 に対するインデックスと、1 つまたは複数の動きベクトルとを含み得る。動き情報 5 0 6 を使用して、動き推定 5 4 4 は、参照フレーム 5 6 4 中の参照ブロックを識別し、予測ユニット 5 0 2 を予測または生成し得る。

【0 1 1 5】

[0125]いくつかのインプリメンテーションでは、動き推定 5 4 4 はまた、フィルタ選択基準 5 1 4 を決定し得る。フィルタ選択基準 5 1 4 は、予測ユニット 5 0 2 に適用可能であるフィルタのサブセット 5 1 2 を選択するために使用され得る。例えば、フィルタ選択基準 5 1 4 は、予測ユニット 5 0 2 のコーディングレベル、予測モード、および / または使用されているコーディングツールに基づき得る。サブセットにおける特定のフィルタおよびサブセットにおけるフィルタの数は、ビットストリーム 5 2 0 が復号され、異なる状況に遭遇するにつれて変化し得る。

【0 1 1 6】

[0126]いくつかのケースでは、ビットストリーム 5 2 0 は、フィルタのサブセット 5 1 2 からフィルタを選択するために使用され得る明示的なフィルタインデックス 5 0 8 を含む。様々なインプリメンテーションでは、フィルタインデックス 5 0 8 は、動き情報 5 0 6 に関連付けられ得る。フィルタインデックス 5 0 8 がビットストリーム 5 2 0 中に含まれるとき、フィルタインデックス 5 0 8 は、明示的にシグナリングされると言える。フィルタインデックス 5 0 8 は、補間のために使用されるべきフィルタのサブセット 5 1 2 の中からのフィルタを識別し得る。例えば、可能なインデックスは、(例示される例では、それぞれ、F 1、F 3、および F 9 に対して) 0、1、または 2 であり得、フィルタ F 1 が選択されたとき、フィルタインデックス 5 0 8 は、「0」になり得る。

【0 1 1 7】

[0127]いくつかのケースでは、デコーダプロセス 5 0 0 は、フィルタインデックス 5 0 8 を導出し得る。例えば、いくつかのケースでは、フィルタ選択基準 5 1 4 は、1 つのフィルタのみを含むフィルタのサブセット 5 1 2 をもたらす。別の例として、いくつかのケースでは、フィルタのサブセット 5 1 2 は、複数のフィルタを含み得るが、動き推定 5 4 4 は、デフォルトフィルタを使用することを決定する(例えば、時間的マージ候補については、フィルタ F 1 が常に選択される)。他の例では、コーディングレベル、予測モード、および / または使用されているコーディングツールなどの、1 つまたは複数のファクタが、適切なフィルタを導出するために、デコーダによって使用され得る。デコーダは、エンコーダによって決定されたものと同じフィルタを使用すべきであるので、デコーダがフ

10

20

30

40

50

フィルタの識別を決定論的に (deterministically) 導出できないケースでは、フィルタの識別は、ビットストリーム 520 中でシグナリングされることになる。

【0118】

[0128]様々な方法が、フィルタインデックスが暗黙的であるときに使用すべき補間フィルタを決定するために使用され得る。いくつかの例では、補間フィルタは、現在のブロックの前に符号化された他のブロックによって決定され得る。例えば、ある特定の条件を満たすブロックは、他のブロックのために選ばれたフィルタを推論または継承することによって、フィルタを決定し得る。例えば、現在の動きベクトルが、(例えば、空間的マージ候補、アフィンマージ候補、および他の候補でのケースのように)以前にコーディングされたブロックから導出されるとき、動きベクトルがそれから導出されているブロックのために選ばれた補間フィルタが、現在のブロックのために使用され得る。この例では、いったんデコーダが動きベクトルを決定すると、デコーダは、いかなる追加の情報も必要とせず、現在のブロックのためのフィルタを導出し得るので、フィルタインデックスの明示的なシグナリングは、必要とされない場合がある。

10

【0119】

[0129]いくつかの例では、ある特定のブロックのための補間フィルタは、所定のデフォルトフィルタを常に選ぶか、または時に選び得る。これらの例では、別のブロックからのフィルタの継承またはフィルタを探索することは、必要ではない。いくつかのケースでは、いったん決定されると、デフォルトフィルタは、全てのコーディングレベルで適用される。いくつかの例では、フィルタのサブセットは、特定のフィルタの選択に偏り得る。例えば、フィルタは、シーケンシャル順にテストされ得、特定のフィルタが、フレームレートアップコンバージョン、時間的動きベクトル予測 (TMVP)、アドバンスド動きベクトル予測 (AMVP)、またはゼロ動きベクトルマージ候補から動きベクトルを導出する、ブロックのための第1の候補として設定され得る。いくつかの例では、デフォルトフィルタは、フレームおよび/またはスライスにわたって変化し得る。例えば、以前の(1つまたは複数の)ピクチャおよび/または(1つまたは複数の)スライスにおいて最も頻繁に選ばれたフィルタが、現在のピクチャまたはスライスに属する全てのブロックのためのデフォルトフィルタとして設定され得る。デコーダがエンコーダからのいかなるサイド情報もなしに同じプロシージャを模倣し得るので、フィルタインデックスの明示的なシグナリングは、必要とされない場合がある。

20

30

【0120】

[0130]フィルタインデックスは、そうでない場合、必要に応じて、ビットストリーム中に明示的に含まれ得る。いくつかの例では、「`interpolation_filter_index`」と呼ばれ得るシンタックス要素が、暗黙的にフィルタを選ぶことができないブロックのために選ばれた補間フィルタを示すために、エンコーダによってデコーダにシグナリングされ得る。シグナリングによってとは、`interpolation_filter_index`が、デコーダによって読み取られるように、ビットストリーム中に含まれ得ることを意味する。

【0121】

[0131]いくつかの例では、`interpolation_filter_index`の値は、固定長コード、指数ゴロムコード、単項コード、または短縮された単項コードなどの、2値化コードを用いて2値化され得る。例えば、フィルタのサブセットにおいて3つのフィルタが存在するとき、短縮された単項コードが、`interpolation_filter_index`を示すために使用され得る。この例では、2つのフラグ(例えば、`FLAG0`および`FLAG1`)が、サブセットにおける3つのフィルタについての3つの可能なインデックス0、1、および2をコーディングするために使用され得る。例えば、3つのインデックスは、それぞれ、(0, N/A)、(1, 0)、および(1, 1)としてコーディングされ得る。この例では、1つのフラグ(`FLAG0`)のみが、`interpolation_filter_index = 0`を示すために使用され得る。

40

【0122】

50

[0132]いくつかの例では、ある特定のコンテキストモデルが、`interpolation__filter__index`に関連付けられ得、それは、`interpolation__filter__index`を符号化するときに使用され得る。例えば、 X 個のコンテキストモデルが、`interpolation__filter__index`に関連付けられ得、例えば、`interpolation__filter__index__C[0]`、`interpolation__filter__index__C[1]`、 \dots 、`interpolation__filter__index__C[X-1]`と名付けられる。この例では、これらのコンテキストモデルの中からインデックス x を有するコンテキストモデル、`interpolation__filter__index__C[x]`が、現在のブロックについての`interpolation__filter__index`をコーディングするためのコンテキストとして選ばれ得る。

10

【0123】

[0133]いくつかの例では、`interpolation__filter__index`が、2つのフラグ（例えば、`FLAG0`および`FLAG1`）を使用して表され得るとき、各フラグに関連付けられたある特定の数のコンテキストモデルが存在し得る。例えば、`FLAG0`に関連付けられた Y 個のコンテキストモデルが存在し得、`interpolation__filter__index__C[0]`、`interpolation__filter__index__C[1]`、 \dots 、`interpolation__filter__index__C[Y-1]`と名付けられる。この例では、`FLAG0`が設定されるとき、インデックス x を有するコンテキストモデル、`interpolation__filter__index__C[x]`が、現在のブロックの`interpolation__filter__index`をコーディングするためのコンテキストとして選ばれ得る。この例では、`FLAG1`は、設定されていないか、または重要でない場合がある。別の例として、`FLAG1`に関連付けられた Z 個のコンテキストモデルが存在し得、`interpolation__filter__index__C[0]`、`interpolation__filter__index__C[1]`、 \dots 、`interpolation__filter__index__C[Z-1]`と名付けられる。この例では、`FLAG1`が設定されるとき、インデックス x を有するコンテキストモデル、`interpolation__filter__index__C[x]`が、現在のブロックの`interpolation__filter__index`をコーディングするためのコンテキストとして選ばれる。この例では、`FLAG0`は、設定されていないか、または重要でない場合がある。

20

30

【0124】

[0134]上記の例では、 x は、様々な基準に基づいて選択され得る。例えば、 x は、現在のブロックのサイズ、現在のブロックの形状、現在のフレームの解像度、インター予測方向、現在のブロックについての（1つまたは複数の）動きベクトル、現在のブロックについての動きベクトルの差分、現在のブロックのための参照フレーム、現在のブロックに対する隣接ブロックの（1つまたは複数の）`interpolation__filter__index`、何らかの他の基準、または基準の組合せに基づき得る。様々な例では、関数が x の値を生成するために使用され得、ここで、関数は、上記の例となる基準のうちの1つまたは複数を入力として受け取り得る。いくつかの例では、異なる関数が、異なるフレーム解像度のために使用され得る。

40

【0125】

[0135]いくつかの例では、 x が現在のブロックのサイズに基づくとき、しきい値 P が、 x の値を決定するために使用され得る。例えば、現在のブロックのサイズが P 未満であるとき、 x は0になり得、それ以外の場合、 x は1になり得る。しきい値 P は、整数であり得る。現在のブロックのサイズは、 $M \times N$ として計算され得、ここで、 M および N は、ピクセル単位での、現在のブロックの幅および高さである。

【0126】

[0136]いくつかの例では、 x が現在のブロックのサイズに基づくとき、 x の値は、整数配列 $\{a_k\}$ から決定され得、ここで、配列は、昇順にソートされる。例えば、 $a_{k-1} <$

50

$= S < a_K$ のとき、 x は K に等しくなり得、ここで、 S は、現在のブロックのサイズである。この例では、 S は、 $M \times N$ として計算される、現在のブロック中のピクセルの数であり得、ここで、 M および N は、ピクセル単位での、現在のブロックの幅および高さである。

【0127】

[0137]いくつかの例では、 x が現在のブロックの形状に基づくとき、 x の値は、ブロックの高さと幅との間の関係から決定され得る。例えば、現在のブロックについての高さ M および幅 N が与えられると、 $M < N$ のとき、 x は 0 に等しくなり得、 $M = N$ のとき、 x は 1 に等しくなり得、 $M > N$ のとき、 x は 2 に等しくなり得る。

【0128】

[0138]いくつかの例では、 x が現在のブロックの予測方向に基づくとき、 x の値は、予測方向から決定され得る。例えば、現在のブロックが、単インター予測を用いてコーディングされるとき、 x は 0 に等しくなり得、現在のブロックが、双インター予測を用いてコーディングされるとき、 x は 1 に等しくなり得る。別の例として、 x の値は、値 $InterDir$ に基づき得、ここで、現在のブロックが参照リスト 0 からの単インター予測を用いてコーディングされるとき、 $InterDir$ は 1 であり、現在のブロックが参照リスト 1 からの単インター予測を用いてコーディングされるとき、 $InterDir$ は 2 であり、現在のブロックが双インター予測を用いてコーディングされるとき、 $InterDir$ は 3 である。この例では、 x は、 $InterDir - 1$ に等しくなり得る。

【0129】

[0139]いくつかの例では、 x が隣接ブロックの `interpolation__filter__index` に基づくとき、 x の値は、1 つまたは複数の隣接ブロックの `interpolation__filter__index` から決定され得る。例えば、 x の値は、関数 $f(left_idx, top_idx)$ によって決定され得、ここで、`left__idx` は、左の隣接ブロックの `interpolation__filter__index` であり、`top__idx` は、上の隣接ブロックの `interpolation__filter__index` である。関数 f は、任意の関数であり得る。いくつかの例では、異なる関数が、異なるフレーム解像度のために使用され得る。代替または追加として、 x の値は、 $a + b$ に等しくなり得、ここで、左の隣接ブロックの `interpolation__filter__index` が利用可能でありかつゼロよりも大きいとき、 a は 1 に等しく（そうでない場合、 a は 0 に等しい）、また、上の隣接ブロックの `interpolation__filter__index` が利用可能でありかつゼロよりも大きい場合、 b は 1 に等しい（そうでない場合、 b は 0 に等しい）。

【0130】

[0140]いくつかの例では、ファクタの組合せが、 x の値を決定するときに考慮され得る。例えば、現在のブロックのサイズは、隣接ブロックから `interpolation__filter__index` 値を選択するために使用され得る。例えば、 x は、現在のピクチャのサイズがしきい値 P 未満であるとき、 $f(left_idx, top_idx)$ の結果に等しくなり得、ここで、 P は、整数であり、現在のブロックのサイズは、計算された $M \times N$ であり、ここで、 M および N は、ピクセル単位での現在のブロックの幅および高さである。現在のブロックのサイズが P 以上であるとき、 x は、関数 $g(left_idx, top_idx)$ の出力に等しくなり得る。

【0131】

[0141]補間フィルタを切り替える例となるアプリケーションが、ここで説明される。上述されたように、最良のフィルタが、異なる特性を有し得る N 個のフィルタの中から、各ブロックのために選ばれ得る。最良のフィルタは、例えば、レート歪み最適化、または何らかの他の選択メカニズムを使用して選択され得る。

【0132】

[0142]エンコードの複雑さを低減させるために、最良のフィルタは、既にコーディングされているブロックから継承されるか、以前の決定から再利用されるか、または全ての利用可能なフィルタのパフォーマンスを評価することなく固定であるか、のいずれかであり

10

20

30

40

50

得る。シグナリングオーバーヘッドを低減させるために、フィルタの固定セットが前もって決定され、エンコーダは、フィルタ係数の代わりに、フィルタインデックスをデコーダにシグナリングし得る。

【 0 1 3 3 】

[0143]以下の例は、補間フィルタの切り替えのアプリケーションを実証するために、H M 1 6 . 6 ベースの J E M 3 . 0 を使用する。J E M は、国際電気通信連合ビデオコーディングエキスパートグループ (I T U - V C E G) および国際標準化機構 / 国際電気標準会議モーションピクチャエキスパートグループ (I S O / I E C M P E G) のプロジェクトである。J E M は、H E V C よりも良い圧縮能力を有するビデオコーディング技術を研究および潜在的に標準化するためのプラットフォームを提供する参照ソフトウェアである。

10

【 0 1 3 4 】

[0144]以下の例では、コーディングユニットは、J E M 3 . 0 上でインプリメントされる Q T B T 構造において定義されるブロックである。他の例では、コーディングユニットは、Q T B T ブロックの集合、C T U、スライス、ピクチャ、シーケンス、または別の好適なコーディングユニットなどの、より大きい構造であり得る。この例は、ルーマ成分のための補間フィルタのみを示しており、補間フィルタの切り替えは、クロマ成分のみならず、整数サンプルにも拡張され得ることに留意されたい。

【 0 1 3 5 】

[0145]この例では、補間フィルタの候補セットは、異なる特性を有する 3 つのフィルタ : 1 つの 1 2 タップフィルタ (F 0) と、2 つの 8 タップフィルタ (F 1 および F 2) とを含む。フィルタ F 0 が、表 3 に例示され、フィルタ F 1 が、表 4 に例示され、フィルタ F 2 が、表 5 に例示される。この例では、フィルタ候補のセットは、異なるピクチャおよび / またはスライスにわたって変化しない。J E M 3 . 0 は、最大で 1 / 1 6 までの動きベクトル解像度をサポートするので、各フィルタ候補は、以下の表に示されるように、1 6 個の異なる位相シフトで構成される。

20

【 0 1 3 6 】

30

40

50

【表 3】

表3

位相シフト	係数(F0)
0	{ 0, 0, 0, 0, 0, 0, 256, 0, 0, 0, 0, 0, 0, 0 },
1	{ -1, 2, -4, 7, -14, 254, 16, -7, 4, -2, 1, 0 },
2	{ -1, 4, -7, 12, -27, 249, 35, -15, 8, -4, 2, 0 },
3	{ -2, 5, -10, 17, -36, 241, 54, -22, 12, -6, 3, 0 },
4	{ -2, 6, -12, 21, -44, 232, 75, -30, 16, -9, 4, -1 },
5	{ -2, 7, -13, 24, -49, 215, 97, -36, 19, -10, 5, -1 },
6	{ -2, 7, -14, 26, -51, 198, 119, -42, 22, -12, 6, -1 },
7	{ -2, 7, -14, 26, -52, 183, 140, -47, 24, -13, 6, -2 },
8	{ -2, 7, -14, 26, -50, 161, 161, -50, 26, -14, 7, -2 },
9	{ -2, 6, -13, 24, -47, 140, 183, -52, 26, -14, 7, -2 },
10	{ -1, 6, -12, 22, -42, 119, 198, -51, 26, -14, 7, -2 },
11	{ -1, 5, -10, 19, -36, 97, 215, -49, 24, -13, 7, -2 },
12	{ -1, 4, -9, 16, -30, 75, 232, -44, 21, -12, 6, -2 },
13	{ 0, 3, -6, 12, -22, 54, 241, -36, 17, -10, 5, -2 },
14	{ 0, 2, -4, 8, -15, 35, 249, -27, 12, -7, 4, -1 },
15	{ 0, 1, -2, 4, -7, 16, 254, -14, 7, -4, 2, -1 },

【 0 1 3 7 】

10

20

30

40

50

【表 4】

表4

位相シフト	係数(F1)
0	{ 0, 0, 0, 256, 0, 0, 0, 0 },
1	{ 2, -9, 21, 185, 80, -27, 4, 0 },
2	{ 2, -9, 15, 181, 90, -27, 3, 1 },
3	{ 2, -8, 9, 176, 100, -26, 2, 1 },
4	{ 2, -7, 4, 171, 108, -24, 1, 1 },
5	{ 2, -6, -1, 165, 117, -22, 0, 1 },
6	{ 2, -5, -6, 158, 127, -20, -1, 1 },
7	{ 2, -4, -10, 151, 134, -17, -2, 2 },
8	{ 2, -3, -14, 143, 143, -14, -3, 2 },
9	{ 2, -2, -17, 134, 151, -10, -4, 2 },
10	{ 1, -1, -20, 127, 158, -6, -5, 2 },
11	{ 1, 0, -22, 117, 165, -1, -6, 2 },
12	{ 1, 1, -24, 108, 171, 4, -7, 2 },
13	{ 1, 2, -26, 100, 176, 9, -8, 2 },
14	{ 1, 3, -27, 90, 181, 15, -9, 2 },
15	{ 0, 4, -27, 80, 185, 21, -9, 2 },

10

【 0 1 3 8 】

20

【表 5】

表5

位相シフト	係数(F2)
0	{ 0, 0, 0, 256, 0, 0, 0, 0 },
1	{ 1, 15, 61, 97, 70, 17, -3, -2 },
2	{ 0, 13, 58, 96, 73, 20, -2, -2 },
3	{ 0, 11, 55, 95, 75, 23, -1, -2 },
4	{ 0, 10, 51, 94, 77, 26, 0, -2 },
5	{ 0, 8, 48, 92, 80, 29, 1, -2 },
6	{ -1, 7, 45, 90, 82, 32, 2, -1 },
7	{ -1, 5, 41, 88, 86, 35, 3, -1 },
8	{ -1, 4, 38, 87, 87, 38, 4, -1 },
9	{ -1, 3, 35, 86, 88, 41, 5, -1 },
10	{ -1, 2, 32, 82, 90, 45, 7, -1 },
11	{ -2, 1, 29, 80, 92, 48, 8, 0 },
12	{ -2, 0, 26, 77, 94, 51, 10, 0 },
13	{ -2, -1, 23, 75, 95, 55, 11, 0 },
14	{ -2, -2, 20, 73, 96, 58, 13, 0 },
15	{ -2, -3, 17, 70, 97, 61, 15, 1 },

30

【 0 1 3 9 】

40

[0146]図 6 は、3つの例となるフィルタ、F 0、F 1、および F 2 の周波数応答を例示するグラフである。各曲線は、各フィルタについて 8 つの同じ位相シフトに対応する。この例によって例示されるように、フィルタ F 0 は、高いカットオフ周波数を有し、一方、フィルタ F 2 は、低いカットオフ周波数を有する。フィルタ F 1 は、極めてニュートラルなカットオフ周波数を有する。

【 0 1 4 0 】

[0147]この例を続けると、所与のコーディングユニットのために、フィルタ F 0、F 1、および F 2 のうちの 1 つが選ばれ得る。選択されたフィルタは、暗黙的または明示的にシグナリングされ得る。

【 0 1 4 1 】

50

[0148]この例では、デコーダが、エンコーダからのいかなるサイド情報もなしに、エンコーダがマージ動きベクトル候補を決定するために実行するのと同じプロシーダを模倣し得るので、フィルタインデックスの暗黙的なシグナリングが、マージモードのためにイネーブルにされる。例えば、コーディングユニットのための補間フィルタは、現在の動きの導出元のブロックからフィルタインデックスを継承するか、あるいは、この例では、フィルタF0である、デフォルトフィルタを選択するかのいずれかによって決定され得る。

【0142】

[0149]この例を続けると、フィルタ継承は、現在の動きが、空間的マージ候補、組み合わせられている2つの候補が同一のフィルタを使用するときの組み合わせられたマージ候補、およびアフィンマージ候補から導出されるときに使用され得る。これらのケースの各々では、継承元のブロックのコンテンツが、フィルタの選択に影響を及ぼし得る。また、この例では、現在の動きベクトルが、フレームレートアップコンバージョンマージ候補または時間的マージ候補（例えば、TMVP、AMVP、STMVP、およびゼロ動きベクトル）候補から導出されるとき、デフォルトフィルタ、F0が選択される。これらのケースの各々では、ブロックにおけるディテールを保存するために、高いカットオフ周波数を有するフィルタが望ましくあり得る。

【0143】

[0150]この例を続けると、フィルタインデックスの明示的なシグナリングは、AMVPモードのためにイネーブルである。JEM3.0では、2つの追加のレート-歪み最適化がエンコーダに含まれ、これは、3つの例となるフィルタ候補から最良のフィルタを選ぶために使用され得る。追加の最適化プロセスは、整数動きベクトルを再利用すること、または特定の構成において選ばれたフィルタを再利用することによって、追加のプロセスをスキップすることによって、単純化され得る。一例として、照度補償(IC)ツールの場合、照度補償がオフであるときに選ばれたフィルタは、照度補償がオンであるときのケースにおいて再利用され得る。これは、符号化の時間を低減させ得る。

【0144】

[0151]この例を続けると、フィルタインデックスの明示的なシグナリングがイネーブルにされるとき、`interpolation_filter_index`は、F0については「0」、F1については「10」、およびF2については「11」を使用して、2値化され得る。以下のように、2つのフラグFLAG0およびFLAG1が、`interpolation_filter_index`をコーディングするために使用され得る：

[0152] FLAG0

[0153] if (FLAG0)

[0154] FLAG1

[0155]上記の例を続けると、2つのコンテキストが、各フラグのために提供され得る：FLAG0については、(FLAG0_ctx0, FLAG0_ctx1)、およびFLAG1については、(FLAG1_ctx0, FLAG1_ctx1)。例として、現在のブロックが単予測を用いてコーディングされるとき、FLAG0_ctx0は、FLAG0をコーディングするために使用され得、FLAG1_ctx0は、FLAG1をコーディングするために使用され得る。別の例として、現在のブロックが双予測を用いてコーディングされるとき、FLAG0_ctx1は、FLAG0をコーディングするために使用され得、FLAG1_ctx1は、FLAG1をコーディングするために使用され得る。

【0145】

[0156]2つの補間候補、F0およびF2を符号化する別の例では、`interpolation_filter_index`は、F0については「0」およびF1については「1」として2値化され得る。この例では、1つのフラグが、`interpolation_filter_index`をコーディングするために使用され得る。各フラグは、関連するコンテキストFLAG0_ctx0およびFLAG0_ctx1を有し得る。現在のブロックのサイズが(この例の目的のために)256未満であるとき、FLAG0_ctx0が、FLAG0をコーディングするために使用され得る。そうでない場合、サイズが

256以上であるとき、FLAG0__ctx1が、FLAG0をコーディングするために使用され得る。現在のブロックのサイズは、 $w \times h$ として計算され、ここで、 w および h は、それぞれ、ピクセル単位で現在のブロックの幅および高さを表す。

【0146】

[0157]図7は、符号化プロセス中に補間フィルタを切り替えるためのプロセス700の例を例示する。702において、プロセス700は、ビデオデータを取得することを含む。ビデオデータは、例えば、カメラのようなビデオキャプチャデバイスから取得され得る。代替または追加として、ビデオデータは、記憶された位置から読み取られ得る。

【0147】

[0158]704において、プロセスは、コーディングユニットを符号化することを含み、
10
ここにおいて、処理を符号化することは、コーディングユニットについての動き推定および動き補償のために補間フィルタを選択することを含み、ここにおいて、補間フィルタは、補間フィルタのセットから選択される。いくつかのインプリメンテーションでは、補間フィルタは、補間フィルタのセットから選択された、補間フィルタのサブセットから選択される。いくつかのインプリメンテーションでは、補間フィルタのサブセットは、コーディングユニットのコーディングレベルに基づいて決定される。いくつかのインプリメンテーションでは、サブセットは、コーディングユニットの予測モードに基づいて決定される。いくつかのインプリメンテーションでは、サブセットは、動き推定の結果に基づいて決定される。いくつかのインプリメンテーションでは、サブセットは、コーディングユニットをコーディングするために使用されるコーディングツールに基づいて決定される。
20

【0148】

[0159]いくつかのインプリメンテーションでは、補間フィルタは、以前にコーディングされたブロックから選択される。いくつかのインプリメンテーションでは、補間フィルタは、隣接ブロックから選択される。

【0149】

[0160]いくつかのインプリメンテーションでは、補間フィルタは、別の補間フィルタとマージされ、結果として得られるマージされた補間フィルタは、動き推定のために使用される。

【0150】

[0161]いくつかのインプリメンテーションでは、補間フィルタは、第2の補間フィルタに結合される。これらのインプリメンテーションでは、補間フィルタは、整数ピクセル位置のためのものであり得、第2の補間フィルタは、分数ピクセル位置のためのものであり得る。いくつかのインプリメンテーションでは、補間フィルタと第2の補間フィルタは、異なるカットオフ周波数を有し得る。
30

【0151】

[0162]706において、プロセス700は、符号化されたビデオビットストリームを生成することを含み、ここにおいて、ビデオビットストリームは、符号化されたコーディングユニットを含む。いくつかのインプリメンテーションでは、符号化されたビデオビットストリームは、フィルタインデックスのような、選択された補間フィルタを識別する明示的な値を含み得る。いくつかのインプリメンテーションでは、補間フィルタを識別する値は、符号化されたビデオビットストリームから暗示され得る。
40

【0152】

[0163]図8は、復号プロセスにおいて補間フィルタを切り替えるためのプロセス800の例である。802において、プロセス800は、符号化されたビデオビットストリームを取得することを含む。ビットストリームは、例えば、記憶位置についておよび/またはネットワークを介して取得され得る。

【0153】

[0164]804において、プロセス800は、符号化されたビデオビットストリームから現在のフレーム中のコーディングユニットのための参照フレームを決定することを含む。参照フレームは、現在のフレームに時間的に先行または後続し得る。参照フレームは、現
50

在のフレームと同じフレームであり得る。

【0154】

[0165] 806において、プロセス800は、補間フィルタのセットから補間フィルタを決定することを含む。いくつかのインプリメンテーションでは、補間フィルタは、補間フィルタのセットからの補間フィルタのサブセットから選択され得る。いくつかのインプリメンテーションでは、補間フィルタのサブセットは、コーディングユニットのコーディングレベルに基づいて決定され得る。いくつかのインプリメンテーションでは、サブセットは、コーディングユニットのための予測モードに基づいて決定され得る。いくつかのインプリメンテーションでは、サブセットは、動き推定の結果に基づいて決定され得る。いくつかのインプリメンテーションでは、サブセットは、コーディングユニットをコーディングするために使用されるコーディングツールに基づいて決定され得る。

10

【0155】

[0166]いくつかのインプリメンテーションでは、補間フィルタは、以前にコーディングされたブロックから選択され得る。いくつかのインプリメンテーションでは、補間フィルタは、隣接ブロックから選択され得る。

【0156】

[0167]いくつかのインプリメンテーションでは、補間フィルタは、第2の補間フィルタとマージされ得、結果として得られるマージされた補間フィルタが、動き推定のために使用され得る。

【0157】

20

[0168]いくつかのインプリメンテーションでは、補間フィルタは、第2の補間フィルタに結合され得る。これらのインプリメンテーションでは、補間フィルタは、整数ピクセル位置のためのものであり得、第2の補間フィルタは、分数ピクセル位置のためのものであり得る。いくつかのインプリメンテーションでは、補間フィルタと第2の補間フィルタは、異なるカットオフ周波数を有し得る。

【0158】

[0169]いくつかのインプリメンテーションでは、符号化されたビデオビットストリームは、選択された補間フィルタを識別するフィルタインデックスのような、明示的な値を含む。いくつかのインプリメンテーションでは、プロセスは、ビットストリームからのデータを使用して、補間フィルタの識別を導出することを含む。

30

【0159】

[0170] 808において、プロセス800は、コーディングユニットを再構成するために、804で決定された参照フレームと、806において決定された補間フィルタとを使用することを含む。

【0160】

[0171]ここで説明されるコーディング技法は、例となるビデオ符号化および復号システム（例えば、システム100）においてインプリメントされ得る。いくつかの例では、システムが、宛先デバイスによって後の時間に復号されることになる符号化されたビデオデータを提供するソースデバイスを含む。特に、ソースデバイスは、コンピュータ可読媒体を介して、宛先デバイスにビデオデータを提供する。ソースデバイスおよび宛先デバイスは、デスクトップコンピュータ、ノートブック（すなわち、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」フォンなどの電話ハンドセット、いわゆる「スマート」パッド、テレビ、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、ビデオストリーミングデバイス、または同様のものを含む、幅広い範囲のデバイスのうちの任意のものを備え得る。いくつかのケースでは、ソースデバイスおよび宛先デバイスは、ワイヤレス通信のために装備され得る。

40

【0161】

[0172]宛先デバイスは、コンピュータ可読媒体を介して、復号されることになる符号化されたビデオデータを受信し得る。コンピュータ可読媒体は、ソースデバイスから宛先デ

50

バイスに、符号化されたビデオデータを移動させることが可能な任意のタイプの媒体またはデバイスを備え得る。一例では、コンピュータ可読媒体は、ソースデバイスがリアルタイムで宛先デバイスに、符号化されたビデオデータを直接送信することを可能にする通信媒体を備え得る。符号化されたビデオデータは、ワイヤレス通信プロトコルなどの通信規格に従って変調され、宛先デバイスに送信され得る。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などの、任意のワイヤレスまたはワイヤード通信媒体を備え得る。通信媒体は、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のようなグローバルネットワークなどの、パケットベースのネットワークの一部を形成し得る。通信媒体は、ルータ、スイッチ、基地局、またはソースデバイスから宛先デバイスへの通信を容易にするのに有用であり得るその他任意の機器を含み得る。

10

【0162】

[0173]いくつかの例では、符号化されたデータは、出力インターフェースから記憶デバイスに出力され得る。同様に、符号化されたデータは、入力インターフェースによって、記憶デバイスからアクセスされ得る。記憶デバイスは、ハードドライブ、ブルーレイディスク、DVD、CD-ROM、フラッシュメモリ、揮発性または不揮発性メモリ、または符号化されたビデオデータを記憶するためのその他任意の好適なデジタル記憶媒体などの、様々な分散されたまたはローカルにアクセスされるデータ記憶媒体のうちの任意のものを含み得る。さらなる例では、記憶デバイスは、ファイルサーバ、またはソースデバイスによって生成される符号化されたビデオを記憶し得る別の中間記憶デバイスに対応し得る。宛先デバイスは、ストリーミングまたはダウンロードを介して、記憶デバイスからの記憶されたビデオデータにアクセスし得る。ファイルサーバは、符号化されたビデオデータを記憶し、宛先デバイスにその符号化されたビデオデータを送信することが可能な任意のタイプのサーバであり得る。例となるファイルサーバは、（例えば、ウェブサイト用の）ウェブサーバ、FTPサーバ、ネットワーク接続記憶（NAS）デバイス、またはローカルディスクドライブを含む。宛先デバイスは、インターネット接続を含む、任意の標準的なデータ接続を通じて、符号化されたビデオデータにアクセスし得る。これは、ワイヤレスチャネル（例えば、Wi-Fi接続）、ワイヤード接続（例えば、DSL、ケーブルモデム、等）、またはファイルサーバ上に記憶された符号化されたビデオデータにアクセスするのに好適な両方の組合せを含み得る。記憶デバイスからの符号化された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、またはこれらの組合せであり得る。

20

30

【0163】

[0174]本開示の技法は、必ずしもワイヤレスアプリケーションまたは設定に限定されるとは限らない。本技法は、無線テレビ放送、ケーブルテレビ送信、衛星テレビ送信、HTTPを介した動的適応型ストリーミング（DASH）のようなインターネットストリーミングビデオ送信、データ記憶媒体上に符号化されるデジタルビデオ、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、様々なマルチメディアアプリケーションのうちの任意のもの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、システムは、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/またはビデオ電話通信などの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。

40

【0164】

[0175]一例では、ソースデバイスは、ビデオソース、ビデオエンコーダ、および出力インターフェースを含む。宛先デバイスは、入力インターフェース、ビデオデコーダ、およびディスプレイデバイスを含み得る。ソースデバイスのビデオエンコーダは、ここで開示される技法を適用するように構成され得る。他の例では、ソースデバイスおよび宛先デバイスは、他の構成要素または配置を含み得る。例えば、ソースデバイスは、外部カメラのような外部ビデオソースからビデオデータを受信し得る。同様に、宛先デバイスは、一体化されたディスプレイデバイスを含むのではなく、外部ディスプレイデバイスとインターフェースし得る。

50

【 0 1 6 5 】

[0176]上記の例となるシステムは、単なる一例に過ぎない。並列にビデオデータを処理するための技法は、任意のデジタルビデオ符号化および/または復号デバイスによって実行され得る。概して、本開示の技法は、ビデオ符号化デバイスによって実行されるが、本技法はまた、典型的に「CODEC」と呼ばれるビデオエンコーダ/デコーダによっても実行され得る。さらに、本開示の技法はまた、ビデオプリプロセッサによって実行され得る。ソースデバイスおよび宛先デバイスは、ソースデバイスが宛先デバイスへの送信のためにコーディングされたビデオデータを生成するそのようなコーディングデバイスの例に過ぎない。いくつかの例では、ソースおよび宛先デバイスは、デバイスの各々がビデオ符号化および復号構成要素を含むような実質的に対称的な方法で動作し得る。したがって、

例となるシステムは、例えば、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、またはビデオ電話通信のために、ビデオデバイス間の一方向または双方向のビデオ送信をサポートし得る。

10

【 0 1 6 6 】

[0177]ビデオソースは、ビデオカメラのような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を含むビデオアーカイブ、および/またはビデオコンテンツプロバイダからのビデオを受信するためのビデオフィードインターフェースを含み得る。さらなる代替として、ビデオソースは、ソースビデオとしてコンピュータグラフィックススペースのデータ、またはコンピュータにより生成されたビデオ、アーカイブされたビデオ、およびライブビデオの組合せを生成し得る。いくつかのケースでは、ビデオソースがビデオカメラである場合、ソースデバイスおよび宛先デバイスは、いわゆるカメラフォンまたはビデオフォンを形成し得る。しかしながら、上述されたように、本開示で説明される技法は、ビデオコーディング全般に適用可能であり得、ワイヤレスおよび/またはワイヤードアプリケーションに適用され得る。各ケースでは、キャプチャされた、予めキャプチャされた、またはコンピュータにより生成されたビデオは、ビデオエンコーダによって符号化され得る。その後、符号化されたビデオ情報は、コンピュータ可読媒体上に出力インターフェースによって出力され得る。

20

【 0 1 6 7 】

[0178]述べられたように、コンピュータ可読媒体は、ワイヤレスブロードキャストまたはワイヤードネットワーク送信のなどの一時的な媒体、またはハードディスク、フラッシュドライブ、コンパクトディスク、デジタルビデオディスク、ブルーレイディスク、または他のコンピュータ可読媒体などの記憶媒体（すなわち、非一時的な記憶媒体）を含み得る。いくつかの例では、ネットワークサーバ（図示せず）は、例えば、ネットワーク送信を介して、宛先デバイスに符号化されたビデオデータを提供し、ソースデバイスから符号化されたビデオデータを受信し得る。同様に、ディスクスタンピング設備のような媒体製造設備のコンピューティングデバイスは、ソースデバイスから符号化されたビデオデータを受信し、符号化されたビデオデータを含むディスクを製造し得る。したがって、コンピュータ可読媒体は、様々な例において、様々な形態の1つまたは複数のコンピュータ可読媒体を含むことが理解され得る。

30

【 0 1 6 8 】

[0179]宛先デバイスの入力インターフェースは、コンピュータ可読媒体から情報を受信する。コンピュータ可読媒体の情報は、ブロックおよび他のコーディングされたユニット、例えば、ピクチャのグループ（GOP: group of pictures）の処理および/または特性を記述するシンタックス要素を含む、ビデオエンコーダによって定義されるシンタックス情報を含み得、これはまた、ビデオデコーダによって使用される。ディスプレイデバイスは、復号されたビデオデータをユーザに表示し得、ブラウン管（CRT）、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、様々なディスプレイデバイスのうちの任意のものを備え得る。本発明の様々な実施形態が、説明されてきた。

40

【 0 1 6 9 】

50

[0180]符号化デバイス104および復号デバイス112の特定の詳細が、それぞれ図9および図10に示される。図9は、本開示で説明される技法のうちの1つまたは複数をインプリメントし得る、例となる符号化デバイス104を例示するブロック図である。符号化デバイス104は、例えば、ここで説明されたシンタックス構造（例えば、VPS、SPS、PPS、または他のシンタックス要素のシンタックス構造）を生成し得る。符号化デバイス104は、ビデオスライス内のビデオブロックのイントラ予測およびインター予測コーディングを実行し得る。前述されたように、イントラコーディングは、所与のビデオフレームまたはピクチャ内の空間的冗長性を低減または取り除くために、空間的予測に少なくとも部分的に依拠する。インターコーディングは、ビデオシーケンスの隣り合うまたは周囲のフレーム内の時間的冗長性を低減または取り除くために、時間的予測に少なくとも部分的に依拠する。イントラモード（Iモード）は、いくつかの空間ベースの圧縮モードのうちの任意のものを指し得る。片方向予測（Pモード）または双予測（Bモード）などのインターモードは、いくつかの時間ベースの圧縮モードのうちの任意のものを指し得る。

【0170】

[0181]符号化デバイス104は、区分化ユニット35、予測処理ユニット41、フィルタユニット63、ピクチャメモリ64、加算器50、変換処理ユニット52、量子化ユニット54、およびエントロピー符号化ユニット56を含む。予測処理ユニット41は、動き推定ユニット42、動き補償ユニット44、およびイントラ予測処理ユニット46を含む。ビデオブロック再構成のために、符号化デバイス104はまた、逆量子化ユニット58、逆変換処理ユニット60、および加算器62を含む。フィルタユニット63は、デブロッキングフィルタ、適応ループフィルタ（ALF）、およびサンプル適応オフセット（SAO）フィルタなどの1つまたは複数のループフィルタを表すように意図される。フィルタユニット63は、インループフィルタとして図9に示されているが、他の構成では、フィルタユニット63は、ポストループフィルタとしてインプリメントされ得る。後処理デバイス57が、符号化デバイス104によって生成される符号化されたビデオデータに対して追加の処理を実行し得る。本開示の技法は、いくつかの事例では、符号化デバイス104によってインプリメントされ得る。しかしながら、他の事例では、本開示の技法のうちの1つまたは複数は、後処理デバイス57によってインプリメントされ得る。

【0171】

[0182]図9に示されているように、符号化デバイス104は、ビデオデータを受信し、区分化ユニット35は、データをビデオブロックに区分化する。区分化はまた、スライス、スライスセグメント、タイル、または他のより大きいユニットへの区分化のみならず、例えば、CUおよびLCUの4分木構造による、ビデオブロック区分化を含み得る。符号化デバイス104は、概して、符号化されるべきビデオスライス内のビデオブロックを符号化する構成要素を例示する。スライスは、複数のビデオブロックに（および、場合によっては、タイルと呼ばれるビデオブロックのセットに）分割され得る。予測処理ユニット41は、エラー結果（例えば、コーディングレートおよび歪みのレベル、または同様のもの）に基づいて、現在のビデオブロックのために、複数のイントラ予測コーディングモードのうちの1つまたは複数のインター予測コーディングモードのうちの1つなど、複数の可能なコーディングモードのうちの1つを選択し得る。予測処理ユニット41は、残差ブロックデータを生成するために加算器50に、および参照ピクチャとして使用するための符号化されたブロックを再構成するために加算器62に、結果として得られるイントラまたはインターコーディングされたブロックを提供し得る。

【0172】

[0183]予測処理ユニット41内のイントラ予測処理ユニット46は、空間的圧縮を提供するために、コーディングされるべき現在のブロックと同じフレームまたはスライス中の1つまたは複数の隣接ブロックに関連して（relative to）、現在のビデオブロックのイントラ予測コーディングを実行し得る。予測処理ユニット41内の動き推定ユニット42および動き補償ユニット44は、時間的圧縮を提供するために、1つまたは複数の参照ピク

チャ中の 1 つまたは複数の予測ブロックに関連して (relative to)、現在のビデオブロックのインター予測コーディングを実行する。

【0173】

[0184]動き推定ユニット 42 は、ビデオシーケンスのための所定のパターンに従って、ビデオスライスのためのインター予測モードを決定するように構成され得る。所定のパターンは、P スライス、B スライス、または G P B スライスとして、シーケンス中のビデオスライスを指定し得る。動き推定ユニット 42 および動き補償ユニット 44 は、高度に一体化され得るが、概念的な目的のために別個に例示されている。動き推定ユニット 42 によって実行される動き推定は、動きベクトルを生成するプロセスであり、これは、ビデオブロックに関する動きを推定する。動きベクトルは、例えば、参照ピクチャ内の予測ブロックに対する現在のビデオフレームまたはピクチャ内のビデオブロックの予測ユニット (PU) の変位を示し得る。

10

【0174】

[0185]予測ブロックは、絶対差分和 (SAD: sum of absolute difference)、2 乗差分和 (SSD: sum of square difference)、または他の差分メトリックによって決定され得るピクセル差分の観点から、コーディングされるべきビデオブロックの PU と密接に一致することが見出されたブロックである。いくつかの例では、符号化デバイス 104 は、ピクチャメモリ 64 に記憶された参照ピクチャのサブ整数ピクセル位置についての値を計算し得る。例えば、符号化デバイス 104 は、参照ピクチャの 4 分の 1 ピクセル位置、8 分の 1 ピクセル位置、または他の分数ピクセル位置の値を補間し得る。したがって、動き推定ユニット 42 は、フルピクセル位置および分数ピクセル位置に対して動き探索を実行し、分数ピクセル精度で動きベクトルを出力し得る。

20

【0175】

[0186]動き推定ユニット 42 は、PU の位置と、参照ピクチャの予測ブロックの位置とを比較することによって、インターコーディングされたスライス中のビデオブロックの PU のための動きベクトルを計算する。参照ピクチャは、その各々がピクチャメモリ 64 に記憶された 1 つまたは複数の参照ピクチャを識別する、第 1 の参照ピクチャリスト (リスト 0) または第 2 の参照ピクチャリスト (リスト 1) から選択され得る。動き推定ユニット 42 は、計算された動きベクトルを、エントロピー符号化ユニット 56 および動き補償ユニット 44 に送る。

30

【0176】

[0187]動き補償ユニット 44 によって実行される動き補償は、動き推定によって決定される動きベクトルに基づいて、予測ブロックをフェッチまたは生成することを伴い得、場合によっては、サブピクセル精度までの (to sub-pixel precision) 補間を実行する。現在のビデオブロックの PU のための動きベクトルを受信すると、動き補償ユニット 44 は、参照ピクチャリストにおいて動きベクトルがそれを指し示す予測ブロックを位置特定し得る。符号化デバイス 104 は、コーディングされている現在のビデオブロックのピクセル値から予測ブロックのピクセル値を減算し、ピクセル差分値を形成することによって、残差ビデオブロックを形成する。ピクセル差分値は、ブロックについての残差データを形成し、ルーマおよびクロマ差分成分の両方を含み得る。加算器 50 は、この減算演算を実行する 1 つまたは複数の構成要素を表す。動き補償ユニット 44 はまた、ビデオスライスのビデオブロックを復号する際に、復号デバイス 112 によって使用するための、ビデオブロックとビデオスライスとに関連付けられたシンタックス要素を生成し得る。

40

【0177】

[0188]イントラ予測処理ユニット 46 は、上記で説明されたように、動き推定ユニット 42 および動き補償ユニット 44 によって実行されるインター予測の代替として、現在のブロックをイントラ予測し得る。具体的には、イントラ予測処理ユニット 46 は、現在のブロックを符号化するために使用すべきイントラ予測モードを決定し得る。いくつかの例では、イントラ予測処理ユニット 46 は、例えば、別個の符号化パス中に、様々なイントラ予測モードを使用して現在のブロックを符号化し得、イントラ予測処理ユニット 46 は

50

、テストされたモードから使用すべき適切なイントラ予測モードを選択し得る。例えば、イントラ予測処理ユニット46は、様々なテストされたイントラ予測モードに関する（for）レート-歪み分析を使用してレート-歪み値を計算し得、テストされたモードの中で最良のレート-歪み特性を有するイントラ予測モードを選択し得る。レート-歪み分析は、一般に、符号化されたブロックと、符号化されたブロックを作成するために符号化された元の符号化されていないブロックとの間の歪み（またはエラー）の量、ならびに、符号化されたブロックを作成するために使用されるビットレート（すなわち、ビット数）を決定する。イントラ予測処理ユニット46は、どのイントラ予測モードがブロックのための最良のレート-歪み値を示すかを決定するために、様々な符号化されたブロックに関するレートおよび歪みから比率（ratio）を計算し得る。

10

【0178】

[0189]いずれのケースでも、ブロックのためのイントラ予測モードを選択した後、イントラ予測処理ユニット46は、エントロピー符号化ユニット56にブロックのための選択されたイントラ予測モードを示す情報を提供し得る。エントロピー符号化ユニット56は、選択されたイントラ予測モードを示す情報を符号化し得る。符号化デバイス104は、様々なブロックのための符号化コンテキストの定義のみならず、最確イントラ予測モード（most probable intra-prediction mode）のインジケーション、イントラ予測モードインデックステーブル、およびコンテキストの各々に対して使用すべき修正されたイントラ予測モードインデックステーブルを、送信されたビットストリーム構成データ中に含め得る。ビットストリーム構成データは、複数のイントラ予測モードインデックステーブルと、複数の修正されたイントラ予測モードインデックステーブル（コードワードマッピングテーブルとも呼ばれる）とを含み得る。

20

【0179】

[0190]予測処理ユニット41が、インター予測またはイントラ予測のいずれかを介して、現在のビデオブロックのための予測ブロックを生成した後、符号化デバイス104は、現在のビデオブロックから予測ブロックを減算することによって、残差ビデオブロックを形成する。残差ブロックにおける残差ビデオデータは、1つまたは複数のTUに含まれ、変換処理ユニット52に適用され得る。変換処理ユニット52は、離散コサイン変換（DCT）または概念的に同様の変換などの変換を使用して、残差変換係数に残差ビデオデータを変換する。変換処理ユニット52は、残差ビデオデータを、ピクセル領域から周波数領域などの変換領域に変換し得る。

30

【0180】

[0191]変換処理ユニット52は、結果として得られた変換係数を量子化ユニット54に送り得る。量子化ユニット54は、ビットレートをさらに低減させるために、変換係数を量子化する。量子化プロセスは、係数の一部または全てに関連付けられたビット深度を低減させ得る。量子化の程度は、量子化パラメータを調整することによって修正され得る。いくつかの例では、量子化ユニット54は、その後、量子化された変換係数を含む行列の走査を実行し得る。代替として、エントロピー符号化ユニット56が走査を実行し得る。

【0181】

[0192]量子化に続いて、エントロピー符号化ユニット56は、量子化された変換係数をエントロピー符号化する。例えば、エントロピー符号化ユニット56は、コンテキスト適応可変長コーディング（CAVLC）、コンテキスト適応バイナリ算術コーディング（CABAC）、シンタックススペースのコンテキスト適応バイナリ算術コーディング（SBAC）、確率間隔区分化エントロピー（PIPE）コーディング、または別のエントロピー符号化技法を実行し得る。エントロピー符号化ユニット56によるエントロピー符号化に続いて、符号化されたビットストリームは、復号デバイス112に送信されるか、または復号デバイス112による後の送信または取り出しのためにアーカイブされ得る。エントロピー符号化ユニット56はまた、コーディングされている現在のビデオスライスのための動きベクトルおよび他のシンタックス要素をエントロピー符号化し得る。

40

【0182】

50

[0193]逆量子化ユニット 5 8 および逆変換処理ユニット 6 0 は、参照ピクチャの参照ブロックとして後に使用するためのピクセル領域における残差ブロックを再構成するために、それぞれ逆量子化および逆変換を適用する。動き補償ユニット 4 4 は、参照ピクチャリスト内の参照ピクチャのうちの 1 つの予測ブロックに残差ブロックを加算することによって、参照ブロックを計算し得る。動き補償ユニット 4 4 はまた、動き推定の際に使用するためのサブ整数ピクセル値を計算するために、再構成された残差ブロックに 1 つまたは複数の補間フィルタを適用し得る。加算器 6 2 は、ピクチャメモリ 6 4 中での記憶のための参照ブロックを作成するために、動き補償ユニット 4 4 によって作成された動き補償された予測ブロックに、再構成された残差ブロックを加算する。参照ブロックは、後続のビデオフレームまたはピクチャ中のブロックをインター予測するための参照ブロックとして、動き推定ユニット 4 2 および動き補償ユニット 4 4 によって使用され得る。

10

【0183】

[0194]このようにして、図 9 の符号化デバイス 1 0 4 は、符号化されたビデオビットストリームについてのシンタックスを生成するように構成されたビデオエンコーダの例を表す。符号化デバイス 1 0 4 は、例えば、上記で説明されたように、V P S、S P S、および P P S パラメータセットを生成し得る。符号化デバイス 1 0 4 は、上記で説明されたプロセスを含む、ここで説明された技法のうちの任意のものを実行し得る。本開示の技法は、概して、符号化デバイス 1 0 4 に関して説明されてきたが、上述されたように、本開示の技法のうちのいくつかはまた、後処理デバイス 5 7 によってインプリメントされ得る。

【0184】

20

[0195]図 1 0 は、例となる復号デバイス 1 1 2 を例示するブロック図である。復号デバイス 1 1 2 は、エントロピー復号ユニット 8 0、予測処理ユニット 8 1、逆量子化ユニット 8 6、逆変換処理ユニット 8 8、加算器 9 0、フィルタユニット 9 1、およびピクチャメモリ 9 2 を含む。予測処理ユニット 8 1 は、動き補償ユニット 8 2 およびイントラ予測処理ユニット 8 4 を含む。復号デバイス 1 1 2 は、いくつかの例では、図 1 0 からの符号化デバイス 1 0 4 に関して説明された符号化パスとは概して逆の復号パスを実行し得る。

【0185】

[0196]復号プロセス中、復号デバイス 1 1 2 は、符号化デバイス 1 0 4 によって送られた符号化されたビデオスライスのビデオブロックと、関連するシンタックス要素とを表す符号化されたビデオビットストリームを受信する。いくつかの実施形態では、復号デバイス 1 1 2 は、符号化デバイス 1 0 4 から符号化されたビデオビットストリームを受信し得る。いくつかの実施形態では、復号デバイス 1 1 2 は、サーバ、媒体認識ネットワーク要素 (M A N E)、ビデオエディタ/スプライサ、または上記で説明された技法のうちの 1 つまたは複数をインプリメントするように構成された他のそのようなデバイスなどの、ネットワークエンティティ 7 9 から符号化されたビデオビットストリームを受信し得る。ネットワークエンティティ 7 9 は、符号化デバイス 1 0 4 を含むことも含まないこともあり得る。本開示で説明された技法のうちのいくつかは、ネットワークエンティティ 7 9 が、符号化されたビデオビットストリームを復号デバイス 1 1 2 に送信する前に、ネットワークエンティティ 7 9 によってインプリメントされ得る。いくつかのビデオ復号システムでは、ネットワークエンティティ 7 9 および復号デバイス 1 1 2 は、別個のデバイスの一部であり得、一方、他の事例では、ネットワークエンティティ 7 9 に関して説明される機能は、復号デバイス 1 1 2 を備える同じデバイスによって実行され得る。

30

40

【0186】

[0197]復号デバイス 1 1 2 のエントロピー復号ユニット 8 0 は、量子化された係数、動きベクトル、および他のシンタックス要素を生成するために、ビットストリームをエントロピー復号する。エントロピー復号ユニット 8 0 は、予測処理ユニット 8 1 に動きベクトルおよび他のシンタックス要素を転送する。復号デバイス 1 1 2 は、ビデオスライスレベルおよび/またはビデオブロックレベルでシンタックス要素を受信し得る。エントロピー復号ユニット 8 0 は、V P S、S P S、および P P S などの、または複数のパラメータセット中の固定長シンタックス要素と可変長シンタックス要素との両方を処理およびパース

50

し得る。

【0187】

[0198]ビデオスライスがイントラコーディングされた(I)スライスとしてコーディングされるとき、予測処理ユニット81のイントラ予測処理ユニット84は、シグナリングされたイントラ予測モードと、現在のフレームまたはピクチャの以前に復号されたブロックからのデータとに基づいて、現在のビデオスライスのビデオブロックについての予測データを生成し得る。ビデオフレームがインターコーディングされた(すなわち、B、PまたはGPB)スライスとしてコーディングされるとき、予測処理ユニット81の動き補償ユニット82は、エントロピー復号ユニット80から受信された動きベクトルと他のシンタックス要素とに基づいて、現在のビデオスライスのビデオブロックについての予測ブロックを作成する。予測ブロックは、参照ピクチャリスト内の参照ピクチャのうちの1つから作成され得る。復号デバイス112は、ピクチャメモリ92に記憶された参照ピクチャに基づいて、デフォルトの構成技法を使用して、参照フレームリスト、すなわち、リスト0およびリスト1を構成し得る。

10

【0188】

[0199]動き補償ユニット82は、動きベクトルと他のシンタックス要素とをパースすることによって、現在のビデオスライスのビデオブロックについての予測情報を決定し、復号されている現在のビデオブロックについての予測ブロックを作成するために予測情報を使用する。例えば、動き補償ユニット82は、ビデオスライスのビデオブロックをコーディングするために使用される予測モード(例えば、イントラまたはインター予測)と、インター予測スライスタイプ(例えば、Bスライス、Pスライス、またはGPBスライス)と、スライスのための1つまたは複数の参照ピクチャリストのための構成情報と、スライスの各インター符号化されたビデオブロックのための動きベクトルと、スライスの各インターコーディングされたビデオブロックのためのインター予測ステータスと、現在のビデオスライス中のビデオブロックを復号するための他の情報とを決定するために、パラメータセット中の1つまたは複数のシンタックス要素を使用し得る。

20

【0189】

[0200]動き補償ユニット82はまた、補間フィルタに基づいて補間を実行し得る。動き補償ユニット82は、参照ブロックのサブ整数ピクセルについての補間された値を計算するために、ビデオブロックの符号化中に符号化デバイス104によって使用されたような補間フィルタを使用し得る。このケースでは、動き補償ユニット82は、受信されたシンタックス要素から、符号化デバイス104によって使用された補間フィルタを決定し得、予測ブロックを作成するために、補間フィルタを使用し得る。

30

【0190】

[0201]逆量子化ユニット86は、ビットストリーム中で提供され、かつエントロピー復号ユニット80によって復号された、量子化された変換係数を逆量子化、すなわち量子化解除する。逆量子化プロセスは、量子化の程度、および同様に、適用されるべき逆量子化の程度を決定するために、ビデオスライス中の各ビデオブロックについて、符号化デバイス104によって計算された量子化パラメータの使用を含み得る。逆変換処理ユニット88は、ピクセル領域における残差ブロックを作成するために、変換係数に、逆変換(例えば、逆DCTまたは他の好適な逆変換)、逆整数変換、または概念的に同様の逆変換プロセスを適用する。

40

【0191】

[0202]動き補償ユニット82が、動きベクトルと他のシンタックス要素とに基づいて、現在のビデオブロックのための予測ブロックを生成した後、復号デバイス112は、逆変換処理ユニット88からの残差ブロックを、動き補償ユニット82によって生成された対応する予測ブロックと加算することによって、復号されたビデオブロックを形成する。加算器90は、この加算演算を実行する1つまたは複数の構成要素を表す。所望される場合、ループフィルタ(コーディンググループ中またはコーディンググループ後のいずれか)もまた、ピクセル遷移を平滑化するか、または別の方法でビデオ品質を改善するために使用さ

50

れ得る。フィルタユニット 91 は、デブロッキングフィルタ、適応ループフィルタ (ALF)、およびサンプル適応オフセット (SAO) フィルタなどの 1 つまたは複数のループフィルタを表すように意図される。フィルタユニット 91 は、インループフィルタとして図 10 に示されているが、他の構成では、フィルタユニット 91 は、ポストループフィルタとしてインプリメントされ得る。その後、所与のフレームまたはピクチャ中の復号されたビデオブロックは、後続の動き補償のために使用される参照ピクチャを記憶するピクチャメモリ 92 に記憶される。ピクチャメモリ 92 はまた、図 1 に示されるビデオ宛先デバイス 122 のようなディスプレイデバイス上での後の提示のために、復号されたビデオを記憶する。

【0192】

[0203] 前述の説明では、本願の態様は、その特定の実施形態を参照して説明されたが、当業者であれば、本発明はそれに限定されないことを認識するであろう。したがって、本願の例示的な実施形態が、ここで詳細に説明されてきた一方で、発明の概念は、他の方法で様々な具現化および用いられ得、添付の特許請求の範囲は、先行技術によって限定される場合を除き、そのような変形例を含むように解釈されるものと意図されていることが理解されるべきである。上記で説明された発明の様々な特徴および態様は、個々にまたは一緒に使用され得る。さらに、実施形態は、本明細書のより広い趣旨および範囲から逸脱することなく、ここで説明されたものを超えた任意の数の環境およびアプリケーションにおいて利用され得る。したがって、本明細書および図面は、限定的ではなく、例示的であると見なされるべきである。例示を目的として、方法は、特定の順序で説明された。代替の実施形態では、方法は、説明されたものとは異なる順序で実行され得ることが理解されるべきである。

【0193】

[0204] 構成要素が、ある特定の動作を実行する「ように構成される」ものとして説明されている場合、このような構成は、例えば、動作を実行するように電子回路または他のハードウェアを設計することによって、動作を実行するようにプログラム可能な電子回路 (例えば、マイクロプロセッサ、または他の好適な電子回路) をプログラミングすることによって、またはこれらの任意の組合せで、達成され得る。

【0194】

[0205] ここで開示された実施形態に関連して説明された様々な例示的な論理ブロック、モジュール、回路、およびアルゴリズムステップは、電子ハードウェア、コンピュータソフトウェア、ファームウェア、またはこれらの組合せとしてインプリメントされ得る。ハードウェアおよびソフトウェアのこの互換性を明確に例示するために、様々な例示的な構成要素、ブロック、モジュール、回路、およびステップが、概してそれらの機能の観点から上記で説明されてきた。このような機能が、ハードウェアとしてインプリメントされるか、あるいはソフトウェアとしてインプリメントされるかは、特定のアプリケーションおよびシステム全体に課せられる設計制約に依存する。当業者は、特定のアプリケーションごとに多様な方法において、説明された機能をインプリメントし得るが、このようなインプリメンテーションの決定は、本発明の範囲から逸脱を引き起こしていると解釈されるべきでない。

【0195】

[0206] ここで説明された技法はまた、電子ハードウェア、コンピュータソフトウェア、ファームウェア、またはこれらの任意の組合せでインプリメントされ得る。このような技法は、汎用コンピュータ、ワイヤレス通信デバイスハンドセット、またはワイヤレス通信デバイスハンドセットおよび他のデバイスにおけるアプリケーションを含む複数の用途を有する集積回路デバイスなどの様々なデバイスのうちの任意のものでインプリメントされ得る。モジュールまたは構成要素として説明された任意の特徴は、集積論理デバイスにおいて共に、またはディスクリートではあるが相互動作可能な論理デバイスとして別々にインプリメントされ得る。ソフトウェアでインプリメントされる場合、本技法は、少なくとも部分的に、命令を含むプログラムコードを備えるコンピュータ可読データ記憶媒体によ

10

20

30

40

50

って実現され得、これら命令は、実行されると、上記で説明された方法のうちの1つまたは複数を実行する。コンピュータ可読データ記憶媒体は、コンピュータプログラム製品の一部を形成し得、それは、パッケージング材料を含み得る。コンピュータ可読媒体は、同期ダイナミックランダムアクセスメモリ (SDRAM) のようなランダムアクセスメモリ (RAM)、読取専用メモリ (ROM)、不揮発ランダムアクセスメモリ (NVRAM)、電氣的に消去可能なプログラマブル読取専用メモリ (EEPROM (登録商標))、FLASHメモリ、磁気または光データ記憶媒体、および同様のものなどの、メモリまたはデータ記憶媒体を備え得る。本技法は、追加または代替として、少なくとも部分的に、伝搬された信号または波などの、コンピュータによってアクセス、読取、および/または実行されることができ、かつ、データ構造または命令の形式でプログラムコードを搬送または通信する、コンピュータ可読通信媒体によって実現され得る。

10

【0196】

[0207] プログラムコードは、プロセッサによって実行され得、それは、1つまたは複数のデジタルシグナルプロセッサ (DSP)、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ロジックアレイ (FPGA)、または他の同等の集積されたまたはディスクリートなロジック回路などの、1つまたは複数のプロセッサを含み得る。このようなプロセッサは、本開示において説明された技法のうちの任意のものを実行するように構成され得る。汎用プロセッサは、マイクロプロセッサであり得るが、代替として、このプロセッサは、任意の従来のプロセッサ、コントローラ、マイクロコントローラ、またはステートマシン (state machine) であり得る。プロセッサはまた、コンピューティングデバイスの組合せ、例えば、DSPとマイクロプロセッサの組合せ、複数のマイクロプロセッサ、DSPコアと連携した1つまたは複数のマイクロプロセッサ、あるいはその他任意のこのような構成としてインプリメントされ得る。したがって、ここで使用される場合、「プロセッサ」という用語は、前述の構造の任意のもの、前述の構造の任意の組合せ、またはここで説明された技法のインプリメンテーションに好適なその他任意の構造または装置を指し得る。加えて、いくつかの態様では、ここで説明された機能は、符号化および復号のために構成された専用のソフトウェアモジュールまたはハードウェアモジュール内に提供され得、または、複合ビデオエンコーダ - デコーダ (CODEC) に組み込まれ得る。

20

以下に本願の出願当初の特許請求の範囲に記載された発明を付記する。

30

[C 1] ビデオデータを符号化する方法であって、

前記ビデオデータを取得することと、

コーディングユニットについて、補間フィルタのセットから補間フィルタのサブセットを決定することと、ここにおいて、前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた、前記ビデオデータ中の情報に基づいて決定され、

前記コーディングユニットを符号化することと、ここにおいて、前記コーディングユニットを符号化することは、前記コーディングユニットについての動き推定および動き補償のために補間フィルタを選択することを含み、ここにおいて、前記補間フィルタは、前記補間フィルタのサブセットから選択され、

符号化されたビデオビットストリームを生成することと、ここにおいて、前記符号化されたビデオビットストリームは、前記符号化されたコーディングユニットを含む、を備える方法。

40

[C 2] 前記補間フィルタのサブセットは、前記コーディングユニットのコーディングレベルに基づいて決定される、C 1に記載の方法。

[C 3] 前記補間フィルタのサブセットは、前記コーディングユニットのための予測モードに基づいて決定される、C 1に記載の方法。

[C 4] 前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた動き情報に基づいて決定される、C 1に記載の方法。

[C 5] 前記補間フィルタのサブセットは、前記コーディングユニットをコーディングするために使用されるコーディングツールに基づいて決定される、C 1に記載の方法。

50

[C 6] 前記補間フィルタのサブセットは、前記コーディングユニットのサイズに基づいて決定される、C 1に記載の方法。

[C 7] 前記補間フィルタは、以前にコーディングされたブロックから選択される、C 1に記載の方法。

[C 8] 前記補間フィルタは、前記コーディングユニットの隣接ブロックから選択される、C 1に記載の方法。

[C 9] 前記補間フィルタを第 2 の補間フィルタとマージすること、ここにおいて、結果として得られるマージされた補間フィルタが、前記動き推定および前記動き補償のために使用される、

をさらに備える、C 1に記載の方法。

10

[C 10] 前記補間フィルタは、第 2 の補間フィルタに関連付けられ、前記補間フィルタは、整数ピクセル位置のためのものであり、前記第 2 の補間フィルタは、分数ピクセル位置のためのものである、C 1に記載の方法。

[C 11] 前記補間フィルタと前記第 2 の補間フィルタは、異なるカットオフ周波数を有する、C 10に記載の方法。

[C 12] 前記符号化されたビデオビットストリームは、前記補間フィルタを識別する値を含む、C 1に記載の方法。

[C 13] 前記補間フィルタのセットは、前記符号化されたビデオビットストリームとともにシグナリングされず、前記補間フィルタの識別は、前記符号化されたビデオビットストリームから暗示され得る、C 1に記載の方法。

20

[C 14] 装置であって、

ビデオデータを記憶するように構成されたメモリと、

前記ビデオデータを取得することと、

コーディングユニットについて、補間フィルタのセットから補間フィルタのサブセットを決定することと、ここにおいて、前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた、前記ビデオデータ中の情報に基づいて決定され、

前記コーディングユニットを符号化することと、ここにおいて、前記処理を符号化することは、動き推定および動き補償のために補間フィルタを選択することを含み、ここにおいて、前記補間フィルタは、前記補間フィルタのサブセットから選択され、

符号化されたビデオビットストリームを生成することと、ここにおいて、前記符号化されたビデオビットストリームは、前記符号化されたコーディングユニットを含む、

30

を行うように構成されたプロセッサと

を備える装置。

[C 15] 前記装置は、ピクチャをキャプチャするためのカメラを有するモバイルデバイスを備える、C 14に記載の装置。

[C 16] 命令を記憶したコンピュータ可読媒体であって、前記命令は、プロセッサによって実行されると、前記プロセッサに、

ビデオデータを取得することと、

コーディングユニットについて、補間フィルタのセットから補間フィルタのサブセットを決定することと、ここにおいて、前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた、前記ビデオデータ中の情報に基づいて決定され、

40

前記コーディングユニットを符号化することと、ここにおいて、前記コーディングユニットを符号化することは、動き推定および動き補償のために補間フィルタを選択することを含み、ここにおいて、前記補間フィルタは、前記補間フィルタのサブセットから選択され、

符号化されたビデオビットストリームを生成することと、ここにおいて、前記符号化されたビデオビットストリームは、前記符号化されたコーディングユニットを含む、

を含む動作を実行させる、コンピュータ可読媒体。

[C 17] ビデオデータを符号化するための装置であって、

ビデオデータを取得するための手段と、

コーディングユニットについて、補間フィルタのセットから補間フィルタのサブセットを

50

決定するための手段と、ここにおいて、前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた、前記ビデオデータ中の情報に基づいて決定され、
前記コーディングユニットを符号化するための手段と、ここにおいて、前記コーディングユニットを符号化するための前記手段は、動き推定および動き補償のために補間フィルタを選択することを含み、
ここにおいて、前記補間フィルタは、補間フィルタのセットから選択され、

符号化されたビデオビットストリームを生成するための手段と、ここにおいて、前記符号化されたビデオビットストリームは、前記符号化されたコーディングユニットを含む、
を備える装置。

[C 1 8] ビデオデータを復号するための方法であって、
符号化されたビデオビットストリームを取得することと、
前記符号化されたビデオビットストリームから現在のフレーム中のコーディングユニットのための参照フレームを決定することと、
補間フィルタのセットから補間フィルタのサブセットを決定することと、ここにおいて、
前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた、前記符号化されたビデオビットストリーム中の情報に基づいて決定され、
前記補間フィルタのサブセットから補間フィルタを決定することと、
前記コーディングユニットを再構成するために、前記参照フレームおよび前記補間フィルタを使用することと
を備える方法。

[C 1 9] 前記補間フィルタのサブセットは、前記コーディングユニットのコーディングレベルに基づいて決定される、C 1 8に記載の方法。

[C 2 0] 前記補間フィルタのサブセットは、前記コーディングユニットのための予測モードに基づいて決定される、C 1 8に記載の方法。

[C 2 1] 前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた動き情報に基づいて決定される、C 1 8に記載の方法。

[C 2 2] 前記補間フィルタのサブセットは、前記コーディングユニットをコーディングするために使用されるコーディングツールに基づいて決定される、C 1 8に記載の方法。

[C 2 3] 前記補間フィルタのサブセットは、前記コーディングユニットのサイズに基づいて決定される、C 1 8に記載の方法。

[C 2 4] 前記補間フィルタは、以前にコーディングされたブロックから選択される、C 1 8に記載の方法。

[C 2 5] 前記補間フィルタは、隣接ブロックから選択される、C 1 8に記載の方法。

[C 2 6] 前記補間フィルタを第2の補間フィルタとマージすること、ここにおいて、結果として得られるマージされた補間フィルタが、前記コーディングユニットを再構成するために使用される、
をさらに備える、C 1 8に記載の方法。

[C 2 7] 前記補間フィルタは、第2の補間フィルタに関連付けられ、前記補間フィルタは、整数ピクセル位置のためのものであり、前記第2の補間フィルタは、分数ピクセル位置のためのものである、C 1 8に記載の方法。

[C 2 8] 前記補間フィルタと前記第2の補間フィルタは、異なるカットオフ周波数を有する、C 2 7に記載の方法。

[C 2 9] 前記符号化されたビデオビットストリームは、前記補間フィルタを識別する値を含む、C 1 8に記載の方法。

[C 3 0] 前記符号化されたビデオビットストリームを使用して、前記補間フィルタの識別を導出すること
をさらに備える、C 1 8に記載の方法。

[C 3 1] 装置であって、
符号化されたビデオデータを記憶するように構成されたメモリと、
符号化されたビデオビットストリームを取得することと、

10

20

30

40

50

前記符号化されたビデオビットストリームから現在のフレーム中のコーディングユニットのための参照フレームを決定することと、
補間フィルタのセットから補間フィルタのサブセットを決定することと、ここにおいて、
前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた、前記符号化されたビデオビットストリーム中の情報に基づいて決定され、
前記補間フィルタのサブセットから補間フィルタを決定することと、
前記コーディングユニットを再構成するために、前記参照フレームおよび前記補間フィルタを使用することと
を行うように構成されたプロセッサと
を備える装置。

10

[C 3 2] 前記ビデオデータを表示するためのディスプレイ
をさらに備える、C 3 1に記載の装置。

[C 3 3] 前記装置は、ピクチャをキャプチャするためのカメラを有するモバイルデバイスを備える、C 3 1に記載の装置。

[C 3 4] 命令を記憶したコンピュータ可読媒体であって、前記命令は、プロセッサによって実行されると、前記プロセッサに、

符号化されたビデオビットストリームを取得することと、
前記符号化されたビデオビットストリームから現在のフレーム中のコーディングユニットのための参照フレームを決定することと、
補間フィルタのセットから補間フィルタのサブセットを決定することと、ここにおいて、
前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた、前記符号化されたビデオビットストリーム中の情報に基づいて決定され、
前記補間フィルタのサブセットから補間フィルタを決定することと、
前記コーディングユニットを再構成するために、前記参照フレームおよび前記補間フィルタを使用することと
を含む動作を実行させる、コンピュータ可読媒体。

20

[C 3 5] ビデオデータを復号するための装置であって、
符号化されたビデオビットストリームを取得するための手段と、
前記符号化されたビデオビットストリームから現在のフレーム中のコーディングユニットのための参照フレームを決定するための手段と、
補間フィルタのセットから補間フィルタのサブセットを決定するための手段と、ここにおいて、前記補間フィルタのサブセットは、前記コーディングユニットに関連付けられた、前記符号化されたビデオビットストリーム中の情報に基づいて決定され、
前記補間フィルタのサブセットから補間フィルタを決定するための手段と、
前記コーディングユニットを再構成するために、前記参照フレームおよび前記補間フィルタを使用するための手段と
を備える装置。

30

40

50

【図 5】

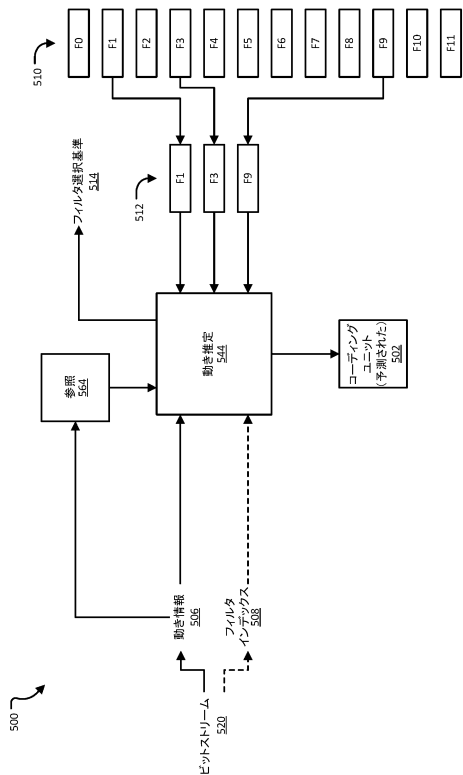


FIG. 5

【図 6】

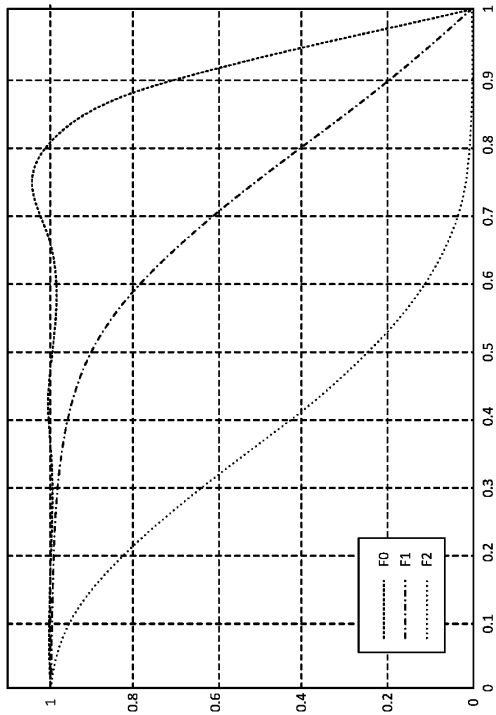


FIG. 6

【図 7】

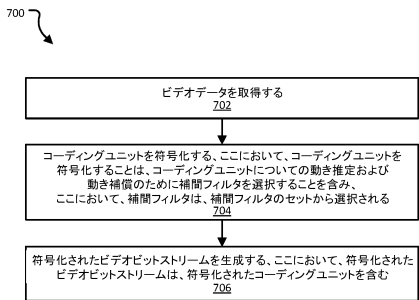


FIG. 7

【図 8】

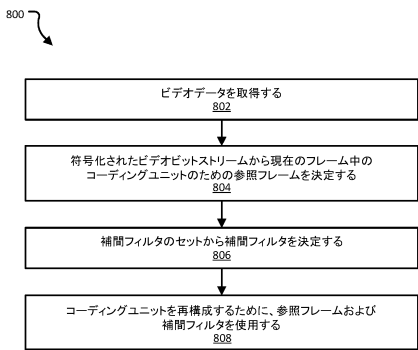


FIG. 8

10

20

30

40

50

フロントページの続き

(33)優先権主張国・地域又は機関
米国(US)

前置審査

2 1 2 1 - 1 7 1 4、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5

(72)発明者 ジャン、カイ

アメリカ合衆国、カリフォルニア州 9 2 1 2 1 - 1 7 1 4、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5

(72)発明者 チェン、ジャンレ

アメリカ合衆国、カリフォルニア州 9 2 1 2 1 - 1 7 1 4、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5

(72)発明者 カルチェビチ、マルタ

アメリカ合衆国、カリフォルニア州 9 2 1 2 1 - 1 7 1 4、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5

審査官 坂東 大五郎

(56)参考文献 特表 2 0 1 3 - 5 2 4 6 7 8 (J P , A)

特開 2 0 0 3 - 2 1 9 4 2 6 (J P , A)

(58)調査した分野 (Int.Cl. , D B 名)

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