

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号  
特許第7384939号  
(P7384939)

(45)発行日 令和5年11月21日(2023.11.21)

(24)登録日 令和5年11月13日(2023.11.13)

(51)国際特許分類 F I  
H 0 4 N 19/577(2014.01) H 0 4 N 19/577

請求項の数 32 (全59頁)

(21)出願番号	特願2021-576697(P2021-576697)	(73)特許権者	504161984 ホアウェイ・テクノロジーズ・カンパニー・リミテッド 中華人民共和国・518129・グアン ドン・シェンツェン・ロンガン・ディス トリクト・バンティアン・(番地なし) ・ホアウェイ・アドミニストレーション ・ビルディング
(86)(22)出願日	令和2年6月23日(2020.6.23)	(74)代理人	100110364 弁理士 実広 信哉
(65)公表番号	特表2022-540771(P2022-540771 A)	(74)代理人	100133569 弁理士 野村 進
(43)公表日	令和4年9月20日(2022.9.20)	(72)発明者	スリラム・セトゥラマン インド・カルナータカ・560017・ バンガロール・ピナヤカナガー・ピー - 最終頁に続く
(86)国際出願番号	PCT/CN2020/097762		
(87)国際公開番号	WO2020/259507		
(87)国際公開日	令和2年12月30日(2020.12.30)		
審査請求日	令和4年2月22日(2022.2.22)		
(31)優先権主張番号	201931025013		
(32)優先日	令和1年6月24日(2019.6.24)		
(33)優先権主張国・地域又は機関	インド(IN)		

(54)【発明の名称】 双予測のオプティカルフロー計算および双予測補正におけるブロックレベル境界サンプル勾配計算のための整数グリッド参照サンプルの位置を計算するための方法

(57)【特許請求の範囲】

【請求項1】

符号化デバイスまたは復号デバイスによって実施される、現在のブロックの双方向のオプティカルフロー予測において使用される予測されたブロックの拡張された領域に属するサンプル位置からのサンプル値に基づく前記双方向のオプティカルフロー予測のための方法であって、

前記現在のブロックの動きベクトル(MV)を取得するステップと、

前記MVに基づいて参照ピクチャ内の前記予測されたブロックの前記拡張された領域内のサブピクセルサンプル位置を取得するステップと、

前記参照ピクチャ内の整数ピクセルサンプル位置を取得するために、前記取得されたサブピクセルサンプル位置を処理するステップと、

10

前記参照ピクチャ内の前記取得された整数ピクセルサンプル位置における参照サンプル値を参照することによって、双方向のオプティカルフロー予測を実行するステップとを含み、

前記取得されたサブピクセルサンプル位置を処理するステップは、

前記取得されたサブピクセルサンプル位置を、水平方向および垂直方向において別々に、それぞれの方向におけるそれぞれの最も近い整数ピクセルサンプル位置に丸めるステップと、

前記水平方向および前記垂直方向における前記最も近い整数ピクセルサンプル位置に基づいて、前記参照ピクチャ内の前記整数ピクセルサンプル位置を取得するステップと

20

を含む、方法。

【請求項 2】

前記参照ピクチャ内の前記取得された整数ピクセルサンプル位置における前記参照サンプル値は、前記双方向のオプティカルフロー予測における前記サブピクセルサンプル位置からの前記サンプル値と称される、請求項1に記載の方法。

【請求項 3】

前記取得されたサブピクセルサンプル位置を処理するステップは、前記参照ピクチャ内の前記整数ピクセルサンプル位置を取得するために、前記取得されたサブピクセルサンプル位置を丸めるステップを含む、請求項1または2に記載の方法。

【請求項 4】

前記方法は、

前記双方向のオプティカルフロー予測を実行するステップの前に、前記参照ピクチャ内の前記取得された整数ピクセルサンプル位置における前記参照サンプル値を、前記双方向のオプティカルフロー予測において使用される予測サンプル値としてフェッチするステップをさらに含み、

参照サンプル値を参照して前記双方向のオプティカルフロー予測を実行するステップは、前記予測サンプル値を使用して双方向のオプティカルフロー予測を実行するステップを含む、

請求項1から3のいずれか一項に記載の方法。

【請求項 5】

前記取得されたサブピクセルサンプル位置を前記水平方向および前記垂直方向において別々に丸めるステップは、

—それぞれの方向における前記取得されたサブピクセルサンプル位置の小数部分を閾値と比較するステップを含み、

前記小数部分が前記閾値以上である場合、前記それぞれの方向における最も近いより高い整数ピクセルサンプル位置は、前記それぞれの方向における前記最も近い整数ピクセルサンプル位置として取得され、

前記小数部分が前記閾値よりも小さい場合、前記それぞれの方向における最も近いより低い整数ピクセルサンプル位置は、前記それぞれの方向における前記最も近い整数ピクセルサンプル位置として取得される、

請求項1に記載の方法。

【請求項 6】

前記それぞれの方向における前記最も近いより高い整数ピクセルサンプル位置は、前記取得されたサブピクセルサンプル位置の前記小数部分を破棄することによって取得された前記それぞれの方向における切り捨てられた整数ピクセルサンプル位置に1のオフセットを加えることによって取得され、

前記それぞれの方向における前記最も近いより低い整数ピクセルサンプル位置は、前記取得されたサブピクセルサンプル位置の前記小数部分を破棄することによって取得された前記それぞれの方向における前記切り捨てられた整数ピクセルサンプル位置に0のオフセットを加えることによって取得される、

請求項5に記載の方法。

【請求項 7】

前記閾値は、 $1/P$ ペルのサブピクセル精度の場合、 $P/2$ によって与えられる、請求項5または6に記載の方法。

【請求項 8】

前記閾値は、 $1/16$ ペルのサブピクセル精度の場合、8である、請求項7に記載の方法。

【請求項 9】

前記取得されたサブピクセルサンプル位置を前記水平方向および前記垂直方向において別々に丸めるステップは、

—それぞれの方向における前記取得されたサブピクセルサンプル位置の成分にシフト演算

10

20

30

40

50

を適用するステップを含む、  
請求項1に記載の方法。

【請求項10】

前記参照ピクチャ内の前記整数ピクセルサンプル位置(x\_pos, y\_pos)は、  

$$x\_pos = (x\_spel\_pos + (1 \quad (shift\_val - 1))) \quad shift\_val$$

$$y\_pos = (y\_spel\_pos + (1 \quad (shift\_val - 1))) \quad shift\_val$$
 に従って取得され、式中、(x\_spel\_pos, y\_spel\_pos)は、前記取得されたサブピクセル  
 サンプル位置であり、shift\_valは、使用されたサブピクセル精度に従って選択される  
 、  
 請求項9に記載の方法。

10

【請求項11】

半ピクセル、1/4ピクセル、ピクセルの1/8、またはピクセルの1/16のサブピクセル精度の場合、そ  
 れぞれ、1、2、3、または4のshift\_valが使用される、請求項10に記載の方法。

【請求項12】

前記現在のブロックの前記MVは、初期MVに基づいて動きベクトルの改善(MVR)によっ  
 て取得された改善されたMVである、請求項1から11のいずれか一項に記載の方法。

【請求項13】

前記MVRは、前記初期MVに関する整数距離改善段階を含み、  
 前記MVRの前記整数距離改善段階が前記初期MVに対する前記改善されたMVの非ゼロの  
 変位をもたらす場合、前記参照ピクチャ内の前記整数ピクセルサンプル位置は、前記取得  
 されたサブピクセルサンプル位置を丸めることによって取得される、  
 請求項12に記載の方法。

20

【請求項14】

前記MVRの整数距離改善段階が前記初期MVに対する前記改善されたMVのゼロの変位を  
 もたらす場合、前記参照ピクチャ内の前記整数ピクセルサンプル位置は、前記取得された  
 サブピクセルサンプル位置の小数部分を破棄することによって取得される、請求項12に記  
 載の方法。

【請求項15】

前記MVRの前記整数距離改善段階が前記初期MVに対する前記改善されたMVのゼロの変  
 位をもたらす場合、前記参照ピクチャ内の前記整数ピクセルサンプル位置(x\_pos, y\_pos)  
 は、

30

$$x\_pos = x\_spel\_pos \quad shift\_val$$

$$y\_pos = y\_spel\_pos \quad shift\_val$$

に従って取得され、式中、(x\_spel\_pos, y\_spel\_pos)は、前記取得されたサブピクセル  
 サンプル位置であり、shift\_valは、使用されたサブピクセル精度に従って選択される  
 、  
 請求項13に記載の方法。

【請求項16】

半ピクセル、1/4ピクセル、ピクセルの1/8、またはピクセルの1/16のサブピクセル精度の場合、そ  
 れぞれ、1、2、3、または4のshift\_valが使用される、請求項15に記載の方法。

40

【請求項17】

前記現在のブロックは、現在のコーディングブロック、または現在のコーディングサブ  
 ブロック、または現在の予測ブロックである、請求項1から16のいずれか一項に記載の方  
 法。

【請求項18】

前記参照ピクチャ内の前記取得された整数ピクセルサンプル位置における前記参照サン  
 プル値は、前記参照ピクチャ内の前記予測されたブロックの境界に対応する境界サンプル  
 勾配の計算において使用される、請求項1から17のいずれか一項に記載の方法。

【請求項19】

請求項1から18のいずれか一項に記載の方法を実行するための処理回路を含む、エンコ

50

ーダ(20)。

【請求項 2 0】

請求項1から18のいずれか一項に記載の方法を実行するための処理回路を含む、デコーダ(30)。

【請求項 2 1】

コンピュータによって実行されるときに、前記コンピュータに、請求項1から18のいずれか一項に記載の方法を行わせる命令を含む、コンピュータプログラム。

【請求項 2 2】

1つまたは複数のプロセッサと、

前記1つまたは複数のプロセッサに結合され、前記1つまたは複数のプロセッサによって 10  
実行するための命令を記憶する非一時的コンピュータ可読記憶媒体と

を含むエンコーダ(20)であって、前記命令は、前記1つまたは複数のプロセッサによって  
実行されるときに、請求項1から18のいずれか一項に記載の方法を行うように前記エン  
コーダを構成する、

エンコーダ(20)。

【請求項 2 3】

1つまたは複数のプロセッサと、

前記1つまたは複数のプロセッサに結合され、前記1つまたは複数のプロセッサによって  
実行するための命令を記憶する非一時的コンピュータ可読記憶媒体と

を含むデコーダ(30)であって、前記命令は、前記1つまたは複数のプロセッサによって実 20  
行されるときに、請求項1から18のいずれか一項に記載の方法を行うように前記デコーダ  
を構成する、

デコーダ(30)。

【請求項 2 4】

現在のブロックの双方向のオプティカルフロー予測を実行するように構成された予測ユ  
ニットと、

前記現在のブロックの動きベクトル(MV)を取得するように構成された第1の取得ユニ  
ットと、

前記MVに基づいて参照ピクチャ内の予測されたブロックの拡張された領域内のサブピ  
クセルサンプル位置を取得するように構成された第2の取得ユニットと、 30

前記参照ピクチャ内の整数ピクセルサンプル位置を取得するために、前記取得されたサ  
ブピクセルサンプル位置を処理するように構成された処理ユニットと

を含むエンコーダ(20)であって、

前記予測ユニットは、前記参照ピクチャ内の前記取得された整数ピクセルサンプル位置  
における参照サンプル値を参照することによって前記双方向のオプティカルフロー予測を  
実行するように構成され、

前記取得されたサブピクセルサンプル位置を処理することは、

前記取得されたサブピクセルサンプル位置を、水平方向および垂直方向において別々に  
、それぞれの方向におけるそれぞれの最も近い整数ピクセルサンプル位置に丸めることと、

前記水平方向および前記垂直方向における前記最も近い整数ピクセルサンプル位置に基  
づいて、前記参照ピクチャ内の前記整数ピクセルサンプル位置を取得することを含む、 40  
エンコーダ(20)。

【請求項 2 5】

前記取得されたサブピクセルサンプル位置を処理することは、前記参照ピクチャ内の前  
記整数ピクセルサンプル位置を取得するために、前記取得されたサブピクセルサンプル位  
置を丸めることを含む、請求項24に記載のエンコーダ(20)。

【請求項 2 6】

前記取得されたサブピクセルサンプル位置を前記水平方向および前記垂直方向において  
別々に丸めることは、

それぞれの方向における前記取得されたサブピクセルサンプル位置の小数部分を閾値と 50

比較することを含み、

前記小数部分が前記閾値以上である場合、前記それぞれの方向における最も近いより高い整数ピクセルサンプル位置は、前記それぞれの方向における前記最も近い整数ピクセルサンプル位置として取得され、

前記小数部分が前記閾値よりも小さい場合、前記それぞれの方向における最も近いより低い整数ピクセルサンプル位置は、前記それぞれの方向における前記最も近い整数ピクセルサンプル位置として取得される、

請求項24に記載のエンコーダ(20)。

【請求項27】

前記それぞれの方向における前記最も近いより高い整数ピクセルサンプル位置は、前記取得されたサブピクセルサンプル位置の前記小数部分を破棄することによって取得された前記それぞれの方向における切り捨てられた整数ピクセルサンプル位置に1のオフセットを加えることによって取得され、

10

前記それぞれの方向における前記最も近いより低い整数ピクセルサンプル位置は、前記取得されたサブピクセルサンプル位置の前記小数部分を破棄することによって取得された前記それぞれの方向における前記切り捨てられた整数ピクセルサンプル位置に0のオフセットを加えることによって取得される、

請求項26に記載のエンコーダ(20)。

【請求項28】

現在のブロックの双方向のオプティカルフロー予測を実行するように構成された予測ユニットと、

20

符号化されたビデオのビットストリームから前記現在のブロックの動きベクトル(MV)を取得するように構成された第1の取得ユニットと、

前記MVに基づいて参照ピクチャ内の予測されたブロックの拡張された領域内のサブピクセルサンプル位置を取得するように構成された第2の取得ユニットと、

前記参照ピクチャ内の整数ピクセルサンプル位置を取得するために、前記取得されたサブピクセルサンプル位置を処理するように構成された処理ユニットとを含むデコーダ(30)であって、

前記予測ユニットは、前記参照ピクチャ内の前記取得された整数ピクセルサンプル位置における参照サンプル値を参照することによって前記双方向のオプティカルフロー予測を実行するように構成され、

30

前記取得されたサブピクセルサンプル位置を処理することは、

前記取得されたサブピクセルサンプル位置を、水平方向および垂直方向において別々に、それぞれの方向におけるそれぞれの最も近い整数ピクセルサンプル位置に丸めることと、

前記水平方向および前記垂直方向における前記最も近い整数ピクセルサンプル位置に基づいて、前記参照ピクチャ内の前記整数ピクセルサンプル位置を取得することを含む、デコーダ(30)。

【請求項29】

前記取得されたサブピクセルサンプル位置を処理することは、前記参照ピクチャ内の前記整数ピクセルサンプル位置を取得するために、前記取得されたサブピクセルサンプル位置を丸めることを含む、請求項28に記載のデコーダ(30)。

40

【請求項30】

前記取得されたサブピクセルサンプル位置を前記水平方向および前記垂直方向において別々に丸めることは、

それぞれの方向における前記取得されたサブピクセルサンプル位置の小数部分を閾値と比較することを含み、

前記小数部分が前記閾値以上である場合、前記それぞれの方向における最も近いより高い整数ピクセルサンプル位置は、前記それぞれの方向における前記最も近い整数ピクセルサンプル位置として取得され、

前記小数部分が前記閾値よりも小さい場合、前記それぞれの方向における最も近いより

50

低い整数ピクセルサンプル位置は、前記それぞれの方向における前記最も近い整数ピクセルサンプル位置として取得される、  
請求項28に記載のデコーダ(30)。

【請求項31】

前記それぞれの方向における前記最も近いより高い整数ピクセルサンプル位置は、前記取得されたサブピクセルサンプル位置の前記小数部分を破棄することによって取得された前記それぞれの方向における切り捨てられた整数ピクセルサンプル位置に1のオフセットを加えることによって取得され、

前記それぞれの方向における前記最も近いより低い整数ピクセルサンプル位置は、前記取得されたサブピクセルサンプル位置の前記小数部分を破棄することによって取得された前記それぞれの方向における前記切り捨てられた整数ピクセルサンプル位置に0のオフセットを加えることによって取得される、  
請求項30に記載のデコーダ(30)。

【請求項32】

コンピュータ可読記憶媒体であって、前記コンピュータ可読記憶媒体は、プロセッサによって実行されるコンピュータプログラムを記憶し、前記コンピュータプログラムが前記プロセッサによって実行されるとき、前記プロセッサは請求項1から18のいずれか一項に記載の方法を行う、コンピュータ可読記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

関連出願の相互参照

本出願は、2019年6月24日にインド特許庁に出願したインド特許出願第IN201931025013号に基づく優先権を主張するものであり、その開示は、参照によりその全体が本明細書に組み込まれる。

【0002】

本出願の実施形態は、概して、ピクチャ処理の分野に関し、より詳細には、オプティカルフローの改善に関する。

【背景技術】

【0003】

ビデオコーディング(ビデオ符号化および復号)は、広範なデジタルビデオアプリケーション、たとえば、ブロードキャストデジタルTV、インターネットおよびモバイルネットワーク上のビデオ送信、ビデオチャットのようなリアルタイム会話アプリケーション、テレビ会議、DVDおよびブルーレイディスク、ビデオコンテンツ獲得および編集システム、ならびにセキュリティアプリケーションのカムコーダにおいて使用される。

【0004】

比較的短いビデオでさえも描くために必要とされるビデオデータの量はかなり多くなり得、それが、データが限られた帯域幅の容量を有する通信ネットワークを介してストリーミングされるかまたはそれ以外の方法で伝達されるべきであるときに困難をもたらす可能性がある。したがって、ビデオデータは、概して、現代の通信ネットワークを介して伝達される前に圧縮される。メモリリソースが限られている可能性があるため、ビデオが記憶装置に記憶されるとき、ビデオのサイズも問題となり得る。多くの場合、ビデオ圧縮デバイスは、送信または記憶の前にビデオデータをコーディングするために送信元においてソフトウェアおよび/またはハードウェアを使用し、それによって、デジタルビデオ画像を表現するために必要とされるデータの量を削減する。そして、圧縮されたデータが、ビデオデータを復号するビデオ解凍デバイスによって送信先において受信される。限られたネットワークリソースおよびより高いビデオ品質のますます増加する需要によって、ピクチャ品質をほとんどまたはまったく犠牲にせずに圧縮比を高める改善された圧縮および解凍技術が、望ましい。

【発明の概要】

10

20

30

40

50

**【課題を解決するための手段】****【0005】**

本出願の実施形態は、独立請求項による符号化および復号のための装置および方法を提供する。

**【0006】**

上述のおよび他の目的は、独立請求項の主題により達成される。さらなる実装の形態は、従属請求項、明細書、および図面から明らかである。

**【0007】**

本開示の第1の実施形態は、符号化デバイスまたは復号デバイスによって実施される、現在のブロックの双方向のオプティカルフロー予測において使用される予測されたブロックの拡張された領域に属するサンプル位置からのサンプル値に基づく双方向のオプティカルフロー予測のための方法を提供し、方法は、現在のブロックの動きベクトル(MV)を取得するステップと、MVに基づいて参照ピクチャ内の予測されたブロックの拡張された領域内のサブピクセルサンプル位置を取得するステップと、参照ピクチャ内の整数ピクセルサンプル位置を取得するために、取得されたサブピクセルサンプル位置を処理するステップと、参照ピクチャ内の取得された整数ピクセルサンプル位置における参照サンプル値を参照することによって、双方向のオプティカルフロー予測を実行するステップとを含む。

10

**【0008】**

参照ピクチャ内の整数ピクセルサンプル位置を取得するために、取得されたサブピクセルサンプル位置を処理するステップは、双方向のオプティカルフロー(BDOF)予測のためのメモリフットプリントを低減し、拡張されたサンプル値(すなわち、サブブロックに対応するM×Nサンプル位置の外側にあるサンプル位置)がそれぞれの参照サンプルの動き補償補間によって取得されないのを、ハードウェア設計においてサブブロックレベルパイプラインが確立されることを可能にする可能性がある。

20

**【0009】**

参照ピクチャ内の取得された整数ピクセルサンプル位置における参照サンプル値は、双方向のオプティカルフロー予測におけるサブピクセルサンプル位置からのサンプル値と称されることがある。

**【0010】**

取得されたサブピクセルサンプル位置を処理するステップは、参照ピクチャ内の整数ピクセルサンプル位置を取得するために、取得されたサブピクセルサンプル位置を丸めるステップを含む可能性がある。

30

**【0011】**

サブピクセル位置を水平方向および垂直方向において丸めることによって、境界勾配は、より正確になる可能性がある。これは、これらのサンプル勾配に基づいて計算されたオプティカルフロー、および計算されたオプティカルフローを使用して適用された補正をも改善する可能性がある。これは、拡張されたサンプル位置のための整数サンプル値を選択するときに、水平方向および垂直方向における小数オフセットを無視する以前の方法与比較して、一貫したコーディングゲインを提供する可能性がある。同時に、水平および垂直の丸めオフセットの追加は、複雑さにおける唯一の増加であるので、複雑さは、あまり増加しない可能性がある。

40

**【0012】**

取得されたサブピクセルサンプル位置を処理するステップは、取得されたサブピクセルサンプル位置を、水平方向および垂直方向において別々に、それぞれの方向におけるそれぞれの最も近い整数ピクセルサンプル位置に丸めるステップと、水平方向および垂直方向における最も近い整数ピクセルサンプル位置に基づいて、参照ピクチャ内の整数ピクセルサンプル位置を取得するステップとを含む可能性がある。

**【0013】**

方法は、双方向のオプティカルフロー予測を実行するステップの前に、参照ピクチャ内の取得された整数ピクセルサンプル位置における参照サンプル値を、双方向のオプティカ

50

ルフロー予測において使用される予測サンプル値としてフェッチするステップをさらに含む可能性があり、参照サンプル値を参照して双方向のオプティカルフロー予測を実行するステップは、予測サンプル値を使用して双方向のオプティカルフロー予測を実行するステップを含む。

【0014】

実装によれば、取得されたサブピクセルサンプル位置を水平方向および垂直方向において別々に丸めるステップは、それぞれの方向における取得されたサブピクセルサンプル位置の小数部分を閾値と比較するステップを含む可能性があり、小数部分が閾値以上である場合、それぞれの方向における最も近いより高い整数ピクセルサンプル位置は、それぞれの方向における最も近い整数ピクセルサンプル位置として取得され、小数部分が閾値よりも小さい場合、それぞれの方向における最も近いより低い整数ピクセルサンプル位置は、それぞれの方向における最も近い整数ピクセルサンプル位置として取得される。

10

【0015】

それぞれの方向における最も近いより高い整数ピクセルサンプル位置は、取得されたサブピクセルサンプル位置の小数部分を破棄することによって取得されたそれぞれの方向における切り捨てられた整数ピクセルサンプル位置に1のオフセットを加えることによって取得される可能性があり、それぞれの方向における最も近いより低い整数ピクセルサンプル位置は、取得されたサブピクセルサンプル位置の小数部分を破棄することによって取得されたそれぞれの方向における切り捨てられた整数ピクセルサンプル位置に0のオフセットを加えることによって取得される可能性がある。

20

【0016】

閾値は、 $1/P$ ペルのサブピクセル精度の場合、 $P/2$ によって与えられる可能性がある。閾値は、 $1/16$ ペルのサブピクセル精度の場合、8である可能性がある。

【0017】

実装によれば、取得されたサブピクセルサンプル位置を水平方向および垂直方向において別々に丸めるステップは、それぞれの方向における取得されたサブピクセルサンプル位置の成分にシフト演算および逆シフト演算を適用するステップを含む可能性がある。

【0018】

参照ピクチャ内の整数ピクセルサンプル位置( $x\_pos$ ,  $y\_pos$ )は、以下の式に従って取得される可能性がある。

30

$$\begin{aligned} x\_pos &= (x\_spel\_pos + (1 \quad (shift\_val - 1))) \quad shift\_val \\ y\_pos &= (y\_spel\_pos + (1 \quad (shift\_val - 1))) \quad shift\_val \end{aligned}$$

式中、( $x\_spel\_pos$ ,  $y\_spel\_pos$ )は、取得されたサブピクセルサンプル位置であり、 $shift\_val$ は、使用されたサブピクセルサンプル精度に従って選択される。半ペル、 $1/4$ ペル、ペルの $1/8$ 、またはペルの $1/16$ のサブピクセル精度の場合、それぞれ、1、2、3、または4の $shift\_val$ が使用される可能性がある。

【0019】

実装によれば、現在のブロックのMVは、初期MVに基づいて動きベクトルの改善(MVR)によって取得された改善されたMVである可能性がある。

【0020】

40

MVRは、初期MVに関する整数距離改善段階を含む可能性があり、MVRの整数距離改善段階が初期MVに対する改善されたMVの非ゼロの変位をもたらす場合、参照ピクチャ内の整数ピクセルサンプル位置は、取得されたサブピクセルサンプル位置を丸めることによって取得される可能性がある。

【0021】

MVRの整数距離改善段階が初期MVに対する改善されたMVのゼロの変位をもたらす場合、参照ピクチャ内の整数ピクセルサンプル位置は、取得されたサブピクセルサンプル位置の小数部分を破棄することによって取得される可能性がある。

【0022】

実装によれば、MVRの整数距離改善段階が初期MVに対する改善されたMVのゼロの変位

50

をもたらす場合、参照ピクチャ内の整数ピクセルサンプル位置(x\_pos, y\_pos)は、以下の式に従って取得される可能性がある。

$$\begin{aligned} x\_pos &= x\_spel\_pos + shift\_val \\ y\_pos &= y\_spel\_pos + shift\_val \end{aligned}$$

式中、(x\_spel\_pos, y\_spel\_pos)は、取得されたサブピクセルサンプル位置であり、shift\_valは、使用されたサブピクセルサンプル精度に従って選択される。半ピクセル、1/4ピクセル、ピクセルの1/8、またはピクセルの1/16のサブピクセルサンプル精度の場合、それぞれ、1、2、3、または4のshift\_valが使用される可能性がある。

【0023】

現在のブロックは、現在のコーディングブロック、または現在のコーディングサブブロック、または現在の予測ブロックである可能性がある。

10

【0024】

参照ピクチャ内の取得された整数ピクセルサンプル位置における参照サンプル値は、参照ピクチャ内の予測されたブロックの境界に対応する境界サンプル勾配の計算において使用される可能性がある。

【0025】

第1の実施形態の態様によれば、第1の実施形態による方法のいずれか1つを実行するための処理回路を含むエンコーダが提供される。

【0026】

第1の実施形態のさらなる態様によれば、第1の実施形態による方法のいずれか1つを実行するための処理回路を含む、デコーダが提供される。

20

【0027】

第1の実施形態のさらなる態様によれば、プログラムがコンピュータによって実行されるときに、コンピュータに第1の実施形態による方法のいずれか1つを行わせる命令を含む、コンピュータプログラム製品が提供される。

【0028】

第1の実施形態のさらなる態様によれば、1つまたは複数のプロセッサと、1つまたは複数のプロセッサに結合され、1つまたは複数のプロセッサによって実行するための命令を記憶する非一時的コンピュータ可読記憶媒体とを含むエンコーダが提供され、命令は、1つまたは複数のプロセッサによって実行されるときに、第1の実施形態による方法のいずれか1つを行うようにエンコーダを構成する。

30

【0029】

第1の実施形態のさらなる態様によれば、1つまたは複数のプロセッサと、1つまたは複数のプロセッサに結合され、1つまたは複数のプロセッサによって実行するための命令を記憶する非一時的コンピュータ可読記憶媒体とを含むデコーダが提供され、命令は、1つまたは複数のプロセッサによって実行されるときに、第1の実施形態による方法のいずれか1つを行うようにデコーダを構成する。

【0030】

第1の実施形態のさらなる態様によれば、現在のブロックの双方向のオプティカルフロー予測を実行するように構成された予測ユニットと、現在のブロックの動きベクトル(MV)を取得するように構成された第1の取得ユニットと、MVに基づいて参照ピクチャ内の予測されたブロックの拡張された領域内のサブピクセルサンプル位置を取得するように構成された第2の取得ユニットと、参照ピクチャ内の整数ピクセルサンプル位置を取得するために、取得されたサブピクセルサンプル位置を処理するように構成された処理ユニットとを含むエンコーダが提供され、予測ユニットは、参照ピクチャ内の取得された整数ピクセルサンプル位置における参照サンプル値を参照することによって双方向のオプティカルフロー予測を実行するように構成される。

40

【0031】

実装によれば、取得されたサブピクセルサンプル位置を処理することは、参照ピクチャ内の整数ピクセルサンプル位置を取得するために、取得されたサブピクセルサンプル位置

50

を丸めることを含む可能性がある。

【 0 0 3 2 】

実装によれば、取得されたサブピクセルサンプル位置を処理することは、取得されたサブピクセルサンプル位置を、水平方向および垂直方向において別々に、それぞれの方向におけるそれぞれの最も近い整数ピクセルサンプル位置に丸めることと、水平方向および垂直方向における最も近い整数ピクセルサンプル位置に基づいて、参照ピクチャ内の整数ピクセルサンプル位置を取得することとを含む可能性がある。

【 0 0 3 3 】

取得されたサブピクセルサンプル位置を水平方向および垂直方向において別々に丸めることは、それぞれの方向における取得されたサブピクセルサンプル位置の小数部分を閾値と比較することを含む可能性があり、小数部分が閾値以上である場合、それぞれの方向における最も近いより高い整数ピクセルサンプル位置は、それぞれの方向における最も近い整数ピクセルサンプル位置として取得される可能性があり、小数部分が閾値よりも小さい場合、それぞれの方向における最も近いより低い整数ピクセルサンプル位置は、それぞれの方向における最も近い整数ピクセルサンプル位置として取得される可能性がある。

【 0 0 3 4 】

それぞれの方向における最も近いより高い整数ピクセルサンプル位置は、取得されたサブピクセルサンプル位置の小数部分を破棄することによって取得されたそれぞれの方向における切り捨てられた整数ピクセルサンプル位置に1のオフセットを加えることによって取得される可能性があり、それぞれの方向における最も近いより低い整数ピクセルサンプル位置は、取得されたサブピクセルサンプル位置の小数部分を破棄することによって取得されたそれぞれの方向における切り捨てられた整数ピクセルサンプル位置に0のオフセットを加えることによって取得される可能性がある。

【 0 0 3 5 】

第1の実施形態のさらなる態様によれば、現在のブロックの双方向のオプティカルフロー予測を実行するように構成された予測ユニットと、符号化されたビデオのビットストリームから現在のブロックの動きベクトル(MV)を取得するように構成された第1の取得ユニットと、MVに基づいて参照ピクチャ内の予測されたブロックの拡張された領域内のサブピクセルサンプル位置を取得するように構成された第2の取得ユニットと、参照ピクチャ内の整数ピクセルサンプル位置を取得するために、取得されたサブピクセルサンプル位置を処理するように構成された処理ユニットとを含み、予測ユニットは、参照ピクチャ内の取得された整数ピクセルサンプル位置における参照サンプル値を参照することによって双方向のオプティカルフロー予測を実行するように構成される、デコーダが提供される。

【 0 0 3 6 】

実装によれば、取得されたサブピクセルサンプル位置を処理することは、参照ピクチャ内の整数ピクセルサンプル位置を取得するために、取得されたサブピクセルサンプル位置を丸めることを含む可能性がある。

【 0 0 3 7 】

実装によれば、取得されたサブピクセルサンプル位置を処理することは、取得されたサブピクセルサンプル位置を、水平方向および垂直方向において別々に、それぞれの方向におけるそれぞれの最も近い整数ピクセルサンプル位置に丸めることと、水平方向および垂直方向における最も近い整数ピクセルサンプル位置に基づいて、参照ピクチャ内の整数ピクセルサンプル位置を取得することとを含む可能性がある。

【 0 0 3 8 】

取得されたサブピクセルサンプル位置を水平方向および垂直方向において別々に丸めることは、それぞれの方向における取得されたサブピクセルサンプル位置の小数部分を閾値と比較することを含む可能性があり、小数部分が閾値以上である場合、それぞれの方向における最も近いより高い整数ピクセルサンプル位置は、それぞれの方向における最も近い整数ピクセルサンプル位置として取得される可能性があり、小数部分が閾値よりも小さい場合、それぞれの方向における最も近いより低い整数ピクセルサンプル位置は、それぞれ

10

20

30

40

50

の方向における最も近い整数ピクセルサンプル位置として取得される可能性がある。

【 0 0 3 9 】

それぞれの方向における最も近いより高い整数ピクセルサンプル位置は、取得されたサブピクセルサンプル位置の小数部分を破棄することによって取得されたそれぞれの方向における切り捨てられた整数ピクセルサンプル位置に1のオフセットを加えることによって取得される可能性があり、それぞれの方向における最も近いより低い整数ピクセルサンプル位置は、取得されたサブピクセルサンプル位置の小数部分を破棄することによって取得されたそれぞれの方向における切り捨てられた整数ピクセルサンプル位置に0のオフセットを加えることによって取得される可能性がある。

【 0 0 4 0 】

第1の実施形態のさらなる態様によれば、第1の実施形態の方法のいずれか1つを使用することによって取得されたビットストリームを記憶する、記憶媒体が提供される。

【 0 0 4 1 】

本開示の第2の実施形態は、現在のコーディングブロックの双方向のオプティカルフロー予測において使用される予測されたブロックの拡張された領域に属するサンプル位置に関する予測されたサンプル値を取得するために使用される方法を提供し、方法は、コーディングブロックの拡張された領域内の予測されたサンプルに対応する第1のMVを取得するステップと、第1のMVの小数部分を破棄することによって第2のMVを取得するステップと、第2のMVに基づいて参照ピクチャ内の拡張された領域の第1の整数ピクセル位置を取得するステップと、第1のMVに基づいて分数オフセットベクトルを決定するステップと、第1の整数ピクセル位置と分数オフセットベクトルとに基づいて参照ピクチャ内の第2の整数ピクセル位置を取得するステップとを含み、第2の整数ピクセル位置は第1の整数ピクセル位置に位置オフセットを加えたものに設定され、位置オフセットは、分数オフセットベクトルの各成分と閾値とを比較し、参照ピクチャ内の第2の整数ピクセル位置内のピクセル値を使用することによって、予測されたサンプルをさらにパディングすることによって決定される。

【 0 0 4 2 】

位置オフセットの成分が1であるか0であるかは、小数オフセットベクトルの対応する成分と閾値との間の比較の結果に依存する可能性があり、成分は、X成分とY成分とを含む。

【 0 0 4 3 】

閾値は、Kである可能性があり、位置オフセットの成分は、小数オフセットの対応する成分がK以上である場合、1に設定され、そうでない場合、0に設定される可能性がある。Kは、7に等しい可能性がある。

【 0 0 4 4 】

第2の実施形態の態様によれば、第2の実施形態による方法のいずれか1つを実行するための処理回路を含む、エンコーダまたはデコーダが提供される。

【 0 0 4 5 】

第2の実施形態のさらなる態様によれば、第2の実施形態による方法のいずれか1つを実行するためのプログラムコードを含む、コンピュータプログラム製品が提供される。

【 0 0 4 6 】

第2の実施形態のさらなる態様によれば、1つまたは複数のプロセッサと、プロセッサに結合され、プロセッサによって実行するためのプログラミングを記憶する非一時的コンピュータ可読記憶媒体とを含む、デコーダまたはエンコーダが提供され、プログラミングは、プロセッサによって実行されるときに、第2の実施形態による方法のいずれか1つを行うようにデコーダまたはエンコーダを構成する。

【 0 0 4 7 】

1つまたは複数の実施形態の詳細が、添付の図面および以下の説明に記載されている。他の特徴、目的、および利点は、明細書、図面、および特許請求の範囲から明らかになるであろう。

【 0 0 4 8 】

10

20

30

40

50

以下で、本開示の実施形態が、添付の図および図面を参照してより詳細に説明される。

【図面の簡単な説明】

【0049】

【図1A】本開示の実施形態を実装するように構成されたビデオコーディングシステムの例を示すブロック図である。

【図1B】本開示の実施形態を実装するように構成されたビデオコーディングシステムの別の例を示すブロック図である。

【図2】本開示の実施形態を実装するように構成されたビデオエンコーダの例を示すブロック図である。

【図3】本開示の実施形態を実装するように構成されたビデオデコーダの例示的な構造を示すブロック図である。

10

【図4】符号化装置または復号装置の例を示すブロック図である。

【図5】符号化装置または復号装置の別の例を示すブロック図である。

【図6】BDOPの勾配計算のための整数サンプルフェッチプロセスにおいて使用される参照ピクチャ内の方向の1つにおけるサブピクセル位置丸めプロセスを示す図である。

【図7】BDOPの勾配計算のための整数サンプルフェッチプロセスにおける参照ピクチャ内のサブピクセル位置丸めプロセスを使用することによる整数位置を示す図である。

【図8】本開示の実施形態によるビデオ符号化/復号の方法の流れ図である。

【図9】本開示の実施形態による符号化/復号装置の例を示すブロック図である。

【発明を実施するための形態】

20

【0050】

以下で、同一の参照符号は、別途明記されない場合、同一のまたは少なくとも機能的に等価な特徴を指す。

【0051】

以下の説明においては、本開示の一部を形成し、本開示の実施形態の特定の態様または本開示の実施形態が使用される可能性がある特定の態様を例として示す添付の図面が参照される。本開示の実施形態は、他の態様において使用され、図面に示されない構造的または論理的变化を含む可能性があることが理解される。したがって、以下の詳細な説明は、限定的意味に理解されるべきでなく、本開示の範囲は、添付の特許請求の範囲によって定義される。

30

【0052】

たとえば、説明される方法に関連する開示は、方法を実行するように構成された対応するデバイスまたはシステムにも当てはまる可能性があり、その逆の可能性もあることが理解される。たとえば、1つまたは複数の特定のステップが説明される場合、対応するデバイスは、説明される1つまたは複数のステップを実行するための1つまたは複数のユニット、たとえば、機能ユニット(たとえば、1つもしくは複数のステップを実行する1つのユニット、または複数のステップのうちの一つもしくは複数を実行する複数のユニット)を、たとえそのような1つまたは複数のユニットが明示的に説明されないかまたは図に示されないとしても含む可能性がある。一方、たとえば、特定の装置が1つまたは複数のユニット、たとえば、機能ユニットに基づいて説明される場合、対応する方法は、1つまたは複数のユニットの機能を実行するための1つのステップ(たとえば、1つもしくは複数のユニットの機能を実行する1つのステップ、または複数のユニットのうちの一つもしくは複数の機能をそれぞれが実行する複数のステップ)を、たとえそのような1つまたは複数のステップが明示的に説明されないかまたは図に示されないとしても含む可能性がある。さらに、本明細書において説明される様々な例示的な実施形態および/または態様の特徴は、そうでないことが明記されない限り互いに組み合わせられる可能性があることが理解される。

40

【0053】

ビデオコーディングは、概して、ビデオまたはビデオシーケンスを形成するピクチャのシーケンスの処理を指す。用語「ピクチャ」の代わりに、用語「フレーム」または「画像

50

」が、ビデオコーディングの分野において同義語として使用される可能性がある。ビデオコーディング(または概してコーディング)は、2つの部分、ビデオ符号化およびビデオ復号を含む。ビデオ符号化は、送信元の側で実行され、概して、(より効率的な記憶および/または送信のために)ビデオピクチャを表現するために必要とされるデータの量を減らすために元のビデオピクチャを(たとえば、圧縮によって)処理することを含む。ビデオ復号は、送信先の側で実行され、概して、ビデオピクチャを再構築するためにエンコーダと比べて逆の処理を含む。ビデオピクチャ(または概してピクチャ)の「コーディング」に言及する実施形態は、ビデオピクチャまたはそれぞれのビデオシーケンスの「符号化」または「復号」に関する理解される。符号化部分と復号部分との組合せは、コーデック(コーディングおよびデコーディング)とも称される。

10

**【0054】**

可逆ビデオコーディングの場合、(記憶または送信中に送信損失または他のデータ損失が発生しないと仮定して)元のビデオピクチャが再構築されることが可能であり、つまり、再構築されたビデオピクチャは元のビデオピクチャと同じ品質を有する。不可逆ビデオコーディングの場合、ビデオピクチャを表現するデータの量を減らすために、たとえば、量子化によるさらなる圧縮が実行され、これは、デコーダにおいて完全に再構築され得ず、つまり、再構築されたビデオピクチャの品質は、元のビデオピクチャの品質に比べてより低いまたはより悪い。

**【0055】**

いくつかのビデオコーディング規格は、「不可逆ハイブリッドビデオコーデック」のグループに属する(つまり、サンプル領域(sample domain)における空間および時間予測と変換領域(transform domain)において量子化を適用するための2D変換コーディングとを組み合わせる)。ビデオシーケンスの各ピクチャは、概して、1組の重なり合わないブロックに区分けされ、コーディングは、概して、ブロックレベルで実行される。言い換えると、エンコーダにおいて、ビデオは、概して、たとえば、空間(イントラピクチャ)予測および/または時間(インターピクチャ)予測を使用して予測ブロック(prediction block)を生成し、現在のブロック(現在処理されている/処理されるブロック)から予測ブロックを差し引いて残差ブロックを取得し、残差ブロックを変換し、変換領域において残差ブロックを量子化して送信されるデータの量を削減する(圧縮)ことによってブロック(ビデオブロック)レベルで処理され、つまり、符号化され、一方、デコーダにおいては、表現するために現在のブロックを再構築するために、エンコーダと比べて逆の処理が、符号化されたまたは圧縮されたブロックに適用される。さらに、エンコーダは、後続のブロックを処理する、つまり、コーディングするために両方が同一の予測(たとえば、イントラおよびインター予測)ならびに/または再構築を生成するようにデコーダの処理ループを複製する。

20

30

**【0056】**

以下で、ビデオコーディングシステム10、ビデオエンコーダ20およびビデオデコーダ30の実施形態が、図1から図3に基づいて説明される。

**【0057】**

図1Aは、本出願の技術を利用する可能性がある例示的なコーディングシステム10、たとえば、ビデオコーディングシステム10(または短くコーディングシステム10)を示す概略的なブロック図である。ビデオコーディングシステム10のビデオエンコーダ20(または短くエンコーダ20)およびビデオデコーダ30(または短くデコーダ30)は、本出願において説明される様々な例による技術を実行するように構成される可能性があるデバイスの例を示す。

40

**【0058】**

図1Aに示されるように、コーディングシステム10は、符号化されたピクチャデータ21を復号するために、たとえば、送信先デバイス14に符号化されたピクチャデータ21を提供するように構成された送信元デバイス12を含む。

**【0059】**

送信元デバイス12は、エンコーダ20を含み、追加的に、つまり、任意選択で、ピクチャ

50

ャソース16、プリプロセッサ(または前処理ユニット)18、たとえば、ピクチャプリプロセッサ18、および通信インターフェースまたは通信ユニット22を含む可能性がある。

【0060】

ピクチャソース16は、任意の種類のパクチャ撮影デバイス、たとえば、実世界のピクチャを撮影するためのカメラ、ならびに/または任意の種類のパクチャ生成デバイス、たとえば、コンピュータによってアニメーションされるピクチャを生成するためのコンピュータグラフィックスプロセッサ、または実世界のピクチャ、コンピュータによって生成されたピクチャ(たとえば、画面コンテンツ(screen content)、仮想現実(VR)ピクチャ)、および/もしくはそれらの任意の組合せ(たとえば、拡張現実(AR)ピクチャ)を取得および/もしくは提供するための任意の種類のパクチャの他のデバイスを含むかまたはそのようなデバイスである可能性がある。ピクチャソースは、上述のパクチャのいずれかを記憶するための任意の種類のパクチャまたはストレージである可能性がある。

10

【0061】

プリプロセッサ18および前処理ユニット18によって実行される処理と区別して、ピクチャまたはピクチャデータ17は、生ピクチャまたは生ピクチャデータ17とも称される可能性がある。

【0062】

プリプロセッサ18は、(生)ピクチャデータ17を受け取り、ピクチャデータ17に対して前処理を実行して前処理されたピクチャ19または前処理されたピクチャデータ19を取得するように構成される可能性がある。プリプロセッサ18によって実行される前処理は、たとえば、トリミング、(たとえば、RGBからYCbCrへの)カラーフォーマット変換、色補正、または雑音除去を含む可能性がある。前処理ユニット18は、任意の構成要素である可能性があることが理解され得る。

20

【0063】

ビデオエンコーダ20は、前処理されたピクチャデータ19を受け取り、符号化されたピクチャデータ21を提供するように構成される可能性がある(さらなる詳細が、下で、たとえば、図2に基づいて説明される)。

【0064】

送信元デバイス12の通信インターフェース22は、符号化されたピクチャデータ21を受け取り、符号化されたピクチャデータ21(またはその任意のさらに処理されたバージョン)を、記憶するかまたは直接再構築するために別のデバイス、たとえば、送信先デバイス14または任意の他のデバイスに通信チャネル13を介して送信するように構成される可能性がある。

30

【0065】

送信先デバイス14は、デコーダ30(たとえば、ビデオデコーダ30)を含み、追加的に、つまり、任意選択で、通信インターフェースまたは通信ユニット28、ポストプロセッサ32(または後処理ユニット32)、およびディスプレイデバイス34を含む可能性がある。

【0066】

送信先デバイス14の通信インターフェース28は、たとえば、送信元デバイス12から直接、または任意の他のソース、たとえば、符号化されたピクチャデータの記憶装置のような記憶装置から、符号化されたピクチャデータ21(またはその任意のさらに処理されたバージョン)を受信し、符号化されたピクチャデータ21をデコーダ30に提供するように構成される可能性がある。

40

【0067】

通信インターフェース22および通信インターフェース28は、送信元デバイス12と送信先デバイス14との間の直接通信リンク、たとえば、直接有線もしくはワイヤレス接続を介して、あるいは任意の種類のパクチャネットワーク、たとえば、有線もしくはワイヤレスネットワークもしくはそれらの任意の組合せ、または任意の種類のパクチャネットワーク、またはそれらの任意の種類のパクチャの組合せを介して符号化されたピクチャデータ21または符号化されたデータ13を送信または受信するように構成される可能性がある。

50

## 【 0 0 6 8 】

通信インターフェース22は、符号化されたピクチャデータ21を適切なフォーマット、たとえば、パケットにパッケージングする、および/または通信リンクもしくは通信ネットワークを介して送信するための任意の種類の変換の符号化もしくは処理を使用して符号化されたピクチャデータを処理するように構成される可能性がある。

## 【 0 0 6 9 】

通信インターフェース22の相手先を形成する通信インターフェース28は、送信されたデータを受信し、任意の種類の変換の復号もしくは処理および/またはパッケージングの解除を使用して送信データを処理して符号化されたピクチャデータ21を取得するように構成される可能性がある。

10

## 【 0 0 7 0 】

通信インターフェース22と通信インターフェース28との両方が、送信元デバイス12から送信先デバイス14の方を指す図1Aの通信チャネル13に関する矢印によって示される単方向通信インターフェースとして、または双方向通信インターフェースとして構成される可能性があり、たとえば、接続をセットアップし、通信リンクおよび/または符号化されたピクチャデータの送信のようなデータ送信、に関連する任意の他の情報を確認し、やりとりするために、メッセージを送信および受信するように構成される可能性がある。

## 【 0 0 7 1 】

デコーダ30は、符号化されたピクチャデータ21を受信し、復号されたピクチャデータ31または復号されたピクチャ31を提供するように構成される可能性がある(さらなる詳細は、たとえば、図3または図5に基づいて以下で説明される)。

20

## 【 0 0 7 2 】

送信先デバイス14のポストプロセッサ32は、後処理されたピクチャ33などの後処理されたピクチャデータ33を取得するために、復号されたピクチャデータ31(再構築されたピクチャデータとも称される)、たとえば、復号されたピクチャ31を後処理するように構成される可能性がある。後処理ユニット32によって実行される後処理は、(たとえば、YCbCrからRGBへの)カラーフォーマット変換、色補正、トリミング、またはリサンプリング、またはたとえばディスプレイデバイス34による表示のためにたとえば復号されたピクチャデータ31を準備するための任意の他の処理の任意の1つまたは複数を含む可能性がある。

## 【 0 0 7 3 】

送信先デバイス14のディスプレイデバイス34は、たとえば、ユーザまたは視聴者に対してピクチャを表示するために後処理されたピクチャデータ33を受け取るように構成される可能性がある。ディスプレイデバイス34は、一体型または外部ディスプレイもしくはモニタのような再構築されたピクチャを示すための任意の種類の変換のディスプレイであるかまたはそのようなディスプレイもしくはモニタを含む可能性がある。ディスプレイは、液晶ディスプレイ(LCD)、有機発光ダイオード(OLED)ディスプレイ、プラズマディスプレイ、プロジェクタ、マイクロLEDディスプレイ、液晶オンシリコン(LCoS: liquid crystal on silicon)、デジタル光プロセッサ(DLP: digital light processor)、または任意の種類の変換のディスプレイを含む可能性がある。

30

## 【 0 0 7 4 】

図1Aは送信元デバイス12および送信先デバイス14を別々のデバイスとして示すが、デバイスの実施形態は、両方のデバイスまたは両方の機能、つまり、送信元デバイス12または対応する機能および送信先デバイス14または対応する機能を含む可能性もある。そのような実施形態において、送信元デバイス12または対応する機能および送信先デバイス14または対応する機能は、同じハードウェアおよび/もしくはソフトウェアを使用してまたは別々のハードウェアおよび/もしくはソフトウェアまたはそれらの任意の組合せによって実装される可能性がある。

40

## 【 0 0 7 5 】

説明に基づいて当業者に明らかになるように、異なるユニットの機能または図1Aに示される送信元デバイス12および/もしくは送信先デバイス14内の機能の存在および(厳密な)

50

分割は、実際のデバイスおよびアプリケーションに応じて変わる可能性がある。

【0076】

エンコーダ20(たとえば、ビデオエンコーダ20)またはデコーダ30(たとえば、ビデオデコーダ30)またはエンコーダ20とデコーダ30との両方は、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、それらのビデオ符号化に専用のまたは任意の組合せなどの、図1Bに示された処理回路によって実装される可能性がある。エンコーダ20は、図2のエンコーダ20および/または本明細書において説明される任意の他のエンコーダシステムもしくはサブシステムに関連して検討される様々なモジュールを具現化するために処理回路46によって実装される可能性がある。デコーダ30は、図3のデコーダ30および/または本明細書において説明される任意の他のデコーダシステムもしくはサブシステムに関連して検討される様々なモジュールを具現化するために処理回路46によって実装される可能性がある。処理回路は、後で検討される様々な動作を実行するように構成される可能性がある。図5に示されるように、技術が部分的にソフトウェアで実装される場合、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ための命令を記憶する可能性があり、本開示の技術を実行するために1つまたは複数のプロセッサを使用するハードウェアにおいて命令を実行する可能性がある。ビデオエンコーダ20およびビデオデコーダ30は、たとえば、図1Bに示されるように単一のデバイス内の組み合わせられたエンコーダ/デコーダ(コーデック)の一部として組み込まれる可能性がある。

【0077】

図1Bに示されたビデオコーディングシステム40は、ビデオエンコーダ20とビデオデコーダ30との両方を実装する処理回路を含む。加えて、実世界のピクチャを撮影するためのカメラなどの1つもしくは複数のイメージングデバイス41、アンテナ42、1つもしくは複数のメモリストア44、1つもしくは複数のプロセッサ43、および/または上で説明されたディスプレイデバイス34のようなディスプレイデバイス45が、ビデオコーディングシステム40の一部として提供される可能性がある。

【0078】

送信元デバイス12および送信先デバイス14は、任意の種類の手ヘルドまたは固定デバイス、たとえば、ノートブックまたはラップトップコンピュータ、モバイル電話、スマートフォン、タブレットまたはタブレットコンピュータ、カメラ、デスクトップコンピュータ、セットトップボックス、テレビ、ディスプレイデバイス、デジタルメディアプレーヤー、ビデオゲームコンソール、(コンテンツサービスサーバまたはコンテンツ配信サーバなどの)ビデオストリーミングデバイス、放送受信機デバイス、放送送信機デバイスなどを含む広範なデバイスのいずれかを含む可能性があり、オペレーティングシステムを使用しないかまたは任意の種類オペレーティングシステムを使用する可能性がある。場合によっては、送信元デバイス12および送信先デバイス14は、ワイヤレス通信に対応している可能性がある。したがって、送信元デバイス12および送信先デバイス14は、ワイヤレス通信デバイスである可能性がある。

【0079】

場合によっては、図1Aに示されたビデオコーディングシステム10は、例であるに過ぎず、本開示の技術は、符号化デバイスと復号デバイスとの間のいかなるデータ通信も含むとは限らないビデオコーディングシステム(たとえば、ビデオの符号化またはビデオの復号)に適用される可能性がある。他の例においては、データが、ローカルメモリから取り出される、またはネットワークを介してストリーミングされる、などである。ビデオ符号化デバイスが、データを符号化し、メモリに記憶する可能性があり、および/またはビデオ復号デバイスが、メモリからデータを取り出し、復号する可能性がある。いくつかの例において、符号化および復号が、互いに通信せず、単に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、復号するデバイスによって実行される。

【0080】

説明の便宜上、本開示の実施形態は、たとえば、高効率ビデオ符号化(HEVC: High-Efficiency Video Coding)、または多目的ビデオ符号化(VVC: Versatile Video coding)、ITU-Tビデオ符号化専門家グループ(VCEG: Video Coding Experts Group)およびISO/IEC動画専門家グループ(MPEG: Motion Picture Experts Group)のビデオ符号化に関する共同作業チーム(JCT-VC: Joint Collaboration Team on Video Coding)によって開発された次世代ビデオコーディング規格の参照ソフトウェアを参照することによって本明細書において説明される。当業者は、本開示の実施形態がHEVCまたはVVCに限定されないことを理解するであろう。

#### 【0081】

エンコーダおよび符号化方法

図2は、本出願の技術を実装するように構成される例示的なビデオエンコーダ20の概略的なブロック図を示す。図2の例において、ビデオエンコーダ20は、入力201(または入力インターフェース201)、残差計算ユニット204、変換処理ユニット206、量子化ユニット208、逆量子化ユニット210、逆変換処理ユニット212、再構築ユニット214、ループフィルタユニット220、復号ピクチャバッファ(DPB: decoded picture buffer)230、モード選択ユニット260、エントロピー符号化ユニット270、および出力272(または出力インターフェース272)を含む。モード選択ユニット260は、インター予測ユニット244、イントラ予測ユニット254、および区分けユニット262を含む可能性がある。インター予測ユニット244は、動き推定ユニットおよび動き補償ユニット(図示せず)を含む可能性がある。図2に示されたビデオエンコーダ20は、ハイブリッドビデオエンコーダまたはハイブリッドビデオコーデックによるビデオエンコーダとも称される可能性がある。

#### 【0082】

残差計算ユニット204、変換処理ユニット206、量子化ユニット208、およびモード選択ユニット260は、エンコーダ20の順方向信号経路を形成すると言われる可能性があり、一方、逆量子化ユニット210、逆変換処理ユニット212、再構築ユニット214、ループフィルタ220、復号ピクチャバッファ(DPB)230、インター予測ユニット244、およびイントラ予測ユニット254は、ビデオエンコーダ20の逆方向信号経路を形成すると言われる可能性があり、ビデオエンコーダ20の逆方向信号経路は、デコーダの信号経路(図3のビデオデコーダ30を参照されたい)に対応する。逆量子化ユニット210、逆変換処理ユニット212、再構築ユニット214、ループフィルタ220、復号ピクチャバッファ(DPB)230、インター予測ユニット244、およびイントラ予測ユニット254は、ビデオエンコーダ20の「内蔵デコーダ」を形成するとも言われる。

#### 【0083】

ピクチャ&ピクチャの区分け(ピクチャ&ブロック)

エンコーダ20は、たとえば、入力201を介してピクチャ17(またはピクチャデータ17)、たとえば、ビデオまたはビデオシーケンスを形成するピクチャのシーケンスのピクチャを受け取るように構成される可能性がある。受け取られたピクチャまたはピクチャデータは、前処理されたピクチャ19(または前処理されたピクチャデータ19)である可能性もある。簡単にするために、以下の説明は、ピクチャ17に言及する。ピクチャ17は、(特に、ビデオコーディングにおいて、現在のピクチャを他のピクチャ、たとえば、同じビデオシーケンス、つまり、現在のピクチャも含むビデオシーケンスの既に符号化されたおよび/または復号されたピクチャと区別するために)現在のピクチャまたはコーディングされるピクチャとも称される可能性がある。

#### 【0084】

(デジタル)ピクチャは、強度値を有するサンプルの2次元配列または行列とみなされるかまたはみなされ得る。配列のサンプルは、ピクセル(ピクチャエレメントの短縮形)またはペルとも称される可能性がある。配列またはピクチャの水平および垂直方向(または軸)のサンプル数は、ピクチャのサイズおよび/または解像度を定義する。色の表現のために、概して、3つの色成分が使用され、つまり、ピクチャが3つのサンプル配列として表現されるかまたは3つのサンプル配列を含む可能性がある。RGBフォーマットまたは色空間で、ピ

10

20

30

40

50

クチャは、対応する赤、緑、および青のサンプル配列を含む。しかし、ビデオコーディングにおいて、各ピクセルは、概して、輝度(luminance)およびクロミナンス(chrominance)フォーマットまたは色空間、たとえば、Y(代わりにLが使用されることもある)によって示される輝度成分ならびにCbおよびCrによって示される2つのクロミナンス成分を含むYCbCrで表される。輝度(または短くルマ(luma))成分Yは、明るさまたは(たとえば、グレースケールピクチャと同様の)グレーレベルの強度を表し、一方、2つのクロミナンス(または短くクロマ(chroma))成分CbおよびCrは、色度または色情報成分を表す。したがって、YCbCrフォーマットのピクチャは、輝度サンプル値(Y)の輝度サンプル配列およびクロミナンス値(CbおよびCr)の2つのクロミナンスサンプル配列を含む。RGBフォーマットのピクチャは、YCbCrフォーマットに変換される(converted)または変換される(transformed)可能性があり、その逆の可能性もある。プロセスは、色変換(transformation)または変換(conversion)としても知られる。ピクチャがモノクロである場合、ピクチャは、輝度サンプル配列のみを含む可能性がある。したがって、ピクチャは、たとえば、モノクロフォーマットにおいてはルマサンプルの配列であり、または4:2:0、4:2:2、および4:4:4カラーフォーマットにおいてはルマサンプルの配列およびクロマサンプルの2つの対応する配列である可能性がある。

10

#### 【0085】

ビデオエンコーダ20の実施形態は、ピクチャ17を複数の(通常は重なり合わない)ピクチャブロック203に区分けするように構成されたピクチャ区分けユニット(図2に示さず)を含む可能性がある。これらのブロックは、ルートブロック、マクロブロック(H.264/AVC)、または符号化ツリーブロック(CTB: coding tree block)もしくは符号化ツリーユニット(CTU: coding tree unit)(H.265/HEVCおよびVVCによる)とも称される可能性がある。ピクチャ区分けユニットは、ビデオシーケンスのすべてのピクチャおよびブロックサイズを定義する対応するグリッドに関して同じブロックサイズを使用するか、あるいはピクチャまたはピクチャのサブセットもしくはグループの間でブロックサイズを変更し、各ピクチャを対応するブロックに区分けするように構成される可能性がある。

20

#### 【0086】

さらなる実施形態において、ビデオエンコーダは、ピクチャ17のブロック203、たとえば、ピクチャ17を形成する1つの、いくつかの、またはすべてのブロックを直接受け取るように構成される可能性がある。ピクチャブロック203は、現在のピクチャブロックまたはコーディングされるピクチャブロックとも称される可能性がある。

30

#### 【0087】

ピクチャ17と同様に、ピクチャブロック203は、ピクチャ17よりも寸法が小さいが、強度値(サンプル値)を有するサンプルの2次元配列または行列とみなされるかまたはみなされ得る。言い換えると、ブロック203は、適用されるカラーフォーマットに応じて、たとえば、1つのサンプル配列(たとえば、モノクロピクチャ17の場合はルマ配列、またはカラーピクチャの場合はルマもしくはクロマ配列)、あるいは3つのサンプル配列(たとえば、カラーピクチャ17の場合はルマおよび2つのクロマ配列)、あるいは任意の他の数および/または種類の配列を含む可能性がある。ブロック203の水平および垂直方向(または軸)のサンプル数は、ブロック203のサイズを定義する。したがって、ブロックは、たとえば、サンプルのM×N(M列×N行)配列または変換係数のM×N配列を含む可能性がある。

40

#### 【0088】

図2に示されたビデオエンコーダ20の実施形態は、ピクチャ17をブロック毎に符号化するように構成される可能性があり、たとえば、符号化および予測が、ブロック203毎に実行される。

#### 【0089】

図2に示されるビデオエンコーダ20の実施形態は、スライス(ビデオスライスとも称される)を使用することによってピクチャを区分けするおよび/または符号化するようにさらに構成される可能性があり、ピクチャは、1つもしくは複数の(概して重なり合わない)スライスに区分けされるかまたは1つもしくは複数の(概して重なり合わない)スライスを使用して

50

符号化される可能性があり、各スライスは、1つまたは複数のブロック(たとえば、CTU)を含む可能性がある。スライスは、長方形のタイルの集合を含む可能性があり、またはタイル内のラスタ順のCTU行の集合であり得る。

#### 【0090】

図2に示されるビデオエンコーダ20の実施形態は、タイルグループ(ビデオタイルグループとも称される)および/またはタイル(ビデオタイルとも称される)を使用することによってピクチャを区別するおよび/または符号化するようにさらに構成される可能性があり、ピクチャは、1つもしくは複数の(概して重なり合わない)タイルグループに区別されるかまたは1つもしくは複数の(概して重なり合わない)タイルグループを使用して符号化される可能性があり、各タイルグループは、1つもしくは複数のブロック(たとえば、CTU)または1つもしくは複数のタイルを含む可能性があり、各タイルは、長方形の形をしている可能性があり、1つまたは複数のブロック(たとえば、CTU)、たとえば、完全なまたは断片的な(fractional)ブロックを含む可能性がある。

10

#### 【0091】

##### 残差の計算

残差計算ユニット204は、たとえば、サンプル毎に(ピクセル毎に)ピクチャブロック203のサンプル値から予測ブロック265のサンプル値を差し引いてサンプル領域において残差ブロック205を取得することによって、ピクチャブロック203および予測ブロック265(予測ブロック265についてのさらなる詳細は後で与えられる)に基づいて残差ブロック205(残差205とも称される)を計算するように構成される可能性がある。

20

#### 【0092】

##### 変換

変換処理ユニット206は、残差ブロック205のサンプル値に対して、離散コサイン変換(DCT)または離散サイン変換(DST)のような変換を適用して変換領域において変換係数207を取得するように構成される可能性がある。変換係数207は、変換残差係数とも称され、変換領域において残差ブロック205を表現する可能性がある。

#### 【0093】

変換処理ユニット206は、H.265/HEVCのために規定された変換などのDCT/DSTの整数近似を適用するように構成される可能性がある。直交DCT変換と比較して、そのような整数近似は、概して、特定の率でスケールされる。順および逆変換によって処理される残差ブロックのノルム(norm)を維持するために、追加的な倍率(scaling factor)が、変換プロセスの一部として適用される。倍率は、概して、倍率がシフト演算のために2の累乗であること、変換係数のビット深度、正確さと実装コストとの間のトレードオフなどのような特定の制約に基づいて選択される。たとえば、特定の倍率が、たとえば、逆変換処理ユニット212による逆変換(およびたとえば、ビデオデコーダ30における逆変換処理ユニット312による対応する逆変換)のために指定され、たとえば、エンコーダ20の変換処理ユニット206による順変換のための対応する倍率が、それに応じて指定される可能性がある。

30

#### 【0094】

ビデオエンコーダ20(それぞれ、変換処理ユニット206)の実施形態は、たとえば、ビデオデコーダ30が変換パラメータを受信し、復号のために使用する可能性があるように、たとえば、そのままであるかまたはエントロピー符号化ユニット270によって符号化されるかもしくは圧縮される変換パラメータ、たとえば、ある種の1つの変換または複数の変換を出力するように構成される可能性がある。

40

#### 【0095】

##### 量子化

量子化ユニット208は、たとえば、スカラー量子化またはベクトル量子化を適用することによって変換係数207を量子化して量子化された係数209を取得するように構成される可能性がある。量子化された係数209は、量子化された変換係数209または量子化された残差係数209とも称される可能性がある。

50

## 【 0 0 9 6 】

量子化プロセスは、変換係数207の一部またはすべてに関連するビット深度を削減する可能性がある。たとえば、 $n$ ビットの変換係数が、量子化中に $m$ ビットの変換係数に切り捨てられる可能性があり、 $n$ は、 $m$ よりも大きい。量子化の度合いは、量子化パラメータ(QP: quantization parameter)を調整することによって修正される可能性がある。たとえば、スカラー量子化に関して、より細かいまたはより粗い量子化を達成するために異なるスケールリングが適用される可能性がある。より小さな量子化ステップサイズは、より細かい量子化に対応し、一方、より大きな量子化ステップサイズは、より粗い量子化に対応する。適用可能な量子化ステップサイズが、量子化パラメータ(QP)によって示される可能性がある。量子化パラメータは、たとえば、適用可能な量子化ステップサイズの予め定義された組のインデックスである可能性がある。たとえば、小さな量子化パラメータが、細かい量子化(小さな量子化ステップサイズ)に対応する可能性があり、大きな量子化パラメータが、粗い量子化(大きな量子化ステップサイズ)に対応する可能性があり、またはその逆である可能性がある。量子化は、量子化ステップサイズによる除算を含む可能性があり、たとえば、逆量子化ユニット210による対応するおよび/または逆量子化は、量子化ステップサイズによる乗算を含む可能性がある。一部の規格、たとえば、HEVCによる実施形態は、量子化パラメータを使用して量子化ステップサイズを決定するように構成される可能性がある。概して、量子化ステップサイズは、除算を含む等式の固定小数点近似(fixed point approximation)を使用して量子化パラメータに基づいて計算される可能性がある。量子化ステップサイズおよび量子化パラメータに関する等式の固定小数点近似において使用されるスケールリングが原因で修正される可能性がある残差ブロックのノルムを復元するために、量子化および逆量子化に関して追加的な倍率が導入される可能性がある。1つの例示的な実装においては、逆変換および逆量子化のスケールリングが組み合わせられる可能性がある。あるいは、カスタマイズされた量子化テーブルが使用され、たとえば、ビットストリーム内でエンコーダからデコーダにシグナリングされる可能性がある。量子化は、不可逆演算で

10

20

あり、損失は、量子化ステップサイズが大きくなるにつれて増加する。

## 【 0 0 9 7 】

ビデオエンコーダ20(それぞれ、量子化ユニット208)の実施形態は、たとえば、ビデオデコーダ30が量子化パラメータを受信し、復号のために適用する可能性があるように、たとえば、そのままであるかまたはエントロピー符号化ユニット270によって符号化される量子化パラメータ(QP)を出力するように構成される可能性がある。

30

## 【 0 0 9 8 】

## 逆量子化

逆量子化ユニット210は、たとえば、量子化ユニット208と同じ量子化ステップサイズに基づいてまたはそれを使用して、量子化ユニット208により適用された量子化方式の逆を適用することによって、量子化された係数に量子化ユニット208の逆量子化を適用して量子化解除された係数211を取得するように構成される。量子化解除された係数211は、量子化解除された残差係数211とも称され、量子化による損失が原因で概して変換係数と同一ではないが、変換係数207に対応する可能性がある。

40

## 【 0 0 9 9 】

## 逆変換

逆変換処理ユニット212は、変換処理ユニット206によって適用された変換の逆変換、たとえば、逆離散コサイン変換(DCT)または逆離散サイン変換(DST)または他の逆変換を適用してサンプル領域において再構築された残差ブロック213(または対応する量子化解除された係数213)を取得するように構成される。再構築された残差ブロック213は、変換ブロック(transform block)213とも称される可能性がある。

## 【 0 1 0 0 】

## 再構築

再構築ユニット214(たとえば、加算器または合算器214)は、たとえば、再構築された

50

残差ブロック213のサンプル値と予測ブロック265のサンプル値とを、サンプル毎に、足すことによって予測ブロック265に変換ブロック213(すなわち、再構築された残差ブロック213)を足してサンプル領域において再構築されたブロック215を取得するように構成される。

#### 【0101】

##### フィルタリング

ループフィルタユニット220(または短く「ループフィルタ」220)は、再構築されたブロック215をフィルタリングしてフィルタリングされたブロック221を取得する、または概して、再構築されたサンプルをフィルタリングしてフィルタリングされたサンプルを取得するように構成される。ループフィルタユニットは、ピクセルの遷移を平滑化するかまたはそれ以外の方法でビデオの品質を改善するように構成される可能性がある。ループフィルタユニット220は、デブロッキングフィルタ、サンプル適応オフセット(SAO: sample-adaptive offset)フィルタ、または、バイラテラルフィルタ、適応ループフィルタ(ALF: adaptive loop filter)、鮮鋭化、平滑化フィルタ、もしくは共同フィルタ(collaborative filter)、もしくはこれらの任意の組合せのような1つもしくは複数の他のフィルタなどの1つまたは複数のループフィルタを含む可能性がある。ループフィルタユニット220は図2にループ内フィルタであるものとして示されるが、他の構成において、ループフィルタユニット220は、ループ後フィルタとして実装される可能性がある。フィルタリングされたブロック221は、フィルタリングされた再構築されたブロック221とも称される可能性がある。

#### 【0102】

ビデオエンコーダ20(それぞれ、ループフィルタユニット220)の実施形態は、たとえば、デコーダ30が同じループフィルタのパラメータまたはそれぞれのループフィルタを受信し、復号のために適用する可能性があるように、たとえば、そのままであるかまたはエントロピー符号化ユニット270によって符号化される(サンプル適応オフセット情報などの)ループフィルタのパラメータを出力するように構成される可能性がある。

#### 【0103】

##### 復号ピクチャバッファ

復号ピクチャバッファ(DPB)230は、ビデオエンコーダ20によってビデオデータを符号化するための参照ピクチャまたは概して参照ピクチャデータを記憶するメモリである可能性がある。DPB230は、同期DRAM(SDRAM)を含むダイナミックランダムアクセスメモリ(DRAM)、磁気抵抗RAM(MRAM)、抵抗変化型RAM(RRAM: resistive RAM)、または他の種類のメモリデバイスなどの様々なメモリデバイスのいずれかによって形成される可能性がある。復号ピクチャバッファ(DPB)230は、1つまたは複数のフィルタリングされたブロック221を記憶するように構成される可能性がある。復号ピクチャバッファ230は、同じ現在のピクチャまたは異なるピクチャ、たとえば、既に再構築されたピクチャの他の既にフィルタリングされたブロック、たとえば、既に再構築され、フィルタリングされたブロック221を記憶するようにさらに構成される可能性があり、たとえば、インター予測のために、完全な既に再構築された、つまり、復号されたピクチャ(および対応する参照ブロックおよびサンプル)ならびに/または部分的に再構築された現在のピクチャ(および対応する参照ブロックおよびサンプル)を提供する可能性がある。復号ピクチャバッファ(DPB)230は、たとえば、再構築されたブロック215がループフィルタユニット220によってフィルタリングされない場合、1つもしくは複数のフィルタリングされていない再構築されたブロック215もしくは概してフィルタリングされていない再構築されたサンプルを記憶し、または再構築されたブロックもしくはサンプルの任意の他のさらに処理されたバージョンを記憶するようにも構成される可能性がある。

#### 【0104】

##### モード選択(区分け&予測)

モード選択ユニット260は、区分けユニット262、インター予測ユニット244、およびイントラ予測ユニット254を含み、元のブロック203(現在のピクチャ17の現在のブロッ

ク203)のような元のピクチャデータと、同じ(現在の)ピクチャの、および/またはたとえば復号ピクチャバッファ230もしくは他のバッファ(たとえば、図示されていないラインバッファ)からの1つもしくは複数の既に復号されたピクチャからのフィルタリングされたおよび/またはフィルタリングされていない再構築されたサンプルまたはブロックのような再構築されたピクチャデータとを受け取るかまたは取得するように構成される。再構築されたピクチャデータは、予測ブロック265または予測子(predictor)265を取得するための予測、たとえば、インター予測またはイントラ予測のための参照ピクチャデータとして使用される。

#### 【0105】

モード選択ユニット260は、(区分けを含まない)現在のブロックの予測モードのための区分けおよび予測モード(たとえば、イントラまたはインター予測モード)を決定するかまたは選択し、残差ブロック205の計算および再構築されたブロック215の再構築のために使用される対応する予測ブロック265を生成するように構成される可能性がある。

10

#### 【0106】

モード選択ユニット260の実施形態は、最良の一致もしくは言い換えると最小の残差(最小の残差は送信もしくは記憶のためのより優れた圧縮を意味する)または最小のシグナリングオーバーヘッド(最小のシグナリングオーバーヘッドは送信もしくは記憶のためのより優れた圧縮を意味する)を提供する、あるいはそれら両方を考慮するかまたは釣り合いを取る区分けおよび予測モードを(たとえば、モード選択ユニット260によってサポートされるかまたはモード選択ユニット260が利用可能な区分けおよび予測モードから)選択するように構成される可能性がある。モード選択ユニット260は、レート歪み最適化(RDO)に基づいて区分けおよび予測モードを決定する、つまり、最小のレート歪みを提供する予測モードを選択するように構成される可能性がある。この文脈の「最良の」、「最小の」、「最適な」などのような用語は、必ずしも全体の「最良の」、「最小の」、「最適な」などを指さず、値が閾値を超えることもしくは下回ることのような終了もしくは選択の基準、または潜在的に「準最適な選択」につながるが、複雑さおよび処理時間を削減する他の制約を満たすことをも指す可能性もある。

20

#### 【0107】

言い換えると、区分けユニット262は、たとえば、四分木区分け(QT)、二分木区分け(BT)、または三分木区分け(TT)、またはこれらの任意の組合せを反復的に使用してブロック203を(やはりブロックを形成する)より小さなブロックの区画または下位ブロックに区分けし、ブロックの区画または下位ブロックの各々に関して予測を実行するように構成される可能性があり、モード選択は、区分けされたブロック203の木構造の選択を含み、予測モードは、ブロックの区画または下位ブロックの各々に適用される。

30

#### 【0108】

以下で、例示的なビデオエンコーダ20によって実行される(たとえば、区分けユニット262による)区分けならびに(インター予測ユニット244およびイントラ予測ユニット254による)予測処理が、より詳細に説明される。

#### 【0109】

##### 区分け

40

区分けユニット262は、現在のブロック203をより小さな区画、たとえば、正方形または長方形のサイズのより小さなブロックに区分けする(または分割する)可能性がある。これらのより小さなブロック(下位ブロックとも称される可能性がある)は、より一層小さな区画にさらに区分けされる可能性がある。これは、木区分けまたは階層的木区分けとも称され、たとえば、ルートツリーレベル0(階層レベル0、深さ0)のルートブロックが、再帰的に区分けされ、たとえば、次に低いツリーレベルの2つ以上のブロック、たとえば、ツリーレベル1(階層レベル1、深さ1)のノードに区分けされる可能性があり、これらのブロックが、次に低いレベル、たとえば、ツリーレベル2(階層レベル2、深さ2)の2つ以上のブロックに再び区分けされる可能性があり、たとえば、終了基準が満たされる、たとえば、最大のツリーの深さまたは最小のブロックサイズが達せられるので区分けが終了されるま

50

で以下同様である。さらに区分けされないブロックは、木の葉ブロックまたは葉ノードとも称される。2つの区画への区分けを使用する木は、二分木(BT)と称され、3つの区画への区分けを使用する木は、三分木(TT)と称され、4つの区画への区分けを使用する木は、四分木(QT)と称される。

#### 【0110】

上述のように、本明細書において使用される用語「ブロック」は、ピクチャの一部分、特に、正方形または長方形の一部分である可能性がある。たとえば、HEVCおよびVVCに関連して、ブロックは、符号化ツリーユニット(CTU)、符号化ユニット(CU: coding unit)、予測ユニット(PU: prediction unit)、または変換ユニット(TU: transform unit)、ならびに/または対応するブロック、たとえば、符号化ツリーブロック(CTB)、符号化ブロック(CB: coding block)、変換ブロック(TB)、または予測ブロック(PB)であるかまたはそれらに対応する可能性がある。

10

#### 【0111】

たとえば、符号化ツリーユニット(CTU)は、ルマサンプルのCTBおよび3つのサンプル配列を有するピクチャのクロマサンプルの2つの対応するCTB、またはモノクロピクチャもしくはサンプルをコーディングするために使用される3つの別々の色平面(colour plane)およびシンタックス(syntax)構造を使用してコーディングされるピクチャのサンプルのCTBであるかまたはそれらを含む可能性がある。それに対応して、符号化ツリーブロック(CTB)は、構成要素のCTBへの分割が区分けであるようなNの何らかの値に関するサンプルのN×Nのブロックである可能性がある。符号化ユニット(CU)は、ルマサンプルの符号化ブロックおよび3つのサンプル配列を有するピクチャのクロマサンプルの2つの対応する符号化ブロック、またはモノクロピクチャもしくはサンプルをコーディングするために使用される3つの別々の色平面およびシンタックス構造を使用してコーディングされるピクチャのサンプルの符号化ブロックであるかまたはそれらを含む可能性がある。それに対応して、符号化ブロック(CB)は、CTBの符号化ブロックへの分割が区分けであるようなMおよびNの何らかの値に関するサンプルのM×Nのブロックである可能性がある。

20

#### 【0112】

たとえば、HEVCによる一部の実施形態において、符号化ツリーユニット(CTU)は、符号化ツリーとして表される四分木構造を使用することによってCUに分割される可能性がある。インターピクチャ(時間)予測を使用してピクチャエリアをコーディングすべきかまたはイントラピクチャ(空間)予測を使用してピクチャエリアをコーディングすべきかの判断は、CUレベルで行われる。各CUは、PU分割タイプに従って1つ、2つ、または4つのPUにさらに分割され得る。1つのPU内では、同じ予測プロセスが適用され、関連する情報がPUに基づいてデコーダに送信される。PU分割タイプに基づいて予測プロセスを適用することによって残差ブロックを取得した後、CUは、CUに関する符号化ツリーと同様の別の四分木構造によって変換ユニット(TU)に区分けされ得る。

30

#### 【0113】

たとえば、多目的ビデオ符号化(VVC)と称される現在開発されている最新のビデオコーディング規格による実施形態においては、組み合わせられた四分木および二分木(QTBT)区分けが、たとえば、符号化ブロックを区分けするために使用される。QTBTブロック構造において、CUは、正方形かまたは長方形かのどちらかの形状を持ち得る。たとえば、符号化ツリーユニット(CTU)が、まず、四分木構造によって区分けされる。四分木の葉ノードが、二分木または三分(ternary)(または三分(triple))木構造によってさらに区分けされる。区分けツリーの葉ノードは、符号化ユニット(CU)と称され、その区分けが、いかなるさらなる区分けもなしに予測および変換処理のために使用される。これは、CU、PU、およびTUがQTBT符号化ブロック構造において同じブロックサイズを有することを意味する。平行して、多区画、たとえば、三分木区画は、QTBTブロック構造と一緒に使用され得る。

40

#### 【0114】

一例において、ビデオエンコーダ20のモード選択ユニット260は、本明細書において説明される区分け技術の任意の組合せを実行するように構成される可能性がある。

50

## 【0115】

上述のように、ビデオエンコーダ20は、1組の(たとえば、予め決定された)予測モードから最良のまたは最適な予測モードを決定するまたは選択するように構成される。1組の予測モードは、イントラ予測モードおよび/またはインター予測モードを含む可能性がある。

## 【0116】

## イントラ予測

1組のイントラ予測モードは、たとえばHEVCにおいて定義された、DC(もしくは平均)モードおよび平面モードのような非方向性モード、または方向性モードなどの35個の異なるイントラ予測モードを含む可能性があり、あるいはたとえばVVCのために定義された、DC(もしくは平均)モードおよび平面モードのような非方向性モード、または方向性モードなどの67個の異なるイントラ予測モードを含む可能性がある。

10

## 【0117】

イントラ予測ユニット254は、1組のイントラ予測モードからのイントラ予測モードによって、同じ現在のピクチャの近傍のブロックの再構築されたサンプルを使用して(イントラ)予測ブロック265を生成するように構成される。

## 【0118】

イントラ予測ユニット254(または概してモード選択ユニット260)は、たとえば、ビデオエンコーダ30が予測パラメータを受信し、復号のために使用する可能性があるように、符号化されたピクチャデータ21に含めるためにシンタックス要素266の形態でエンコードされた符号化ユニット270にイントラ予測パラメータ(または概してブロックに関する選択されたイントラ予測モードを示す情報)を出力するようにさらに構成される可能性がある。

20

## 【0119】

## インター予測

1組の(または可能な)インター予測モードは、利用可能な参照ピクチャ(つまり、たとえば、DBP230に記憶された前の少なくとも部分的に復号されたピクチャ)ならびに他のインター予測パラメータ、たとえば、最もよく一致する参照ブロックを探索するために参照ピクチャ全体が使用されるのかもしくは参照ピクチャの一部のみ、たとえば、現在のブロックのエリアの周りの探索窓(search window)エリアのみが使用されるか、ならびに/またはたとえば、半/セミペル(half/semi-pel)および/もしくは4分の1ペル補間のようなピクセル補間が適用されるか否かに依存する。

30

## 【0120】

上述の予測モードに加えて、スキップモードおよび/またはダイレクトモードが、適用される可能性がある。

## 【0121】

インター予測ユニット244は、動き推定(ME)ユニットおよび動き補償(MC)ユニット(どちらも図2に示さず)を含む可能性がある。動き推定ユニットは、動き推定のために、ピクチャブロック203(現在のピクチャ17の現在のピクチャブロック203)および復号されたピクチャ231、または1つもしくは複数の既に復号されたピクチャ231の再構築されたブロックのような少なくとも1つのもしくは複数の既に再構築されたブロックを受信するかまたは取得するように構成される可能性がある。例として、ビデオシーケンスは、現在のピクチャおよび既に復号されたピクチャ231を含む可能性があり、または言い換えると、現在のピクチャおよび既に復号されたピクチャ231は、ビデオシーケンスを形成するピクチャのシーケンスの一部であるかもしくはそのようなピクチャのシーケンスを形成する可能性がある。

40

## 【0122】

エンコーダ20は、複数の既に復号されたピクチャのうちの同じまたは異なるピクチャの複数の参照ブロックから参照ブロックを選択し、参照ピクチャ(もしくは参照ピクチャインデックス)および/または参照ブロックの位置(x、y座標)と現在のブロックの位置との間のオフセット(空間オフセット)をインター予測パラメータとして動き推定ユニットに提供す

50

るように構成される可能性がある。このオフセットは、動きベクトル(MV)とも称される。特定の実装において、MVは、予測精度を向上させ、最終的に圧縮パフォーマンスを向上させるために、符号化側と復号側との両方においてさらに改善され得る。さらに、MVまたは改善されたMVによって取得された予測は、ピクセルレベルにおいて微細な動きが存在する場合に予測が補正され得る双予測のオプティカルフローの改善(BPOF: Bi-predictive optical flow refinement)を使用することによってさらに修正または調整され得る。復号側の動きベクトルの改善(DMVR: decoding side motion vector refinement)および(BPOF)の技法については、以下で詳述される。

#### 【0123】

動き補償ユニットは、インター予測パラメータを取得、たとえば、受信し、インター予測パラメータに基づいてまたはインター予測パラメータを使用してインター予測を実行して(インター)予測ブロック265を取得するように構成される可能性がある。動き補償ユニットによって実行される動き補償は、おそらくはサブピクセルの精度の補間を実行する動き推定によって決定された動き/ブロックベクトルに基づく予測ブロックのフェッチまたは生成を含む可能性がある。補間フィルタリングが、知られているピクセルサンプルから追加的なピクセルサンプルを生成する可能性があり、したがって潜在的に、ピクチャブロックをコーディングするために使用される可能性がある候補予測ブロックの数を増やす。現在のピクチャブロックのPUに関する動きベクトルを受信すると、動き補償ユニットは、参照ピクチャリストのうちの1つにおいて動きベクトルが指す予測ブロックを見つける可能性がある。

#### 【0124】

動き補償ユニットは、ビデオスライスのピクチャブロックを復号する際にビデオデコーダ30によって使用するためのブロックおよびビデオスライスに関連するシンタックス要素も生成する可能性がある。スライスおよびそれぞれのシンタックス要素に加えて、またはスライスおよびそれぞれのシンタックス要素の代替として、タイルグループおよび/またはタイルならびにそれぞれのシンタックス要素が、生成されるかまたは使用される可能性がある。

#### 【0125】

##### エントロピーコーディング

エントロピー符号化ユニット270は、たとえば、ビデオデコーダ30がパラメータを受信し、復号のために使用される可能性があるように、たとえば、符号化されたビットストリーム21の形態で出力272を介して出力され得る符号化されたピクチャデータ21を得るために、量子化された係数209、インター予測パラメータ、イントラ予測パラメータ、ループフィルタパラメータ、および/または他のシンタックス要素に対して、たとえば、エントロピー符号化アルゴリズムもしくは方式(たとえば、可変長符号化(VLC: variable length coding)方式、コンテキスト適応VLC方式(CAVLC: context adaptive VLC)、算術符号化方式、2値化、コンテキスト適応2値算術符号化(CABAC: context adaptive binary arithmetic coding)、シンタックスに基づくコンテキスト適応2値算術符号化(SBAC: syntax-based context-adaptive binary arithmetic coding)、確率間隔区分エントロピー(PIPE: probability interval partitioning entropy)符号化、もしくは別のエントロピー符号化方法もしくは技術)またはバイパス(bypass)(非圧縮)を適用するように構成される。符号化されたビットストリーム21は、ビデオデコーダ30に送信されるか、または後の送信またはビデオデコーダ30による取り出しのためにメモリに記憶される可能性がある。

#### 【0126】

ビデオエンコーダ20他の構造の変化形が、ビデオストリームを符号化するために使用され得る。たとえば、変換に基づかないエンコーダ20は、特定のブロックまたはフレームに関して変換処理ユニット206なしに残差信号を直接量子化し得る。別の実装において、エンコーダ20は、単一のユニットに組み合わせられた量子化ユニット208および逆量子化ユニット210を持ち得る。

#### 【0127】

10

20

30

40

50

デコーダおよび復号方法

図3は、本出願の技術を実装するように構成されるビデオデコーダ30の例を示す。ビデオデコーダ30は、復号されたピクチャ331を取得するために、たとえば、エンコーダ20によって符号化された符号化されたピクチャデータ21(たとえば、符号化されたビットストリーム21)を受信するように構成される。符号化されたピクチャデータまたはビットストリームは、符号化されたピクチャデータ、たとえば、符号化されたビデオスライス(および/またはタイルグループもしくはタイル)のピクチャブロックならびに関連するシンタックス要素を表すデータを復号するための情報を含む。

【0128】

図3の例において、デコーダ30は、エンтроピー復号ユニット304、逆量子化ユニット310、逆変換処理ユニット312、再構築ユニット314(たとえば、合算器314)、ループフィルタ320、復号ピクチャバッファ(DBP)330、モード適用ユニット360、インター予測ユニット344、およびイントラ予測ユニット354を含む。インター予測ユニット344は、動き補償ユニットであるかまたは動き補償ユニットを含む可能性がある。ビデオデコーダ30は、いくつかの例において、図2のビデオエンコーダ20に関連して説明された符号化パスと概して逆である復号パスを実行する可能性がある。

10

【0129】

エンコーダ20に関連して説明されたように、逆量子化ユニット210、逆変換処理ユニット212、再構築ユニット214、ループフィルタ220、復号ピクチャバッファ(DPB)230、インター予測ユニット244、およびイントラ予測ユニット254は、ビデオエンコーダ20の「内蔵デコーダ」を形成するともいわれる。したがって、逆量子化ユニット310は、逆量子化ユニット210と機能的に同一である可能性があり、逆変換処理ユニット312は、逆変換処理ユニット212と機能的に同一である可能性があり、再構築ユニット314は、再構築ユニット214と機能的に同一である可能性があり、ループフィルタ320は、ループフィルタ220と機能的に同一である可能性があり、復号ピクチャバッファ330は、復号ピクチャバッファ230と機能的に同一である可能性がある。したがって、ビデオエンコーダ20のそれぞれのユニットおよび機能に関して与えられた説明が、ビデオデコーダ30のそれぞれのユニットおよび機能に準用される。

20

【0130】

エンтроピー復号

エンтроピー復号ユニット304は、ビットストリーム21(または概して符号化されたピクチャデータ21)を解析し、たとえば、符号化されたピクチャデータ21にエンтроピー復号を実行して、たとえば、量子化された係数309ならびに/あるいはインター予測パラメータ(たとえば、参照ピクチャインデックスおよび動きベクトル)、イントラ予測パラメータ(たとえば、イントラ予測モードもしくはインデックス)、変換パラメータ、量子化パラメータ、ループフィルタパラメータ、および/または他のシンタックス要素のいずれかまたはすべてのような復号されたコーディングパラメータ366を取得するように構成される。エンтроピー復号ユニット304は、エンコーダ20のエンтроピー符号化ユニット270に関連して説明された符号化方式に対応する復号アルゴリズムまたは方式を適用するように構成される可能性がある。エンтроピー復号ユニット304は、インター予測パラメータ、イントラ予測パラメータ、および/または他のシンタックス要素をモード適用ユニット360に提供し、他のパラメータをデコーダ30の他のユニットに提供するようにさらに構成される可能性がある。ビデオデコーダ30は、ビデオスライスのレベルおよび/またはビデオブロックのレベルでシンタックス要素を受信する可能性がある。スライスおよびそれぞれのシンタックス要素に加えて、またはスライスおよびそれぞれのシンタックス要素の代替として、タイルグループおよび/またはタイルならびにそれぞれのシンタックス要素が、受信されるおよび/または使用される可能性がある。

30

40

【0131】

逆量子化

逆量子化ユニット310は、(たとえば、エンтроピー復号ユニット304によって、たとえ

50

ば、解析および/または復号することによって)符号化されたピクチャデータ21から量子化パラメータ(QP)(または概して逆量子化に関連する情報)および量子化された係数を受け取り、復号された量子化された係数309に対して量子化パラメータに基づいて逆量子化を適用して、変換係数311とも称される可能性がある量子化解除された係数311を取得するように構成される可能性がある。逆量子化プロセスは、量子化の度合いと、同様に、適用されるべき逆量子化の度合いとを決定するために、ビデオスライス(またはタイルまたはタイルグループ)内の各ビデオブロックに関してビデオエンコーダ20によって決定された量子化パラメータを使用することを含む可能性がある。

#### 【0132】

##### 逆変換

逆変換処理ユニット312は、変換係数311とも称される量子化解除された係数311を受け取り、サンプル領域において再構築された残差ブロック313を取得するために、量子化解除された係数311に変換を適用するように構成される可能性がある。再構築された残差ブロック313は、変換ブロック213とも称される可能性がある。変換は、逆変換、たとえば、逆DCT、逆DST、逆整数変換、または概念的に同様の逆変換プロセスである可能性がある。逆変換処理ユニット312は、量子化解除された係数311に適用される変換を決定するために、(たとえば、エントロピー復号ユニット304によって、たとえば、解析および/または復号することによって)符号化されたピクチャデータ21から変換パラメータまたは対応する情報を受け取るようにさらに構成される可能性がある。

#### 【0133】

##### 再構築

再構築ユニット314(たとえば、加算器または合算器314)は、たとえば、再構築された残差ブロック313のサンプル値と予測ブロック365のサンプル値とを足すことによって予測ブロック365に再構築された残差ブロック313を足してサンプル領域において再構築されたブロック315を取得するように構成される可能性がある。

#### 【0134】

##### フィルタリング

(コーディンググループ内かまたはコーディンググループの後かのどちらかの)ループフィルタユニット320は、たとえば、ピクセルの遷移を平滑化するかまたはそれ以外の方法でビデオの品質を改善するために再構築されたブロック315をフィルタリングしてフィルタリングされたブロック321を取得するように構成される。ループフィルタユニット320は、デブロッキングフィルタ、サンプル適応オフセット(SAO)フィルタ、または1つもしくは複数の他のフィルタ、たとえば、バイラテラルフィルタ、適応ループフィルタ(ALF)、鮮鋭化、平滑化フィルタ、もしくは共同フィルタ、もしくはこれらの任意の組合せなどの1つまたは複数のループフィルタを含む可能性がある。ループフィルタユニット320は図3にループ内フィルタであるものとして示されるが、他の構成において、ループフィルタユニット320は、ループ後フィルタとして実装される可能性がある。

#### 【0135】

##### 復号ピクチャバッファ

それから、ピクチャの復号されたビデオブロック321は、他のピクチャに関するその後の動き補償のための参照ピクチャとしておよび/または出力もしくはそれぞれ表示するために復号されたピクチャ331を記憶する復号ピクチャバッファ330に記憶される。

#### 【0136】

デコーダ30は、復号されたピクチャ331を、ユーザへの提示または視聴のために、たとえば、出力332を介して出力するように構成される。

#### 【0137】

##### 予測

インター予測ユニット344は、インター予測ユニット244と(特に動き補償ユニットと)同一である可能性があり、イントラ予測ユニット354は、イントラ予測ユニット254と機能的に同一である可能性があり、(たとえば、エントロピー復号ユニット304によって、た

10

20

30

40

50

例えば、解析および/または復号することによって)復号されたピクチャデータ21から受け取られた区分けおよび/または予測パラメータまたはそれぞれの情報に基づいて分割または区分けの判断および予測を実行する。モード適用ユニット360は、予測ブロック365を得るために、(フィルタリングされたまたはフィルタリングされていない)再構築されたピクチャ、ブロック、またはそれぞれのサンプルに基づいてブロック毎に予測(イントラまたはインター予測)を実行するように構成される可能性がある。

#### 【0138】

ビデオスライスまたはピクチャ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時、モード適用ユニット360のイントラ予測ユニット354は、シグナリングされたイントラ予測モードおよび現在のピクチャの既に復号されたブロックからのデータに基づいて現在のビデオスライスのピクチャブロックに関する予測ブロック365を生成するように構成される。ビデオスライスまたはピクチャがインター符号化された(つまり、BまたはP)スライスとしてコーディングされる時、モード適用ユニット360のインター予測ユニット344(たとえば、動き補償ユニット)は、エンтроピー復号ユニット304から受け取られたモーションベクトルおよび他のシンタックス要素に基づいて現在のビデオスライスのビデオブロックに関する予測ブロック365を生成するように構成される。インター予測に関して、予測ブロックは、参照ピクチャリストのうちの1つの中の参照ピクチャのうちの1つから生成される可能性がある。ビデオデコーダ30は、DPB330に記憶された参照ピクチャに基づいてデフォルトの構築技術を使用して参照ピクチャリスト、List 0およびList 1を構築する可能性がある。同じまたは同様の手法が、スライス(たとえば、ビデオスライス)に加えてまたはスライス(たとえば、ビデオスライス)の代替としてタイルグループ(たとえば、ビデオタイルグループ)および/またはタイル(たとえば、ビデオタイル)を使用する実施形態のためにまたはそのような実施形態によって適用される可能性があり、たとえば、ビデオは、I、P、またはBタイルグループおよび/またはタイルを使用してコーディングされる可能性がある。

#### 【0139】

モード適用ユニット360は、動きベクトルまたは関連する情報および他のシンタックス要素を解析することによって現在のビデオスライスのビデオ/ピクチャブロックに関する予測情報を決定し、予測情報を使用して、復号されている現在のビデオブロックに関する予測ブロックを生成するように構成される。たとえば、モード適用ユニット360は、受信されたシンタックス要素の一部を使用して、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た予測モード(たとえば、イントラまたはインター予測)、インター予測のスライスタイプ(たとえば、Bスライス、Pスライス、またはGPBスライス)、スライスのための参照ピクチャリストのうちの1つまたは複数に関する構築情報、スライスのそれぞれのインター符号化されたビデオブロックに関する動きベクトル、スライスのそれぞれのインター符号化されたビデオブロックに関するインター予測のステータス、および現在のビデオスライス内のビデオブロックを復号するための他の情報を決定する。同じまたは同様の手法が、スライス(たとえば、ビデオスライス)に加えてまたはスライス(たとえば、ビデオスライス)の代替としてタイルグループ(たとえば、ビデオタイルグループ)および/またはタイル(たとえば、ビデオタイル)を使用する実施形態のためにまたはそのような実施形態によって適用される可能性があり、たとえば、ビデオは、I、P、またはBタイルグループおよび/またはタイルを使用してコーディングされる可能性がある。

#### 【0140】

図3に示されるビデオデコーダ30の実施形態は、スライス(ビデオスライスとも称される)を使用することによってピクチャを区分けするおよび/または復号するように構成される可能性があり、ピクチャは、1つもしくは複数の(概して重なり合わない)スライスに区分けされるかまたは1つもしくは複数の(概して重なり合わない)スライスを使用して復号される可能性があり、各スライスは、1つまたは複数のブロック(たとえば、CTU)を含む可能性がある。

#### 【0141】

10

20

30

40

50

図3に示されるビデオデコーダ30の実施形態は、タイルグループ(ビデオタイルグループとも称される)および/またはタイル(ビデオタイルとも称される)を使用することによってピクチャを区分けするおよび/または復号するように構成される可能性があり、ピクチャは、1つもしくは複数の(概して重なり合わない)タイルグループに区分けされるかまたは1つもしくは複数の(概して重なり合わない)タイルグループを使用して復号される可能性があり、各タイルグループは、1つもしくは複数のブロック(たとえば、CTU)または1つもしくは複数のタイルを含む可能性があり、各タイルは、長方形の形をしている可能性があり、1つまたは複数のブロック(たとえば、CTU)、たとえば、完全なまたは断片的なブロックを含む可能性がある。

#### 【0142】

ビデオデコーダ30の他の変化形が、符号化されたピクチャデータ21を復号するために使用され得る。たとえば、デコーダ30は、ループフィルタリングユニット320なしで出力ビデオストリームを生成し得る。たとえば、変換に基づかないデコーダ30は、特定のブロックまたはフレームに関して逆変換処理ユニット312なしに残差信号を直接逆量子化し得る。別の実装において、ビデオデコーダ30は、単一のユニットに組み合わされた逆量子化ユニット310および逆変換処理ユニット312を持ち得る。

#### 【0143】

エンコーダ20およびデコーダ30において、現在のステップの処理結果は、さらに処理され、それから次のステップに出力される可能性があることを理解されたい。たとえば、補間フィルタリング、動きベクトルの導出、またはループフィルタリングの後、Clipまたはシフトなどのさらなる演算が、補間フィルタリング、動きベクトルの導出、またはループフィルタリングの処理結果に対して実行される可能性がある。

#### 【0144】

さらなる演算が、(アフィンモードの制御点動きベクトル(control point motion vector)、アフィン、平面、ATMVPモードの下位ブロック動きベクトル、時間動きベクトル(temporal motion vector)などを含むがこれらに限定されない)現在のブロックの導出された動きベクトルに適用される可能性があることに留意されたい。たとえば、動きベクトルの値は、その表現ビット数に従って所定の範囲に制約される。動きベクトルの表現ビット数がbitDepthである場合、範囲は、 $-2^{(bitDepth-1)} \sim 2^{(bitDepth-1)}-1$ であり、「 $\wedge$ 」は、累乗を意味する。たとえば、bitDepthが16に等しいように設定される場合、範囲は、 $-32768 \sim 32767$ であり、bitDepthが18に等しいように設定される場合、範囲は、 $-131072 \sim 131071$ である。たとえば、導出された動きベクトル(たとえば、1つの $8 \times 8$ ブロック内の4つの $4 \times 4$ 下位ブロックのMV)の値は、4つの $4 \times 4$ 下位ブロックのMVの整数部分の間の最大の差が1ピクセル以下などNピクセル以下であるように制約される。以下の説明は、bitDepthに従って動きベクトルを制約するための2つの方法を提供する。

#### 【0145】

方法1: 以下の演算によってあふれ(overflow)MSB(最上位ビット)を削除する

$$ux = (mvx + 2^{bitDepth}) \% 2^{bitDepth} \quad (1)$$

$$mvx = (ux = 2^{bitDepth-1}) ? (ux - 2^{bitDepth}) : ux \quad (2)$$

$$uy = (mvy + 2^{bitDepth}) \% 2^{bitDepth} \quad (3)$$

$$mvy = (uy = 2^{bitDepth-1}) ? (uy - 2^{bitDepth}) : uy \quad (4)$$

式中、mvxは、画像ブロックまたは下位ブロックの動きベクトルの水平成分であり、mvyは、画像ブロックまたは下位ブロックの動きベクトルの垂直成分であり、uxおよびuyは、それぞれの中間値を示す。

#### 【0146】

たとえば、mvxの値が-32769である場合、式(1)および(2)を適用した後、結果として得られる値は、32767である。コンピュータシステムにおいて、10進数は、2の補数として記憶される。-32769の2の補数は、1,0111,1111,1111,1111(17ビット)である。そのとき、MSBが破棄され、したがって、結果として得られる2の補数は、0111,1111,1111,1111(10進数は32767)であり、これは、式(1)および(2)を適用することによる出力

10

20

30

40

50

と同じである。

$$ux = ( mvp_x + mvd_x + 2^{bitDepth} ) \% 2^{bitDepth} \quad (5)$$

$$mv_x = ( ux = 2^{bitDepth-1} ) ? ( ux - 2^{bitDepth} ) : ux \quad (6)$$

$$uy = ( mvp_y + mvd_y + 2^{bitDepth} ) \% 2^{bitDepth} \quad (7)$$

$$mv_y = ( uy = 2^{bitDepth-1} ) ? ( uy - 2^{bitDepth} ) : uy \quad (8)$$

演算は、式(5)から(8)に示されるように、動きベクトル予測子mvpと動きベクトルの差mvdとの合計中に適用される可能性がある。

【 0 1 4 7 】

方法2: 値をクリッピングすることによってあふれMSBを削除する

$$vx = \text{Clip3}(-2^{bitDepth-1}, 2^{bitDepth-1} - 1, vx)$$

$$vy = \text{Clip3}(-2^{bitDepth-1}, 2^{bitDepth-1} - 1, vy)$$

式中、vxは、画像ブロックまたは下位ブロックの動きベクトルの水平成分であり、vyは、画像ブロックまたは下位ブロックの動きベクトルの垂直成分であり、x、y、およびzは、MVのクリッピングプロセスの3つの入力値にそれぞれ対応し、関数Clip3の定義は、以下の通りである。

【 0 1 4 8 】

【数 1】

$$\text{Clip3}(x, y, z) = \begin{cases} x & ; & z < x \\ y & ; & z > y \\ z & ; & \text{それ以外の場合} \end{cases}$$

【 0 1 4 9 】

図4は、本開示の実施形態によるビデオコーディングデバイス400の概略図である。ビデオコーディングデバイス400は、下で説明されるように開示される実施形態を実装するのに好適である。実施形態において、ビデオコーディングデバイス400は、図1Aのビデオデコーダ30などのデコーダまたは図1Aのビデオエンコーダ20などのエンコーダである可能性がある。

【 0 1 5 0 】

ビデオコーディングデバイス400は、データを受信するための着信ポート410(または入力ポート410)および1つまたは複数の受信機ユニット(Rx)420、データを処理するためのプロセッサ、論理ユニット、または中央演算処理装置(CPU)430、データを送信するための1つまたは複数の送信機ユニット(Tx)440および発信ポート450(または出力ポート450)、ならびにデータを記憶するためのメモリ460を含む可能性がある。ビデオコーディングデバイス400は、光または電気信号の発信または着信のために着信ポート410、受信機ユニット420、送信機ユニット440、および発信ポート450に結合された光-電気(OE)構成要素および電気-光(EO)構成要素も含む可能性がある。

【 0 1 5 1 】

プロセッサ430は、ハードウェアおよびソフトウェアによって実装される可能性がある。プロセッサ430は、1つまたは複数のCPUチップ、コア(たとえば、マルチコアプロセッサとして)、FPGA、ASIC、およびDSPとして実装される可能性がある。プロセッサ430は、着信ポート410、受信機ユニット420、送信機ユニット440、発信ポート450、およびメモリ460と通信する可能性がある。プロセッサ430は、コーディングモジュール470を含む可能性がある。コーディングモジュール470は、上述のおよび下で説明される開示された実施形態を実装する。たとえば、コーディングモジュール470は、様々なコーディング動作を実装するか、処理するか、準備するか、または提供する可能性がある。したがって、コーディングモジュール470を含むことは、ビデオコーディングデバイス400の機能を大幅に改善し、ビデオコーディングデバイス400の異なる状態への転換をもたらす。あるいは、コーディングモジュール470は、メモリ460に記憶され、プロセッサ430によっ

10

20

30

40

50

て実行される命令として実装される可能性がある。

【0152】

メモリ460は、1つまたは複数のディスク、テープドライブ、およびソリッドステートドライブを含む可能性があり、プログラムが実行するために選択されるときにそのようなプログラムを記憶するためならびにプログラムの実行中に読まれる命令およびデータを記憶するためのオーバーフローデータ記憶装置(overflow data storage device)として使用される可能性がある。メモリ460は、たとえば、揮発性および/または不揮発性である可能性があり、読み出し専用メモリ(ROM)、ランダムアクセスメモリ(RAM)、3値連想メモリ(TCAM: ternary content-addressable memory)、および/またはスタティックランダムアクセスメモリ(SRAM)である可能性がある。

10

【0153】

図5は、例示的な実施形態による、図1Aの送信元デバイス12および送信先デバイス14のどちらかまたは両方として使用される可能性がある装置500の簡略化されたブロック図である。

【0154】

装置500のプロセッサ502は、中央演算処理装置であることが可能である。あるいは、プロセッサ502は、既存のまたは今後開発される、情報を操作または処理することができる任意の他の種類の1つのデバイスまたは複数のデバイスであることが可能である。開示される実装は示されるように単一のプロセッサ、たとえば、プロセッサ502によって実施され得るが、2つ以上のプロセッサを使用することによって速度および効率面の利点が発現され得る。

20

【0155】

装置500のメモリ504は、実装において、読み出し専用メモリ(ROM)デバイスまたはランダムアクセスメモリ(RAM)デバイスであることが可能である。任意の他の好適な種類の記憶装置が、メモリ504として使用され得る。メモリ504は、バス512を使用してプロセッサ502によってアクセスされるコードおよびデータ506を含み得る。メモリ504は、オペレーティングシステム508およびアプリケーションプログラム510をさらに含むことが可能であり、アプリケーションプログラム510は、プロセッサ502が本明細書において説明される方法を実行すること可能にする少なくとも1つのプログラムを含む。たとえば、アプリケーションプログラム510は、本明細書において説明される方法を実行するビデオコーディングアプリケーションをさらに含むアプリケーション1からNを含み得る。

30

【0156】

装置500は、ディスプレイ518などの1つまたは複数の出力デバイスも含み得る。ディスプレイ518は、一例において、ディスプレイをタッチ入力を感じ取るように動作可能であるタッチ感知要素と組み合わせるタッチ式ディスプレイである可能性がある。ディスプレイ518は、バス512を介してプロセッサ502に結合され得る。

【0157】

ここでは単一のバスとして示されるが、装置500のバス212は、複数のバスから構成され得る。さらに、二次ストレージ(図示せず)は、装置500の他の構成要素に直接結合されることが可能であり、またはネットワークを介してアクセスされることが可能であり、メモリカードなどの単一の統合されたユニットもしくは複数のメモリカードなどの複数のユニットを含むことが可能である。したがって、装置500は、多種多様な構成で実装され得る。

40

【0158】

動きベクトルの改善(MVR)

通常、動きベクトルは、エンコーダ側で少なくとも部分的に決定され、符号化されたビットストリーム内でデコーダにシグナリングされる。しかし、動きベクトルは、ビットストリーム内に示された初期動きベクトルから開始してデコーダにおいて(およびエンコーダにおいても)改善される可能性もある。そのような場合、たとえば、初期動きベクトルによって指し示された既に復号されたピクセルのパッチまたはブロックの間の類似性が、初期

50

動きベクトルの正確性を向上させるために使用される可能性がある。そのような動きベクトルの改善は、シグナリングのオーバーヘッドを削減するという利点をもたらす、つまり、初期の動きベクトルの正確性が、エンコーダとデコーダとの両方において同じ方法で高められ、したがって、改善のための追加のシグナリングが、必要とされない。

#### 【0159】

改善前の初期動きベクトルは、最良の予測をもたらす最良の動きベクトルではない可能性があることが留意される。初期動きベクトルはビットストリーム内でシグナリングされるので、初期動きベクトルを(ビットレートを高くする)非常に高い正確性で表現することが不可能である可能性がある。したがって、動きベクトル改善プロセスが、初期動きベクトルをより良くするために利用される。初期動きベクトルは、たとえば、現在のブロックの近傍のブロックの予測において使用される動きベクトルである可能性がある。この場合、どの近傍のブロックの動きベクトルが現在のブロックによって使用されるかを示すインジケーションをビットストリーム内でシグナリングすれば十分である。そのような予測メカニズムは、初期動きベクトルを表すためのビット数を削減するのに非常に有効である。しかし、概して、2つの近傍のブロックの動きベクトルは同一であると予測されないので、初期動きベクトルの正確性は低い可能性がある。

10

#### 【0160】

シグナリングのオーバーヘッドのさらなる増加なしに動きベクトルの正確性をさらに高めるために、エンコーダ側で導出され、ビットストリーム内で提供される(シグナリングされる)動きベクトルをさらに改善することが、有益である可能性がある。動きベクトルの改善は、エンコーダからの支援なしにデコーダにおいて実行される可能性がある。エンコーダは、そのエンコーダのデコーダグループ内で、デコーダにおいて利用可能である、対応する改善された動きベクトルを取得するための同じ改善を使用する可能性がある。現在のピクチャ内の再構築されている現在のブロックに関する改善は、再構築されたサンプルのテンプレートを決定し、現在のブロックのための初期動き情報の周辺の探索空間(search space)を決定し、探索空間内でテンプレートに最もよく一致する参照ピクチャの部分を見つけることによって実行される。最もよく一致する部分は、現在のブロックに関する改善された動きベクトルを決定し、そして、その改善された動きベクトルが、現在のブロック、つまり、再構築されている現在のブロックに関するインター予測されたサンプルを得るために使用される。

20

30

#### 【0161】

動きベクトルの改善は、図2のインター予測ユニット244および図3の344の一部である。

#### 【0162】

動きベクトルの改善は、以下のステップに従って実行される可能性がある。

#### 【0163】

概して、初期動きベクトルが、ビットストリーム内のインジケーションに基づいて決定され得る。たとえば、候補動きベクトルのリスト内の位置を示すインデックスが、ビットストリーム内でシグナリングされる可能性がある。別の例においては、動きベクトル予測子インデックスおよび動きベクトルの差の値が、ビットストリーム内でシグナリングされ得る。ビットストリーム内のインジケーションに基づいて決定される動きベクトルは、初期動きベクトルであるものとして定義される。現在のブロックに関するインター予測が2つの動きベクトルによって決定されるサンプルの予測されたブロックの重み付けされた組合せとして取得される双予測(bi-prediction)の場合、第1の参照ピクチャリストL0の第1の参照ピクチャ内の初期動きベクトルがMV0と表記され、第2の参照ピクチャリストL1の第2の参照ピクチャ内の初期動きベクトルがMV1と表記されるものとする。

40

#### 【0164】

初期動きベクトルを使用して、改善候補動きベクトル(MV)のペアが決定される。少なくとも2つの改善候補のペアが、決定される可能性がある。概して、改善候補動きベクトルのペアは、初期動きベクトルのペア(MV0, MV1)に基づいて決定される。さらに、候補M

50

Vのペアは、MV0およびMV1に小さな動きベクトルの差を足すことによって決定される可能性がある。たとえば、候補MVのペアは、以下を含む可能性がある。

- ・ (MV0, MV1)
- ・ (MV0 + (0,1), MV1 + (0,-1))
- ・ (MV0 + (1,0), MV1 + (-1,0))
- ・ (MV0 + (0,-1), MV1 + (0,1))
- ・ (MV0 + (-1,0), MV1 + (1,0))
- ・ ...

ここで、(1,-1)は、水平(またはx)方向の1整数ピクセルの変位および垂直(またはy)方向の-1整数ピクセルの変位を有するベクトルを表す。

【0165】

候補MVのペアの上のリストは、例示の目的のための単なる例であり、本開示は、候補の特定のリストに限定されないことが留意される。

【0166】

改善候補動きベクトル(MV)のペアは、動きベクトル改善プロセスの探索空間を形成する。

【0167】

現在のブロックの双予測においては、リストL0のそれぞれの第1の動きベクトルおよびリストL1の第2の動きベクトルを使用して得られた2つの予測ブロックが、単一の予測信号またはブロックへと組み合わせられ、これは、単予測(uni-prediction)よりも元の信号により優れた適応をもたらすことができ、結果として、より少ない残差情報と、おそらくは、より効率的な圧縮とをもたらす。

【0168】

動きベクトルの改善においては、候補MVのペアのそれぞれの第1の動きベクトルおよび第2の動きベクトルを使用して得られた2つの予測ブロックが、改善候補MVのペアの各々に関して類似性の測定基準に基づいて互いに比較される。通常、最も高い類似性をもたらす候補MVのペアが、MV0'およびMV1'、つまり、それぞれリストL0の第1の参照ピクチャ内の改善された動きベクトルおよびリストL1の第2の参照ピクチャ内の改善された動きベクトル、と表記される、改善された動きベクトルとして選択される。言い換えると、候補動きベクトルのペアのリストL0の動きベクトルおよびリストL1の動きベクトルに対応する予測が得られ、それから、それらの予測が、類似性の測定基準に基づいて互いに比較される。最も高い関連する類似性を有する候補動きベクトルのペアが、改善されたMVのペアとして選択される。

【0169】

概して、改善プロセスの出力は、改善されたMVである。改善されたMVは、初期MVと同じである可能性がありまたは初期MVと異なる可能性がある。どの候補MVのペアが最も高い類似性を実現するかに応じて、初期MVによって形成される候補MVのペアも、候補MVのペアの中にある。言い換えると、最も高い類似性を実現する候補MVのペアが初期MVによって形成される場合、改善されたMVおよび初期MVは、互いに等しい。

【0170】

類似性の測定基準を最大化する位置を選択する代わりに、別の方法は、相違の測定基準を最小化する位置を選択することである。相違の比較の尺度は、SAD(差分絶対値和)、MR SAD(平均を引いた差分絶対値和: mean removed sum of absolute differences)、SSE(残差平方和)などである可能性がある。2つの予測ブロックの間のSADが、候補MVのペア(CMV0, CMV1)を使用して取得される可能性があり、SADは、以下のように計算され得る。

【0171】

【数2】

10

20

30

40

50

$$\text{SAD}(\text{CMV0}, \text{CMV1}) = \sum_{x=0}^{\text{nCbW}-1} \sum_{y=0}^{\text{nCbH}-1} \text{abs}(\text{predSamplesL0}[x][y] - \text{predSamplesL1}[x][y])$$

## 【0172】

式中、nCbHおよびnCbWは、予測ブロックの高さおよび幅であり、関数abs(a)は、引数aの絶対値を指定し、predSamplesL0およびpredSamplesL1は、(CMV0, CMV1)によって表される候補MVのペアによって得られる予測ブロックサンプルである。

10

## 【0173】

あるいは、相違の比較の尺度は、計算の回数を減らすために予測ブロック内のサンプルのサブセットのみを評価することによって取得される可能性がある。例は、以下に与えられ、サンプルの行が、代替的にSAD計算に含められる(1行おきの行が評価される)。

## 【0174】

## 【数3】

$$\text{SAD}(\text{CMV0}, \text{CMV1})$$

$$= \sum_{x=0}^{\text{nCbW}-1} \sum_{y=0}^{\text{nCbH}/2-1} \text{abs}(\text{predSamplesL0}[x][2 * y] - \text{predSamplesL1}[x][2 * y])$$

20

## 【0175】

動きベクトルの改善の例示としての一例が、<http://phenix.it-sudparis.eu/jvet/>に公開されている(ITU-T SG 16 WP 3およびISO/IEC JTC 1/SC 29/WG 11の)JVETの文書JVET-M1001-v3、「Versatile Video Coding (Draft 4)」に説明されている。文書のセクション「8.4.3 Decoder side motion vector refinement process」が、動きベクトルの改善を例示する。

30

## 【0176】

改善のための内部メモリの要件を下げるために、一部の実施形態において、動きベクトル改善プロセスは、ルマサンプル内の特定の予め決められた幅および/または予め決められた高さを超えるサンプルの符号化ブロックをルマの予め決められた幅および予め決められた高さ以下であるサンプルの下位ブロックに区分けすることによって得られたルマサンプルのブロックに対して独立して実行される可能性がある。区分けされた符号化ブロック内の各下位ブロックに関する改善されたMVのペアは、異なる可能性がある。そして、ルマとクロマとの両方に関するインター予測が、各下位ブロックに関してその下位ブロックの改善されたMVのペアを使用して実行される。

40

## 【0177】

初期MVのペアの各MVは、小数(fractional)ピクセルの精度を持ち得る。言い換えると、MVは、サンプルの現在のブロックと再サンプリングされた参照領域との間の変位を示し、この変位は、再構築された参照サンプルの整数グリッドから水平方向および垂直方向の小数位置を指し示し得る。概して、再構築された参照サンプルの整数グリッドの2次元補間が、小数サンプルオフセット位置のサンプル値を得るために実行される。候補MVのペアを使用して再構築された参照ピクチャから予測されたサンプルを取得するプロセスは、以下の方法のうちの1つによる可能性がある。

・初期MVのペアの小数部を最も近い整数位置に丸め、再構築された参照ピクチャの整数グリッド値を得る。

50

・初期MVのペアによって示される小数ピクセルの正確性で予測されたサンプル値を得るために2タップ(たとえば、バイリニア)の分離可能なバイリニア補間を実行する。

・初期MVのペアによって示される小数ピクセルの正確性で予測されたサンプル値を得るためにより多いタップ(たとえば、8タップまたは6タップ)の分離可能な補間を実行する。

#### 【0178】

候補MVのペアは初期MVのペアに対して任意のサブピクセルのオフセットを持ち得るが、一部の実施形態においては、探索を簡単にするために、初期MVのペアに対して整数ピクセルの距離または整数の変位の候補MVのペアが、選択される。初期MVのペアに対して整数ピクセルの距離を有するそのような候補MVのペアは、動きベクトルの改善(MVR)プロセスの第1の段階、いわゆるMVRの整数距離改善段階において使用される可能性もある。そのような場合、すべての候補MVのペアの予測されたサンプルが、初期MVのペアの周りのすべての改善の位置を包含するように初期MVのペアの周りのサンプルのブロックに関する予測を実行することによって取得され得る。

10

#### 【0179】

一部の実施形態においては、初期MVのペアから整数の距離(整数の変位)にあるすべての候補MVのペアにおける相違のコスト値が評価されると、最良のコスト値の位置からサブピクセルの距離オフセットにある追加の候補MVのペアが、追加される可能性がある。この第2の段階は、MVRプロセスの小数の距離の改善段階として実行される可能性がある。予測されたサンプルが、上述の方法のうちの1つを使用してこれらの位置の各々に関して取得される可能性があり、相違のコストが、最も低い相違のコストを有する位置を得るために評価され、互いに比較される。特定の他の実施形態においては、最も低いコストの整数の距離の位置の周りのそれぞれのサブピクセルの距離の位置に関するこの計算コストの高い予測プロセスを避けるために、評価された整数の距離のコスト値が、記憶され、パラメトリック誤差曲面が、整数の距離の位置の近くに当てはめられる。そして、この誤差曲面の最小値が、解析的に計算され、最小の相違を有する位置として使用される。そのような場合、相違のコスト値は、計算された整数の距離のコスト値から導出されると言われる。

20

#### 【0180】

サンプルの所与の符号化ブロックのための動きベクトルの改善の適用は、サンプルの符号化ブロックの特定のコーディングのプロパティによって条件付けられる可能性がある。そのようなコーディングのプロパティのいくつかの例は、以下を含む可能性がある。

30

・現在のピクチャからサンプルのコーディングブロックの双予測のために使用される2つの参照ピクチャまでの(一様なフレームレートでサンプリングされるとき)ピクチャの数で表された距離、またはピクチャ順序カウント(POC)の差が、等しく、現在のピクチャの両側で減少する。

・初期MVのペアを使用して得られた2つの予測されたブロックの間の初期の相違が、予め決められたサンプル毎の閾値未満である。

#### 【0181】

双予測のオブティカルフローの改善(BPOF)

双予測のオブティカルフロー(BPOF)または双方向のオブティカルフロー(BDOF)の改善は、双予測のためのシグナリング以外のビットストリーム内の明らかな追加的なシグナリングなしにブロックの双予測の正確性を高めるプロセスである。双予測のオブティカルフローまたは双方向のオブティカルフローの改善は、図2のインター予測ユニット244および図3の344の一部である。

40

#### 【0182】

双予測においては、2つのインター予測が、2つの動きベクトルに従って得られ、それから、予測が、加重平均の適用によって組み合わせられる。組み合わせられた予測は、2つの参照パッチまたはブロック内の量子化雑音が打ち消されるので、削減された残差エネルギーをもたらすことができ、それによって、単予測よりも高い符号化効率を提供する。双予測の重み付けされた組合せが、以下の式

50

$$\text{Bi-prediction} = \text{Prediction1} * W1 + \text{Prediction2} * W2 + K$$

に従って実行されることが可能であり、式中、W1およびW2は、ビットストリーム内でシグナリングされる可能性があり、またはエンコーダ側および/もしくはデコーダ側で予め定義される可能性がある重み係数である。Kは、やはりビットストリーム内でシグナリングされるかまたはエンコーダ側および/もしくはデコーダ側で予め定義される可能性がある加算因子である。例として、双予測は、

$$\text{Bi-prediction} = (\text{Prediction1} + \text{Prediction2})/2$$

を使用して得られる可能性があり、式中、W1およびW2は、1/2に設定され、Kは、0に設定される。

【0183】

オプティカルフローの改善の目的は、双予測の正確性を高めることである。オプティカルフローは、2つの連続するフレーム間の画像オブジェクトの目に見える動きのパターンである。オプティカルフローは、オブジェクトおよび/またはカメラの移動によって引き起こされる。オプティカルフロー改善プロセスは、オプティカルフローの式の適用(オプティカルフローの式を解くこと)によって双予測の正確性を高める。

【0184】

例においては、ピクセル $l(x,y,t)$ が、第1のフレーム内にある( $x$ および $y$ は、空間座標に対応し、 $t$ は、時間の次元に対応する)。ピクセルによって表されるオブジェクトが、時間 $d$ の後を取得された次のフレームにおいて距離( $dx, dy$ )だけ移動する。それらのピクセルが同じであり、強度が変わらないという仮定の下で、オプティカルフローの式は、

$$l(x,y,t) = l(x+dx, y+dy, t+dt)$$

によって与えられ、式中、 $l(x,y,t)$ は、座標( $x,y,t$ )のピクセルの強度(サンプル値)を指定する。

【0185】

別の例においては、テイラー級数展開におけるより高次の項が無視される小さな変位の場合、オプティカルフローの式は

【0186】

【数4】

$$\frac{\partial l}{\partial t} + v_x \frac{\partial l}{\partial x} + v_y \frac{\partial l}{\partial y} = 0$$

【0187】

のように記述されることも可能であり、式中、

【0188】

【数5】

$$\frac{\partial l}{\partial x}$$

【0189】

および

【0190】

【数6】

$$\frac{\partial l}{\partial y}$$

10

20

30

40

50

【 0 1 9 1 】

は、位置(x,y)における水平方向および垂直方向の空間的なサンプルの勾配であり、

【 0 1 9 2 】

【 数 7 】

$$\frac{\partial I}{\partial t}$$

【 0 1 9 3 】

は、位置(x,y)における時間偏導関数であり、 $v_x=dx/dt$ および $v_y=dy/dt$ は、変位の速度を表す。

【 0 1 9 4 】

オプティカルフローの改善は、双予測の品質を高めるために上の原理を利用する。

【 0 1 9 5 】

オプティカルフローの改善の実装は、概して、以下のステップを含む。

1. サンプルの勾配を計算する
2. 第1の予測と第2の予測との間の差を計算する
3. オプティカルフローの式を使用して2つの参照パッチまたはブロックの間の誤差

【 0 1 9 6 】

【 数 8 】

$$\Delta = (I^{(0)} - I^{(1)}) + v_x(\tau_0 \frac{\partial I^{(0)}}{\partial x} + \tau_1 \frac{\partial I^{(1)}}{\partial x}) + v_y(\tau_0 \frac{\partial I^{(0)}}{\partial y} + \tau_1 \frac{\partial I^{(1)}}{\partial y})$$

【 0 1 9 7 】

を最小化するピクセルまたはピクセルのグループの変位(dx, dy)を計算し、式中、 $I^{(0)}$ は、第1の予測のサンプル値に対応し、 $I^{(1)}$ は、第2の予測のサンプル値であり、 $v_x$ および $v_y$ は、xおよびy方向において計算された変位速度であり、 $I^{(0)}/x$ 、 $I^{(0)}/y$ 、 $I^{(1)}/x$ 、および $I^{(1)}/y$ は、それぞれ、第1および第2の予測の-xおよび-y方向の勾配である。 $\tau_1$ および $\tau_0$ は、表示順に現在のピクチャからそれぞれの参照ピクチャまでの時間的距離を表し、第1の予測および第2の予測が、得られる。一部の手法は、残差平方和を最小化し、一方、一部の手法は、絶対誤差の和を最小化する。

4. 以下のもののようなオプティカルフローの式の実装を使用する。

$$\text{pred}_{\text{BIO}} = 1/2 \cdot (I^{(0)} + I^{(1)}) + v_x/2 \cdot (\tau_1 I^{(1)}/x - \tau_0 I^{(0)}/x) + v_y/2 \cdot (\tau_1 I^{(1)}/y - \tau_0 I^{(0)}/y)$$

式中、 $\text{pred}_{\text{BIO}}$ は、オプティカルフロー改善プロセスの出力である修正された予測を指定する。

【 0 1 9 8 】

サンプルの勾配は、以下の式に従って取得される可能性がある。

- ・  $I(x, y, t)/x = I(x + 1, y, t) - I(x - 1, y, t)$
- ・  $I(x, y, t)/y = I(x, y + 1, t) - I(x, y - 1, t)$

【 0 1 9 9 】

一部の実施形態においては、各ピクセルに関する変位の推定の複雑さを減らすために、変位が、ピクセルのグループに関して推定される。一部の例においては、 $4 \times 4$ ルマサンプルのブロックに関する改善された双予測を計算するために、変位が、サンプルの $4 \times 4$ ブロックをその中心として $8 \times 8$ ルマサンプルのブロックのサンプル値を使用して推定される。

【 0 2 0 0 】

オプティカルフロー改善プロセスの入力は、2つの参照ピクチャからの予測サンプルで

10

20

30

40

50

あり、オプティカルフローの改善の出力は、オプティカルフローの式によって計算される組み合わせられた予測(predBIO)である。

【 0 2 0 1 】

オプティカルフローの改善の一例が、<http://phenix.it-sudparis.eu/jvet/>に公開されており、参照のために以下で参照される文書JVET-N1001、Versatile Video Coding (Draft 5)のセクション8.5.6.4「Bidirectional optical flow prediction process」に説明されている。

【 0 2 0 2 】

このプロセスへの入力は、以下である。

- 現在のコーディングブロックの幅および高さを指定する2つの変数nCbWおよびnCbH
- 2つの $(nCbW + 2) \times (nCbH + 2)$ ルマ予測サンプル配列predSamplesL0およびpredSamplesL1
- 予測リストの利用フラグpredFlagL0およびpredFlagL1
- 参照インデックスrefIdxL0およびrefIdxL1
- 双方向オプティカルフローの利用フラグbdofUtilizationFlag[ xIdx ][ yIdx ] with  $xIdx = 0..(nCbW - 2) - 1$ ,  $yIdx = 0..(nCbH - 2) - 1$

【 0 2 0 3 】

このプロセスの出力は、ルマ予測サンプル値の $(nCbW) \times (nCbH)$ 配列pbSamplesである。

【 0 2 0 4 】

変数bitDepth、shift1、shift2、shift3、shift4、offset4、およびmvRefineThresは、以下のように導出される。

- 変数bitDepthは、 $BitDepth_Y$ 、すなわちルマ成分のビット深度に等しく設定される。
- 変数shift1は、 $Max(6, bitDepth - 6)$ に等しくなるように設定される。
- 変数shift2は、 $Max(4, bitDepth - 8)$ に等しくなるように設定される。
- 変数shift3は、 $Max(1, bitDepth - 11)$ に等しくなるように設定される。
- 変数shift4は、 $Max(3, 15 - bitDepth)$ に等しく設定され、変数offset4は、 $1 + (shift4 - 1)$ に等しく設定される。
- 変数mvRefineThresは、 $1 + Max(5, bitDepth - 7)$ に等しく設定される。

【 0 2 0 5 】

$xIdx = 0..(nCbW - 2) - 1$ および $yIdx = 0..(nCbH - 2) - 1$ に関して、以下が適用される。

- 変数xSbは、 $(xIdx + 2) + 1$ に等しく設定され、ySbは、 $(yIdx + 2) + 1$ に等しく設定される。
- bdofUtilizationFlag[ xIdx ][ yIdx ]がFALSEに等しい場合、 $x = xSb - 1..xSb + 2$ ,  $y = ySb - 1..ySb + 2$ に関して、現在のサブブロックの予測サンプル値は、以下のように導出される。

$pbSamples[x][y] = Clip3(0, (2^{bitDepth}) - 1, (predSamplesL0[x + 1][y + 1] + offset4 + predSamplesL1[x + 1][y + 1]) \ll shift4)$

- そうでない場合(bdofUtilizationFlag[ xIdx ][ yIdx ]がTRUEに等しい)、現在のサブブロックの予測サンプル値は、以下のように導出される。

-  $x = xSb - 1..xSb + 4$ ,  $y = ySb - 1..ySb + 4$ に関して、以下の順序付けられたステップが適用される。

1. 予測サンプル配列内の対応するサンプル位置 $(x, y)$ の各々に関する位置 $(h_x, v_y)$ は、以下のように導出される。

$$h_x = Clip3(1, nCbW, x)$$

$$v_y = Clip3(1, nCbH, y)$$

2. 変数gradientHL0[ x ][ y ], gradientVL0[ x ][ y ], gradientHL1[ x ][ y ], およびgradientVL1[ x ][ y ]は、以下のように導出される。

$$gradientHL0[x][y] = (predSamplesL0[h_x + 1][v_y] - predSampleL0[h_x][v_y])$$

10

20

30

40

50

$x - 1][v_y]$  )      shift1  
 $\text{gradientVL0}[x][y] = (\text{predSampleL0}[h_x][v_y + 1] - \text{predSampleL0}[h_x][v_y - 1])$       shift1  
 $\text{gradientHL1}[x][y] = (\text{predSamplesL1}[h_x + 1][v_y] - \text{predSampleL1}[h_x - 1][v_y])$       shift1  
 $\text{gradientVL1}[x][y] = (\text{predSampleL1}[h_x][v_y + 1] - \text{predSampleL1}[h_x][v_y - 1])$       shift1

3.変数diff[x][y]、tempH[x][y]、およびtempV[x][y]は、以下のように導出される。

$\text{diff}[x][y] = (\text{predSamplesL0}[h_x][v_y] \text{ shift2}) - (\text{predSamplesL1}[h_x][v_y] \text{ shift2})$       10

$\text{tempH}[x][y] = (\text{gradientHL0}[x][y] + \text{gradientHL1}[x][y])$       shift3

$\text{tempV}[x][y] = (\text{gradientVL0}[x][y] + \text{gradientVL1}[x][y])$       shift3

- 変数sGx2、sGy2、sGxGy、sGxdI、およびsGydIは、以下のように導出される。

$i, j = -1..4$ として、 $sGx2 = \sum_i \sum_j (\text{tempH}[xSb + i][ySb + j] * \text{tempH}[xSb + i][ySb + j])$

$i, j = -1..4$ として、 $sGy2 = \sum_i \sum_j (\text{tempV}[xSb + i][ySb + j] * \text{tempV}[xSb + i][ySb + j])$

$i, j = -1..4$ として、 $sGxGy = \sum_i \sum_j (\text{tempH}[xSb + i][ySb + j] * \text{tempV}[xSb + i][ySb + j])$       20

$sGxGy_m = sGxGy \quad 12$

$sGxGy_s = sGxGy \& ((1 \quad 12) - 1)$

$i, j = -1..4$ として、 $sGxdI = \sum_i \sum_j (-\text{tempH}[xSb + i][ySb + j] * \text{diff}[xSb + i][ySb + j])$

$i, j = -1..4$ として、 $sGydI = \sum_i \sum_j (-\text{tempV}[xSb + i][ySb + j] * \text{diff}[xSb + i][ySb + j])$

- 現在のサブブロックの水平および垂直動きオフセットは、以下のように導出される。

$v_x = sGx2 \quad 0 ? \text{Clip3}(-\text{mvRefineThres}, \text{mvRefineThres}, -(\text{sGxdI} \quad 3) \text{ Floor}(\text{Log2}(sGx2))) : 0$

$v_y = sGy2 \quad 0 ? \text{Clip3}(-\text{mvRefineThres}, \text{mvRefineThres}, ((\text{sGydI} \quad 3) \quad 30) - ((v_x * sGxGy_m) \quad 12 + v_x * sGxGy_s) \quad 1) \text{ Floor}(\text{Log2}(sGx2))) : 0$

-  $x = xSb - 1..xSb + 2$ ,  $y = ySb - 1..ySb + 2$ に関して、現在のサブブロックの予測サンプル値は、以下のように導出される。

$\text{bdofOffset} = \text{Round}((v_x * (\text{gradientHL1}[x + 1][y + 1] - \text{gradientHL0}[x + 1][y + 1])) \quad 1) + \text{Round}((v_y * (\text{gradientVL1}[x + 1][y + 1] - \text{gradientVL0}[x + 1][y + 1])) \quad 1)$

[Round()演算は、float入力に対して定義される。Round()演算は、ここでは省略される可能性がある。]

$\text{pbSamples}[x][y] = \text{Clip3}(0, (2^{\text{bitDepth}}) - 1, (\text{predSamplesL0}[x + 1][y + 1] + \text{offset4} + \text{predSamplesL1}[x + 1][y + 1] + \text{bdofOffset}) \quad \text{shift4})$       40  
**【0206】**

オプティカルフローの改善、双予測のオプティカルフローの改善、および双方向のオプティカルフローの改善という用語は、用語が本質的に等価であるので本開示において交換可能であるように使用される。

**【0207】**

本開示が対処する技術的問題

現在採用されているBPOFのバージョンにおいて、BPOFのためのメモリフットプリントを削減し、ハードウェア設計においてサブブロックレベルのパイプラインが確立されることを可能にするために、コーディングユニットが、サブブロックに区分けされる。16よりも大きい幅を有する任意のコーディングユニットが、16のサブブロック幅で水平方向に区

分けされる。16よりも大きい高さを有する任意のコーディングユニットが、16のサブブロック高さで垂直方向に区分けされる。

【0208】

オプティカルフローを計算し、計算されたオプティカルフローに基づいて補正を適用するために、(BPOFの背景セクションにおいて説明されたように)4×4ブロック内のすべてのサンプル位置に関して、サンプル勾配が、必要とされる。4×4のサンプル勾配を取得することは、4×4のサンプル値を超えて、両方の参照において拡張されたサンプル値を必要とする。勾配を計算するために[-1 0 1]フィルタを使用する場合、各参照において、4×4ブロック全周に1つの拡張されたサンプル値が必要とされる。言い換えると、4×4のサンプル勾配は、6×6のサンプルエリアを必要とする(6×6エリアの4つのコーナーサンプルを除く)。

10

【0209】

サブブロックレベルにおいて、サイズM×Nのサブブロックが、M×Nのサンプル位置に関するサンプル勾配を計算するために(M+2)×(N+2)のサンプル値を必要とする。複雑さを低く保つために、現在採用されているBDofのバージョンにおいて、拡張されたサンプル値(すなわち、サブブロックに対応するM×Nのサンプル位置の外側にあるサンプル位置)は、それぞれの参照サンプルの動き補償補間によって取得されない。代わりに、各参照ピクチャにおいてサブピクセル精度で拡張されたサンプル位置が、取得され、水平および垂直方向における小数部分は、各参照における対応する整数グリッドで再構築された参照サンプル位置のサンプル値を取得するために無視される。

20

【0210】

これらの拡張されたサンプル値は、各参照における境界サンプル勾配の計算において使用される。4×4のオプティカルフローの計算は、4×4を中心とする6×6のサンプル位置のサンプル勾配値とサンプル値とを使用するので、拡張された位置のサンプル勾配値が、最も近いサブブロック境界サンプルのサンプル勾配値に設定される。整数グリッドで再構築された参照サンプル位置の値から取得された拡張されたサンプル値が、境界勾配計算のために使用されると、拡張されたサンプルのサンプル値は、最も近いサブブロック境界サンプルのサンプル値に設定される。

【0211】

拡張されたサンプルのための整数グリッドで再構築された参照サンプル値を取得するために、サンプル位置の水平および垂直方向における小数部分を削除するこの態様は、JVET-M0487に示されているように最も近い境界サンプル値を取ってパディングされたサンプルを用いて2次元の分離可能なKタップ補間を実行する場合と比較して、小さい圧縮損失を導入する。

30

【0212】

提案された方法

本開示において、小数の水平および垂直部分の削除によって導入されるコーディング損失をほぼ除去しながら、拡張されたサンプル値を取得することの低い複雑さの側面を維持する方法が、提供される。

【0213】

詳細には、各再構築された参照サンプルグリッド内の拡張されたサンプル位置の小数の水平および垂直オフセットを無視する代わりに、小数オフセットは、対応する再構築された参照フレームの整数サンプルグリッド内の最も近い整数サンプル位置を識別するために、水平方向と垂直方向との両方において丸め演算を実行するために使用される。そのような識別された位置におけるサンプル値は、次いで、拡張されたサンプルに割り当てられる。

40

【0214】

概して、現在のブロックの双方向のオプティカルフロー予測において使用される予測されたブロックの拡張された領域に属するサンプル位置からのサンプル値に基づく双方向のオプティカルフロー予測のための提案された方法が、図8に示されるように、以下のステップによって実施される可能性がある。

50

ステップ1310、現在のブロックの動きベクトル(MV)を取得する。

ステップ1320、MVに基づいて、参照ピクチャ内の予測されたブロックの拡張された領域内のサブピクセルサンプル位置を取得する。

ここで、参照ピクチャ内の予測されたブロックの拡張された領域は、MVに基づく。特に、MVを使用することによって、現在のブロックに対応する併置されたブロックへの参照としてMVによって示されるオフセットを有する予測ブロックが、見つけれ得る。拡張された領域は、予測ブロックに隣接し、予測ブロックを取り囲む/包含するピクセルの少なくとも1つの行/列を含む領域である。拡張された領域は、参照ピクチャ内の予測ブロックを取り囲むパディング領域と考えられ得、たとえば、ブロック境界におけるKタップ補間フィルタの適用および/または3タップサンプル勾配の計算に使用される。

10

ステップ1330、参照ピクチャ内の整数ピクセルサンプル位置を取得するために、取得されたサブピクセルサンプル位置を処理する。

ステップ1340、参照ピクチャ内の取得された整数ピクセルサンプル位置における参照サンプル値を参照することによって、双方向のオプティカルフロー予測を実行する。

【0215】

以下の実施形態は、この方法がどのように実施され得るかを例示する。

【0216】

実施形態1

BPOF/BDOPFによる双予測が実行されている次元M×Nサンプルの現在のサブブロックの左上サンプルが、(sb\_start\_x, sb\_start\_y)にあるものとする。(mvx0, mvy0)が、第1の参照フレームL0に対する現在のサブブロックの1/16のピクセル精度のMVであり、(mvx1, mvy1)が、第2の参照フレームL1に対する現在のサブブロックの1/16のピクセル精度のMVであるとする。

20

【0217】

L0内の現在のサブブロックの左上の位置に対応するL0内の1/16のピクセル精度の位置は、((sb\_start\_x + 4) + mvx0, (sb\_start\_y + 4) + mvy0)によって与えられる。L0内の現在のサブブロックの左上の位置に対応するL1内の1/16のピクセル精度の位置は、((sb\_start\_x + 4) + mvx1, (sb\_start\_y + 4) + mvy1)によって与えられる。半ペル、1/4ペル、またはペルの1/8などの様々なサブピクセル精度が、使用される可能性があり、現在のサブブロックの左上の位置は、それぞれ、1、2、または3だけ対応する左シフトを適用することによって決定される。

30

【0218】

サブブロックに対応するL0およびL1の参照フレーム内のM×Nの位置に関して、2次元の分離可能なKタップ動き補償補間が、たとえば、1/16のサブピクセル精度の対応するサンプル位置に基づいて、それぞれの参照フレームにおいて実行され、サンプル勾配を計算するためにBDOPFプロセスにおいて使用される予測されたサンプル値として使用される。

【0219】

M×Nのサブブロック境界サンプル勾配計算の目的のために、サンプル位置の(M+2)×(N+2)ブロック内の中央のM×Nサンプル位置の外側にある残りの拡張された対応するサンプル位置に関する予測されたサンプル値を取得するために、以下の手順(一般的なサブペルサンプル精度の場合)に従う。

40

x\_spel\_posが、拡張されたサンプル位置の水平方向のサブペルサンプル精度の位置であり、y\_spel\_posが、垂直方向のサブペルサンプル精度の位置であるとする。この拡張されたサンプル位置に関するサンプル値は、以下の

$$\begin{aligned} x\_pos &= (x\_spel\_pos + (1 - \text{shift\_val})) \times \text{shift\_val} \\ y\_pos &= (y\_spel\_pos + (1 - \text{shift\_val})) \times \text{shift\_val} \end{aligned}$$

のように計算された位置(x\_pos, y\_pos)における再構築された参照サンプル値として取得され、式中、shift\_valは、使用されるサブペルサンプル精度に従って選択される。半ペル、1/4ペル、ペルの1/8、ペルの1/16のサブペル精度に関して、それぞれ、1、2、3、または4のshift\_valが、使用される。

50

## 【 0 2 2 0 】

水平および垂直方向においてサブピクセル位置を丸めるこのプロセスによって、図6および図7に見られ得るように、境界勾配は、より正確になる。

## 【 0 2 2 1 】

図6において、参照ピクチャ内のどの位置からピクセル値がフェッチされるかが、示される。最新技術を示す図面(A)において、参照ピクチャ内の小数位置を指すMVとして、小数部分は、破棄され、整数部分のみが、ピクセル値をフェッチするために使用される。たとえば、Q4フォーマットにおけるMV (4, 0)、MV (9, 0)に関して、参照ピクチャ内の(0, 0)位置は、ピクセル値(参照サンプル)をフェッチするために使用される。対照的に、本開示による図面(B)において、小数部分は、ピクセル値をフェッチするために最終的に使用される最も近い整数位置に丸めるために考慮される。Q4フォーマットにおけるMV (4, 0)、MV (9, 0)に関して、参照ピクチャ内の位置(0, 0)および位置(16, 0)は、それぞれ、参照サンプルをフェッチする精度を大幅に向上させることができるそれぞれのピクセル値をフェッチするために使用される。

10

## 【 0 2 2 2 】

図7Bにおいて、サブピクセル精度(すなわち、ピクセルの1/16)のサンプル位置に基づいて整数グリッド参照サンプル位置を決定するための説明された方法が、示される。水平および垂直方向において、最も近い整数サンプル位置は、位置の水平および垂直成分を(半分)丸めることに基づいて選択されることがわかる。したがって、水平方向において( $i - 8/16$ )から( $i + 7/16$ )まで、垂直方向において( $j - 8/16$ )から( $j + 7/16$ )までのサブピクセル精度のオフセットが、最も近い整数サンプル位置として( $i, j$ )を使用する。これと比較して、図7Aは、切り捨てによって整数サンプル位置を選択する際の問題を示し、水平方向において $i$ から( $i + 15/16$ )まで、垂直方向において $j$ から( $j + 15/16$ )までのサブピクセル精度のオフセットが、整数サンプル位置として( $i, j$ )を使用する。

20

## 【 0 2 2 3 】

$M \times N$ ブロックの左側境界サンプル位置において、水平境界サンプル勾配は、(2次元の分離可能なKタップ補間を使用して計算された)その右側の位置のサンプル値と(上で説明されたように丸められた最も近い再構築された参照サンプル値が割り当てられた)その左側の拡張されたサンプル位置のサンプル値との間の差を取ることによって計算される。同様に、 $M \times N$ ブロックの右側境界サンプル位置において、水平境界サンプル勾配は、(上で説明されたように丸められた最も近い再構築された参照サンプル値が割り当てられた)その右側の位置のサンプル値と(2次元の分離可能なKタップ補間を使用して計算された)その左側の位置のサンプル値との間の差を取ることによって計算される。

30

## 【 0 2 2 4 】

$M \times N$ ブロックの上部境界サンプル位置において、垂直境界サンプル勾配は、(2次元の分離可能なKタップ補間を使用して計算された)その下の位置のサンプル値と(上で説明されたように丸められた最も近い再構築された参照サンプル値が割り当てられた)上の拡張されたサンプル位置のサンプル値との間の差を取ることによって計算される。同様に、 $M \times N$ ブロックの下部境界サンプル位置において、垂直境界サンプル勾配は、(上で説明されたように丸められた最も近い再構築された参照サンプル値が割り当てられた)その下の位置のサンプル値と、(2次元の分離可能なKタップ補間を使用して計算された)その上の位置のサンプル値との間の差を取ることによって計算される。

40

## 【 0 2 2 5 】

## 実施形態2

この実施形態において、拡張されたサンプル位置の値は、BPOF/BDOFが適用され、動きベクトルの改善(MVR)が、現在のサブブロックが属する所与のコーディングユニットに適用されない場合、またはMVRおよびBDOFが、両方とも、現在のサブブロックが属する所与のコーディングユニットに適用されるが、MVRが初期MVのペアと比較してMVRの整数距離改善段階中に整数変位を生じない場合はいつでも、実施形態1で説明された手順を使用して導出される。言い換えると、MVRおよびBDOFが、両方とも適用され、MVRの整

50

数距離改善段階が初期MVのペア、すなわち、ゼロの変位のMVのペアを生じる場合、拡張されたサンプル位置の値は、実施形態1で説明された手順を使用して導出される。

【0226】

MVRおよびBDOFが、両方とも、現在のサブブロックが属する所与のコーディングユニットに適用されるが、MVRが、初期MVのペアと比較してMVRの整数距離改善段階中に整数変異を生じる場合、すなわち、整数距離改善段階後の改善されたMVのペアが、初期MVのペアとは異なる場合、拡張されたサンプル位置(x\_spel\_pos, y\_spel\_pos)に関するサンプル値は、以下の

$$\begin{aligned} x\_pos &= x\_spel\_pos & shift\_val \\ y\_pos &= y\_spel\_pos & shift\_val \end{aligned}$$

10

のように計算された位置(x\_pos, y\_pos)における再構築された参照サンプル値として取得され、式中、shift\_valは、使用されるサブペルサンプル精度に従って選択される。半ペル、1/4ペル、ペルの1/8、ペルの1/16のサブペル精度に関して、それぞれ、1、2、3、または4のshift\_valが、使用される。言い換えると、この場合のサンプル位置を取得する際に、水平方向と垂直方向との両方における位置の小数部分は、無視される。

【0227】

提案された方法の利点

水平および垂直方向におけるサブペル位置を丸める説明された方法により、境界勾配は、より正確になる。これは、これらのサンプル勾配に基づいて計算されたオプティカルフローを改善し、計算されたオプティカルフローを使用して適用される補正も改善する。これは、拡張されたサンプル位置のための整数サンプル値を選択するとき、水平および垂直方向における小数オフセットを無視する以前の方法と比較して、一貫したコーディングゲインを提供する。同時に、水平および垂直の丸めオフセットの追加は、複雑さにおける唯一の増加であるので、複雑さは、あまり変更されない。

20

【0228】

解決策に基づいて、標準のVVCドラフト5を組み合わせる際の可能な実施形態が、以下のように示される可能性がある。

【0229】

ルマ整数サンプルフェッチプロセス

このプロセスへの入力は、以下である。

- フルサンプルユニットにおけるルマ位置( xInt<sub>L</sub>, yInt<sub>L</sub> )
- 小数サンプルユニットにおけるルマ位置( xFrac<sub>L</sub>, yFrac<sub>L</sub> )
- ルマ参照サンプル配列refPicLX<sub>L</sub>

このプロセスの出力は、予測されたルマサンプル値predSampleLX<sub>L</sub>である。

30

【0230】

変数shiftは、Max( 2, 14 - BitDepth<sub>Y</sub> )に等しく設定される。

【0231】

変数picWは、pic\_width\_in\_luma\_samplesに等しく設定され、変数picHは、pic\_height\_in\_luma\_samplesに等しく設定される。

【0232】

フルサンプルユニットにおけるルマ位置( xInt, yInt )は、以下のように導出される。

$$xOffset = xFrac_L \ll 7 \text{ ? } 1 : 0$$

$$yOffset = yFrac_L \ll 7 \text{ ? } 1 : 0$$

$$xInt = Clip3( 0, picW - 1, sps\_ref\_wraparound\_enabled\_flag \text{ ? }$$

$$ClipH( ( sps\_ref\_wraparound\_offset\_minus1 + 1 ) * MinCbSizeY, picW, xInt_L - 1 + xOffset) : xInt_L - 1 + xOffset)$$

$$yInt = Clip3( 0, picH - 1, yInt_L - 1 + yOffset)$$

40

【0233】

予測されたルマサンプル値predSampleLX<sub>L</sub>は、以下のように導出される。

$$predSampleLX_L = refPicLX_L[ xInt ][ yInt ] \ll shift3$$

50

## 【 0 2 3 4 】

VVCドラフトにおけるこれらの例示的な変更は、ルマ成分に対してのみ行われるが、それらは、クロマ成分の特性を採用するために、同じ方法でまたはいくつかの変更を加えて、クロマ成分に対して使用され得ることに留意されたい。

## 【 0 2 3 5 】

図8は、本開示の実施形態によるビデオ符号化/復号の方法に関する流れ図を示す。示された方法は、以下のように、現在のブロックの双方向のオプティカルフロー予測において使用される予測されたブロックの拡張された領域に属するサンプル位置からのサンプル値に基づいて双方向のオプティカルフロー予測を実行する。ステップ1310において、現在のブロックの動きベクトル(MV)が取得される。ステップ1320において、参照ピクチャ内の予測されたブロックの拡張された領域内のサブピクセルサンプル位置が、MVに基づいて取得される。ステップ1330において、取得されたサブピクセルサンプル位置は、参照ピクチャ内の整数ピクセルサンプル位置を取得するために、処理され、特に、丸められる。ステップ1340において、最後に、参照ピクチャ内の取得された整数ピクセルサンプル位置における参照サンプル値を参照することによって、双方向のオプティカルフロー予測が、実行される。

## 【 0 2 3 6 】

図9は、本開示の実施形態による符号化/復号装置の例を示すブロック図である。符号化/復号装置20/30は、現在のブロックの双方向のオプティカルフロー予測を実行するように構成された予測ユニット1440と、現在のブロックの動きベクトル(MV)を取得するように構成された第1の取得ユニット1410と、MVに基づいて参照ピクチャ内の予測されたブロックの拡張された領域内のサブピクセルサンプル位置を取得するように構成された第2の取得ユニット1420と、参照ピクチャ内の整数ピクセルサンプル位置を取得するために、取得されたサブピクセルサンプル位置を処理する、特に、丸めるように構成された処理ユニット1430とを含み、予測ユニット1440は、参照ピクチャ内の取得された整数ピクセルサンプル位置における参照サンプル値を参照することによって、双方向のオプティカルフロー予測を実行するように構成される。

## 【 0 2 3 7 】

取得ユニット1410および1420、処理ユニット1430、ならびに予測ユニット1440は、ハードウェア、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実装される可能性がある。ソフトウェアに実装される場合、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または通信媒体上で送信され、ハードウェアに基づく処理ユニットによって実行される可能性がある。命令は、1つまたは複数のデジタル信号プロセッサ(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)、または他の等価な集積もしくはディスクリート論理回路などの1つまたは複数のプロセッサによって実行される可能性がある。したがって、用語「プロセッサ」は、本明細書において使用されるとき、上述の構造または本明細書において説明された技術の実装に好適な任意の他の構造のいずれかを指す可能性がある。加えて、一部の態様において、本明細書において説明された機能は、符号化および復号のために構成された専用のハードウェアおよび/もしくはソフトウェアモジュール内に提供されるか、または組み合わされたコーデックに組み込まれる可能性がある。また、技術は、1つまたは複数の回路または論理要素にすべて実装される可能性がある。

## 【 0 2 3 8 】

いくつかの特定の实施形態が、本開示の以下の態様において概説される。

## 【 0 2 3 9 】

第1の態様によれば、現在のコーディングブロックの双方向のオプティカルフロー予測において使用される予測されたブロックの拡張された領域に属するサンプル位置に関する予測されたサンプル値を取得するために使用される方法が提供され、方法は、現在のコーディングブロックのMVを取得するステップと、MVと、現在のコーディングブロックのサ

ンプル位置を取り囲む現在のピクチャ内の拡張されたサンプル位置とに基づいて、参照ピクチャ内のサブピクセル位置を取得するステップと、取得されたサブピクセルサンプル位置を最も近い整数ピクセル位置(nearest integer pixel local)に向けて丸める/再配置することによって、参照ピクチャ内の整数ピクセル位置を取得するステップと、予測されたブロックの拡張された領域に属するサンプル位置に関する予測されたサンプル値を、取得された整数ピクセル位置における参照サンプル値として設定するステップとを含む。

**【0240】**

取得されたサブピクセルサンプル位置を最も近い整数ピクセル位置に向けて丸める/再配置するステップは、取得されたサブピクセルサンプル位置を、水平および垂直方向において別々に、その方向における最も近い整数サンプル位置に丸めるステップを含む可能性があり、2つの整数サンプル位置の間の半サンプル位置は、その方向においてより高い整数サンプル位置に丸められる。

10

**【0241】**

取得されたサブピクセルサンプル位置を最も近い整数ピクセル位置に向けて丸める/再配置するステップは、取得されたサブピクセルサンプル位置を、水平および垂直方向において別々に、閾値と比較するステップを含む可能性があり、サブピクセルサンプル位置が閾値以上の場合、最も近いより高い整数サンプル位置が、整数ピクセル位置として取得される可能性があり、サブピクセルサンプル位置が閾値よりも小さい場合、最も近いより低い整数サンプル位置が、整数ピクセル位置として取得される可能性がある。

**【0242】**

第2の態様によれば、現在のコーディングブロックの双方向のオプティカルフロー予測において使用される予測されたブロックの拡張された領域に属するサンプル位置に関する予測されたサンプル値を取得するために使用される方法が提供され、方法は、コーディングブロックの拡張された領域内の予測されたサンプルに対応する第1のMVを取得するステップと、第1のMVの小数部分を破棄することによって、第2のMVを取得するステップと、第2のMVに基づいて、参照ピクチャ内の拡張された領域の第1の整数ピクセル位置を取得するステップと、第1のMVに基づいて、小数オフセットベクトルを決定するステップと、第1の整数ピクセル位置と小数オフセットベクトルとに基づいて、参照ピクチャ内の第2の整数ピクセル位置を取得するステップとを含み、第2の整数ピクセル位置が、第1の整数ピクセル位置に位置オフセットを加えたものに設定され、位置オフセットが、小数オフセットベクトルの各成分と閾値との間の比較によって決定され、方法は、参照ピクチャ内の第2の整数ピクセル位置におけるピクセル値を使用することによって、予測されたサンプルをパディングするステップをさらに含む。

20

30

**【0243】**

位置オフセットの成分が1であるか0であるかは、小数オフセットベクトルの対応する成分と閾値との間の比較の結果に依存する可能性があり、成分は、X成分とY成分とを含む。

**【0244】**

閾値は、Kである可能性があり、位置オフセットの成分は、小数オフセットの対応する成分がK以上である場合、1に設定され、そうでない場合、0に設定される。Kは、8に等しい可能性がある。

40

**【0245】**

さらなる態様によれば、上で説明された第1および第2の態様による方法のいずれか1つを実行するための処理回路を含む、エンコーダまたはデコーダが提供される。

**【0246】**

さらなる態様によれば、上で説明された第1および第2の態様による方法のいずれか1つを実行するためのプログラムコードを含む、コンピュータプログラム製品が提供される。

**【0247】**

さらなる態様によれば、1つまたは複数のプロセッサと、プロセッサに結合され、プロセッサによって実行するためのプログラミングを記憶する非一時的コンピュータ可読記憶媒体とを含む、デコーダまたはエンコーダが提供され、プログラミングは、プロセッサに

50

よって実行されるときに、上で説明された第1および第2の態様による方法のいずれか1つを行うようにデコーダを構成する。

【0248】

#### 数学演算子

本出願において使用される数学演算子は、Cプログラミング言語において使用される数学演算子に似ている。しかし、整数の除算および算術シフト演算の結果は、より厳密に定義され、累乗および実数値の除算などの追加の演算が、定義される。付番およびカウントの規則は、概して0から始まり、つまり、「第1」は、0番と等価であり、「第2」は、1番と等価であり、以下同様である。

【0249】

#### 算術演算子

以下の算術演算子が、以下の通り定義される。

【0250】

10

20

30

40

50

【表 1】

+	加算	
-	減算(2引数の演算子として)または否定(単項前置演算子として)	
*	行列の乗算を含む、乗算	
$x^y$	累乗。xのy乗を規定する。他の文脈では、このような表記は、累乗として解釈されることが意図されない上付き文字のために使用される。	10
/	結果をゼロへ向かって切り捨てする整数の除算。たとえば、7/4および-7/-4は、1に切り捨てられ、-7/4および7/-4は、-1に切り捨てられる。	20
÷	切り捨てまたは丸めが意図されない数学的方程式において除算を表すために使用される。	
$\frac{x}{y}$	切り捨てまたは丸めが意図されない数学的方程式において除算を表すために使用される。	
$\sum_{i=x}^y f(i)$	iがxからyを含んでyまでのすべての整数値を取るf(i)の総和。	30
$x \% y$	法。x ≥ 0およびy > 0である整数xおよびyについてのみ定義される、xをyで割った余り。	

## 【0251】

## 論理演算子

以下の論理演算子が、以下の通り定義される。

$x \ \&\& \ y$  xおよびyのブール論理「積」

$x \ || \ y$  xおよびyのブール論理「和」

! ブール論理「否定」

$x \ ? \ y \ : \ z$  xが真であるかまたは0に等しくない場合、値yと評価され、そうでない場合、値zと評価される。

## 【0252】

## 関係演算子

以下の関係演算子が、以下の通り定義される。

より大きい

10

20

30

40

50

= 以上  
 未満  
 = 以下  
 == 等しい  
 != 等しくない

## 【0253】

関係演算子が値「na」(該当なし)を割り振られたシンタックス要素または変数に適用されるとき、値「na」は、シンタックス要素または変数に関する異なる値として扱われる。値「na」は、いかなる他の値とも等しくないとみなされる。

## 【0254】

## ビット演算子

以下のビット演算子が、以下の通り定義される。

& ビット毎の「論理積」。整数引数に対する演算のとき、整数値の2の補数表現に対して作用する。別の引数よりも少ないビットを含む2進数引数に対する演算のとき、より短い引数が、0に等しいさらに上位桁のビットを追加することによって拡張される。

| ビット毎の「論理和」。整数引数に対する演算のとき、整数値の2の補数表現に対して作用する。別の引数よりも少ないビットを含む2進数引数に対する演算のとき、より短い引数が、0に等しいさらに上位桁のビットを追加することによって拡張される。

^ ビット毎の「排他的論理和」。整数引数に対する演算のとき、整数値の2の補数表現に対して作用する。別の引数よりも少ないビットを含む2進数引数に対する演算のとき、より短い引数が、0に等しいさらに上位桁のビットを追加することによって拡張される。

x y xの2の補数による整数の表現の、2進数のy桁分の算術右シフト。この関数は、yの非負の整数値に対してのみ定義される。右シフトの結果として最上位ビット(MSB)にシフトされるビットは、シフト演算の前のxのMSBに等しい値を有する。

x y xの2の補数による整数の表現の、2進数のy桁分の算術左シフト。この関数は、yの非負の整数値に対してのみ定義される。左シフトの結果として最下位ビット(LSB)にシフトされるビットは、0に等しい値を有する。

## 【0255】

## 代入演算子

以下の算術演算子が、以下の通り定義される。

= 代入演算子

++ インクリメント、つまり、x++は、 $x = x + 1$ と等価であり、配列のインデックスに使用されるとき、インクリメント演算の前に変数の値と評価される。

-- デクリメント、つまり、x--は、 $x = x - 1$ と等価であり、配列のインデックスに使用されるとき、デクリメント演算の前に変数の値と評価される。

+= 指定された量のインクリメント、つまり、 $x += 3$ は、 $x = x + 3$ と等価であり、 $x += (-3)$ は、 $x = x + (-3)$ と等価である。

-= 指定された量のデクリメント、つまり、 $x -= 3$ は、 $x = x - 3$ と等価であり、 $x -= (-3)$ は、 $x = x - (-3)$ と等価である。

## 【0256】

## 範囲の表記

以下の表記が、値の範囲を指定するために使用される。

x = y..z xは、x、y、およびzが整数値であり、zがyよりも大きいものとして、yおよびzを含んでyからzまでの整数値を取る。

## 【0257】

## 数学関数

以下の数学関数が、定義される。

## 【0258】

## 【数9】

10

20

30

40

50

$$\text{Abs}(x) = \begin{cases} x & ; x \geq 0 \\ -x & ; x < 0 \end{cases}$$

## 【 0 2 5 9 】

Asin( x ) -1.0および1.0を含んで-1.0から1.0までの範囲内の引数xに作用し、ラジアンを単位として-  $\frac{\pi}{2}$ および  $\frac{\pi}{2}$ を含んで-  $\frac{\pi}{2}$ から  $\frac{\pi}{2}$ までの範囲の出力値を有する三角法の逆正弦関数

Atan( x ) 引数xに作用し、ラジアンを単位として-  $\frac{\pi}{2}$ および  $\frac{\pi}{2}$ を含んで-  $\frac{\pi}{2}$ から  $\frac{\pi}{2}$ までの範囲の出力値を有する三角法の逆正接関数

## 【 0 2 6 0 】

## 【数 1 0 】

$$\text{Atan2}(y, x) = \begin{cases} \text{Atan}\left(\frac{y}{x}\right) & ; x > 0 \\ \text{Atan}\left(\frac{y}{x}\right) + \pi & ; x < 0 \ \&\& \ y \geq 0 \\ \text{Atan}\left(\frac{y}{x}\right) - \pi & ; x < 0 \ \&\& \ y < 0 \\ +\frac{\pi}{2} & ; x == 0 \ \&\& \ y \geq 0 \\ -\frac{\pi}{2} & ; \text{それ以外の場合} \end{cases}$$

10

20

## 【 0 2 6 1 】

Ceil( x ) x以上の最小の整数。

Clip1<sub>y</sub>( x ) = Clip3( 0, ( 1 - BitDepth<sub>y</sub> ) - 1, x )

Clip1<sub>c</sub>( x ) = Clip3( 0, ( 1 - BitDepth<sub>c</sub> ) - 1, x )

## 【 0 2 6 2 】

## 【数 1 1 】

$$\text{Clip3}(x, y, z) = \begin{cases} x & ; z < x \\ y & ; z > y \\ z & ; \text{それ以外の場合} \end{cases}$$

30

## 【 0 2 6 3 】

Cos( x ) ラジアンを単位とする引数xに作用する三角法の余弦関数。

Floor(x) x以下の最大の整数。

## 【 0 2 6 4 】

## 【数 1 2 】

$$\text{GetCurrMsb}(a, b, c, d) = \begin{cases} c + d & ; b - a \geq d/2 \\ c - d & ; a - b > d/2 \\ c & ; \text{それ以外の場合} \end{cases}$$

40

## 【 0 2 6 5 】

Ln( x ) xの自然対数(eを底とする対数であり、eは、自然対数の底の定数2.718281828...である)。

50

$\text{Log}_2(x)$   $x$ の2を底とする対数。

$\text{Log}_{10}(x)$   $x$ の10を底とする対数。

【0266】

【数13】

$$\text{Min}(x, y) = \begin{cases} x & ; x \leq y \\ y & ; x > y \end{cases}$$

$$\text{Max}(x, y) = \begin{cases} x & ; x \geq y \\ y & ; x < y \end{cases}$$

10

【0267】

$\text{Round}(x) = \text{Sign}(x) * \text{Floor}(\text{Abs}(x) + 0.5)$

【0268】

【数14】

$$\text{Sign}(x) = \begin{cases} 1 & ; x > 0 \\ 0 & ; x == 0 \\ -1 & ; x < 0 \end{cases}$$

20

【0269】

$\text{Sin}(x)$  ラジアンを単位とする引数 $x$ に作用する三角法の正弦関数

【0270】

【数15】

$$\text{Sqrt}(x) = \sqrt{x}$$

30

$$\text{Swap}(x, y) = (y, x)$$

【0271】

$\text{Tan}(x)$  ラジアンを単位とする引数 $x$ に作用する三角法の正接関数

【0272】

演算の優先順位

式中の優先順位が括弧を使用して明示されないとき、以下の規則が、適用される。

- より高い優先度の演算は、より低い優先度のいかなる演算よりも前に評価される。
- 同じ優先度の演算は、左から右に順に評価される。

40

【0273】

下の表は、最も高い方から最も低い方へ演算の優先度を明示し、表のより上の位置は、より高い優先度を示す。

【0274】

Cプログラミング言語においても使用される演算子に関して、本明細書において使用される優先順位は、Cプログラミング言語において使用されるのと同じである。

【0275】

50

【表 2】

表: (表の一番上の)最も高い方から(表の一番下の)最も低い方への演算の優先度

(オペランド $x$ 、 $y$ および $z$ での) 演算	
" $x++$ ", " $x--$ "	
"! $x$ ", " $-x$ " (単項前置演算子として)	10
$x^y$	
" $x * y$ ", " $x / y$ ", " $x \div y$ ", " $\frac{x}{y}$ ", " $x \% y$ "	
" $x + y$ ", " $x - y$ " (2 引数の演算子として), " $\sum_{i=x}^y f(i)$ "	20
" $x \ll y$ ", " $x \gg y$ "	
" $x < y$ ", " $x \leq y$ ", " $x > y$ ", " $x \geq y$ "	
" $x == y$ ", " $x != y$ "	
" $x \& y$ "	
" $x   y$ "	
" $x \&\& y$ "	30
" $x    y$ "	
" $x ? y : z$ "	
" $x..y$ "	
" $x = y$ ", " $x += y$ ", " $x -= y$ "	

10

20

30

40

【 0 2 7 6 】

論理演算のテキストの記述

本文中、以下の形態で、すなわち、

if( 条件0 )

    ステートメント0

else if( 条件1 )

    ステートメント1

...

else /\* 残りの条件に関する情報を伝えるコメント \*/

50

ステートメントn

の形態で数学的に記述される論理演算のステートメントは、以下のように記述される可能性がある。

以下のように... / ...以下が適用される。

- 条件0の場合、ステートメント0
- そうではなく、条件1の場合、ステートメント1
- ...
- それ以外の場合(残りの条件に関する情報を伝えるコメント)、ステートメントn

【0277】

本文中のそれぞれの「...の場合、...、そうではなく...の場合、...、それ以外の場合、...」のステートメントは、「...の場合、...」が直後に続く「以下のように...」または「...以下が適用される」によって導入される。「...の場合、...、そうではなく...の場合、...、それ以外の場合、...」の最後の条件は、常に「それ以外の場合、...」である可能性がある。交互に挿入された「...の場合、...、そうではなく...の場合、...、それ以外の場合、...」のステートメントは、「以下のように...」または「...以下が適用される」を終わりの「それ以外の場合、...」とマッチングすることによって特定され得る。

10

【0278】

本文中、以下の形態で、すなわち、

if( 条件0a && 条件0b )

ステートメント0

else if( 条件1a || 条件1b )

ステートメント1

...

else

ステートメントn

の形態で数学的に記述される論理演算のステートメントは、以下のように記述される可能性がある。

以下のように... / ...以下が適用される。

- 以下の条件のすべてが真である場合、ステートメント0
  - 条件0a
  - 条件0b
- そうでなく、以下の条件のうちの1つまたは複数が真である場合、ステートメント1
  - 条件1a
  - 条件1b
- ...
- それ以外の場合、ステートメントn

30

【0279】

本文中、以下の形態で、すなわち、

if( 条件0 )

ステートメント0

if( 条件1 )

ステートメント1

の形態で数学的に記述される論理演算のステートメントは、以下のように記述される可能性がある。

条件0のとき、ステートメント0

条件1のとき、ステートメント1

【0280】

本開示の実施形態が主にビデオコーディングに基づいて説明されたが、コーディングシステム10、エンコーダ20、およびデコーダ30(およびそれに対応してシステム10)の実施形態、ならびに本明細書において説明された他の実施形態は、静止ピクチャの処理または

50

コーディング、つまり、ビデオコーディングと同様のいかなる先行するまたは連続するピクチャからも独立した個々のピクチャの処理またはコーディングのために構成される可能性もあることに留意されたい。概して、ピクチャの処理コーディングが単一のピクチャ17に制限される場合、インター予測ユニット244(エンコーダ)および344(デコーダ)のみが、利用可能でない可能性がある。ビデオエンコーダ20およびビデオデコーダ30のすべての他の機能(ツールまたはテクノロジーとも称される)、たとえば、残差計算204/304、変換206、量子化208、逆量子化210/310、(逆)変換212/312、区分け262、イントラ予測254/354、および/またはループフィルタ220、320、およびエントロピーコーディング270、およびエントロピー復号304が、静止ピクチャの処理のために等しく使用される可能性がある。

10

#### 【0281】

たとえば、エンコーダ20およびデコーダ30、ならびにたとえばエンコーダ20およびデコーダ30に関連して本明細書において説明された機能の実施形態は、ハードウェア、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実装される可能性がある。ソフトウェアに実装される場合、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または通信媒体上で送信され、ハードウェアに基づく処理ユニットによって実行される可能性がある。コンピュータ可読媒体は、データ記憶媒体などの有形の媒体に対応するコンピュータ可読記憶媒体、またはたとえば通信プロトコルによる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容易にする任意の媒体を含む通信媒体を含む可能性がある。このようにして、概して、コンピュータ可読媒体は、(1)非一時的である有形のコンピュータ可読記憶媒体または(2)信号もしくは搬送波などの通信媒体に対応する可能性がある。データ記憶媒体は、本開示において説明された技術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つもしくは複数のコンピュータまたは1つもしく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る可能性がある。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む可能性がある。

20

#### 【0282】

限定ではなく例として、そのようなコンピュータ可読記憶媒体は、RAM、ROM、EEPROM、CD-ROMもしくは他の光ディスクストレージ、磁気ディスクストレージもしくは他の磁気記憶装置、フラッシュメモリ、または命令もしくはデータ構造の形態で所望のプログラムコードを記憶するために使用されることが可能であり、コンピュータによってアクセスされることが可能である任意の他の媒体を含み得る。また、任意の接続が、適切にコンピュータ可読媒体と称される。たとえば、命令が、同軸ケーブル、光ファイバケーブル、ツイストペア、デジタル加入者線(DSL)、または赤外線、ラジオ波、およびマイクロ波などのワイヤレステクノロジーを用いてウェブサイト、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、光ファイバケーブル、ツイストペア、DSL、または赤外線、ラジオ波、およびマイクロ波などのワイヤレステクノロジーは、媒体の定義に含まれる。しかし、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、または他の一時的媒体を含まず、その代わりに、非一時的な有形の記憶媒体を対象とすることを理解されたい。本明細書において使用されるとき、ディスク(disk)およびディスク(disc)は、コンパクトディスク(CD: compact disc)、レーザディスク(laser disc)、光ディスク(optical disc)、デジタルバーサタイルディスク(DVD: digital versatile disc)、フロッピーディスク(floppy disk)、およびブルーレイディスク(登録商標) disc)を含み、ディスク(disk)が、通常、磁気的にデータを再生する一方、ディスク(disc)は、レーザを用いて光学的にデータを再生する。上記のものの組合せも、コンピュータ可読媒体の範囲に含まれるべきである。

30

40

#### 【0283】

命令は、1つまたは複数のデジタル信号プロセッサ(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)、または他の等価な集積もしくはディスクリート論理回路などの1つまたは複数のプロセッサによ

50

って実行される可能性がある。したがって、用語「プロセッサ」は、本明細書において使用されるとき、上述の構造または本明細書において説明された技術の実装に好適な任意の他の構造のいずれかを指す可能性がある。加えて、一部の態様において、本明細書において説明された機能は、符号化および復号のために構成された専用のハードウェアおよび/もしくはソフトウェアモジュール内に提供されるか、または組み合わされたコーデックに組み込まれる可能性がある。また、技術は、1つまたは複数の回路または論理要素にすべて実装される可能性がある。

【0284】

本開示の技術は、ワイヤレスハンドセット、集積回路(IC)、または1組のIC(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実装される可能性がある。様々な構成要素、モジュール、またはユニットが、開示された技術を実行するように構成されたデバイスの機能の態様を強調するために本開示において説明されているが、異なるハードウェアユニットによる実現を必ずしも必要としない。むしろ、上述のように、様々なユニットが、コーデックハードウェアユニットにおいて組み合わされるか、または好適なソフトウェアおよび/もしくはファームウェアと連携した、上述の1つもしくは複数のプロセッサを含む相互運用性のあるハードウェアユニットの集合によって提供される可能性がある。

【符号の説明】

【0285】

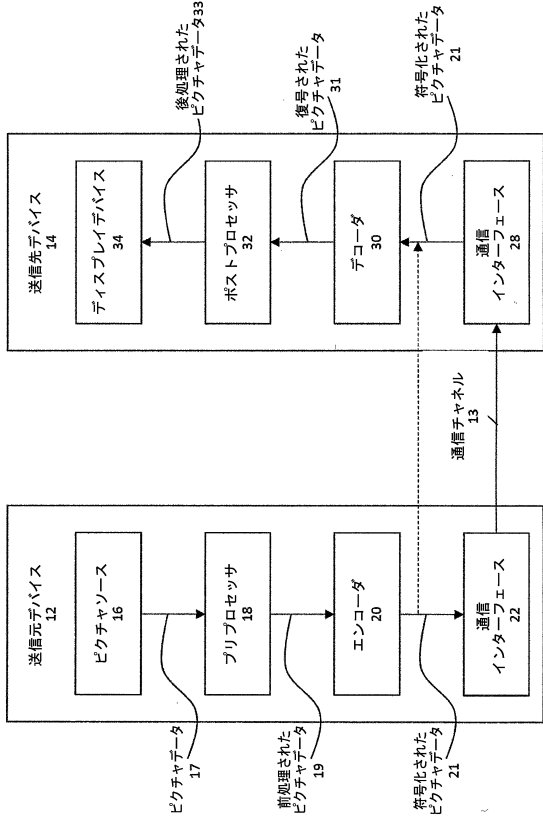
- |     |  |    |
|-----|--|----|
| 10  | ビデオコーディングシステム、コーディングシステム                             |    |
| 12  | 送信元デバイス  | 20 |
| 13  | 符号化されたピクチャデータ、通信チャネル                                 |    |
| 14  | 送信先デバイス  |    |
| 16  | ピクチャソース  |    |
| 17  | ピクチャ、ピクチャデータ、生ピクチャ、生ピクチャデータ、モノクロピクチャ、カラーピクチャ、現在のピクチャ |    |
| 18  | プリプロセッサ、前処理ユニット、ピクチャプリプロセッサ                          |    |
| 19  | 前処理されたピクチャ、前処理されたピクチャデータ                             |    |
| 20  | ビデオエンコーダ、エンコーダ、符号化装置                                 |    |
| 21  | 符号化されたピクチャデータ、符号化されたビットストリーム                         |    |
| 22  | 通信インターフェース、通信ユニット                                    | 30 |
| 28  | 通信インターフェース、通信ユニット                                    |    |
| 30  | デコーダ、ビデオデコーダ、復号装置                                    |    |
| 31  | 復号されたピクチャデータ、復号されたピクチャ                               |    |
| 32  | ポストプロセッサ、後処理ユニット                                     |    |
| 33  | 後処理されたピクチャデータ、後処理されたピクチャ                             |    |
| 34  | ディスプレイデバイス   |    |
| 40  | ビデオコーディングシステム  |    |
| 41  | イメージングデバイス   |    |
| 42  | アンテナ   |    |
| 43  | プロセッサ  | 40 |
| 44  | メモリストア   |    |
| 45  | ディスプレイデバイス   |    |
| 46  | 処理回路   |    |
| 201 | 入力、入力インターフェース  |    |
| 203 | ピクチャブロック、元のブロック、現在のブロック、区分けされたブロック、現在のピクチャブロック       |    |
| 204 | 残差計算ユニット、残差計算  |    |
| 205 | 残差ブロック、残差  |    |
| 206 | 変換処理ユニット、変換  |    |
| 207 | 変換係数   | 50 |

208	量子化ユニット、量子化	
209	量子化された係数、量子化された変換係数、量子化された残差係数	
210	逆量子化ユニット、逆量子化	
211	逆量子化された係数、逆量子化された残差係数	
212	逆変換処理ユニット、(逆)変換	
213	再構築された残差ブロック、逆量子化された係数、変換ブロック	
214	再構築ユニット、加算器、合算器	
215	再構築されたブロック	
220	ループフィルタユニット、ループフィルタ	
221	フィルタリングされたブロック、フィルタリングされた再構築されたブロック	10
230	復号ピクチャバッファ(DPB)	
231	復号されたピクチャ	
244	インター予測ユニット	
254	イントラ予測ユニット、イントラ予測	
260	モード選択ユニット	
262	区分けユニット、区分け	
265	予測ブロック、予測子	
266	シンタックス要素	
270	エントロピー符号化ユニット、エントロピーコーディング	
272	出力、出力インターフェース	20
304	エントロピー復号ユニット、残差計算、エントロピー復号	
309	量子化された係数	
310	逆量子化ユニット、逆量子化	
311	逆量子化された係数、変換係数	
312	逆変換処理ユニット、(逆)変換、出力	
313	再構築された残差ブロック	
314	再構築ユニット、合算器、加算器	
315	再構築されたブロック	
320	ループフィルタ、ループフィルタユニット	
321	フィルタリングされたブロック、復号されたビデオブロック	30
330	復号ピクチャバッファ(DPB)	
331	復号されたピクチャ	
344	インター予測ユニット	
354	イントラ予測ユニット、イントラ予測	
360	モード適用ユニット	
365	予測ブロック	
400	ビデオコーディングデバイス	
410	着信ポート、入力ポート	
420	受信機ユニット(Rx)	
430	プロセッサ、論理ユニット、中央演算処理装置(CPU)	40
440	送信機ユニット(Tx)	
450	発信ポート、出力ポート	
460	メモリ	
470	コーディングモジュール	
500	装置	
502	プロセッサ	
504	メモリ	
506	データ	
508	オペレーティングシステム	
510	アプリケーションプログラム	50

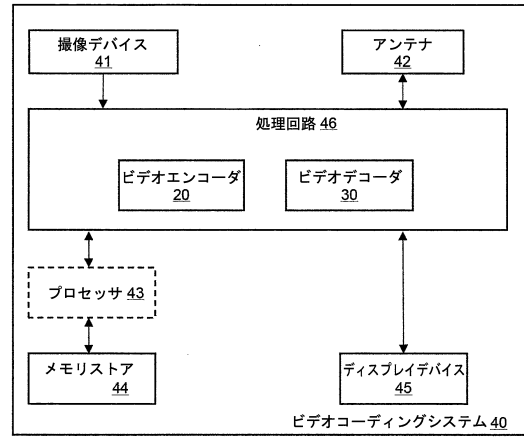
- 512 バス
- 518 ディスプレイ
- 1410 第1の取得ユニット、取得ユニット
- 1420 第2の取得ユニット、取得ユニット
- 1430 処理ユニット
- 1440 予測ユニット

【図面】

【図1A】



【図1B】



10

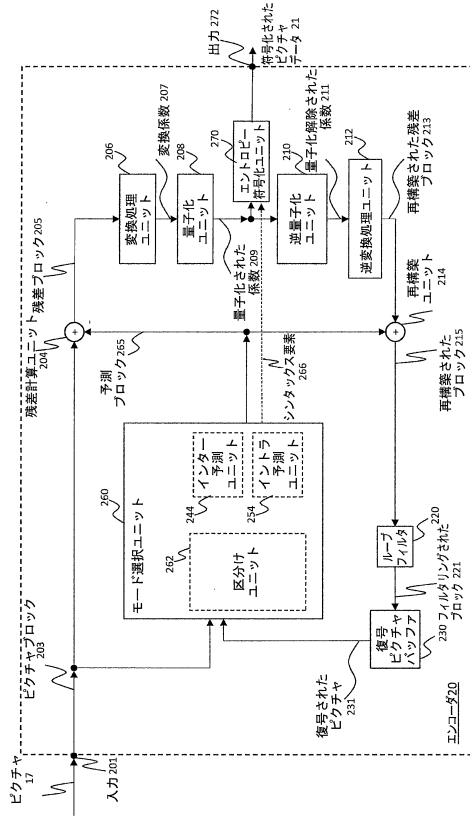
20

30

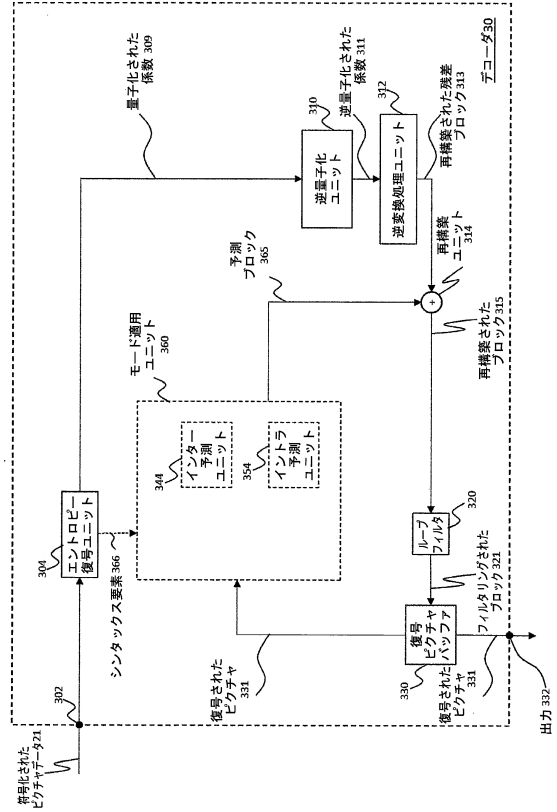
40

50

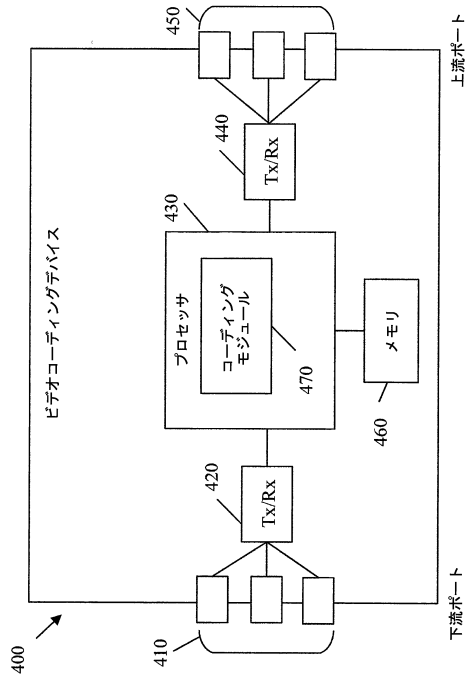
【図 2】



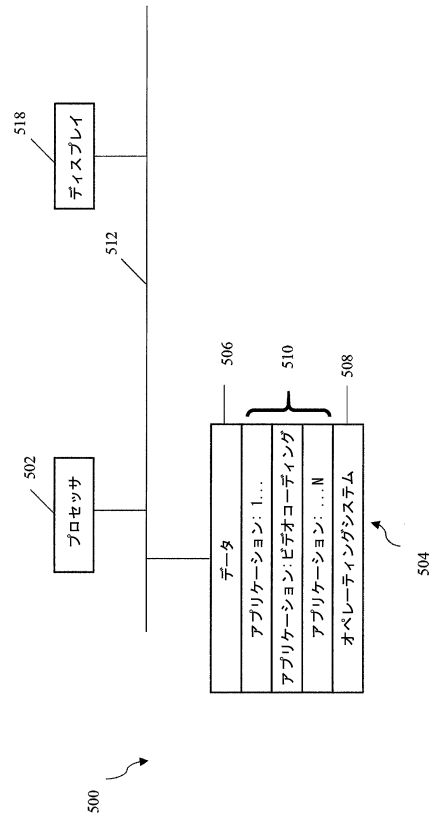
【図 3】



【図 4】



【図 5】



10

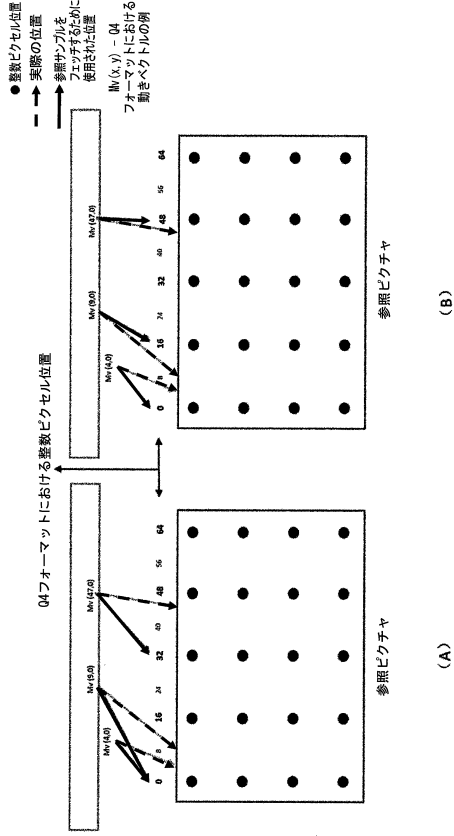
20

30

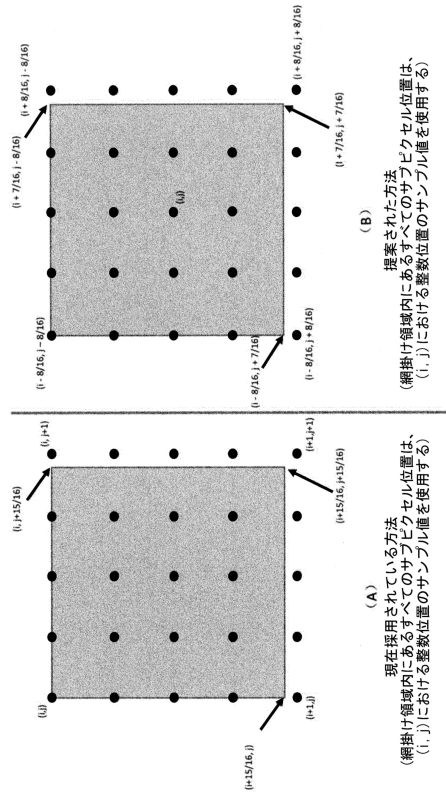
40

50

【図 6】



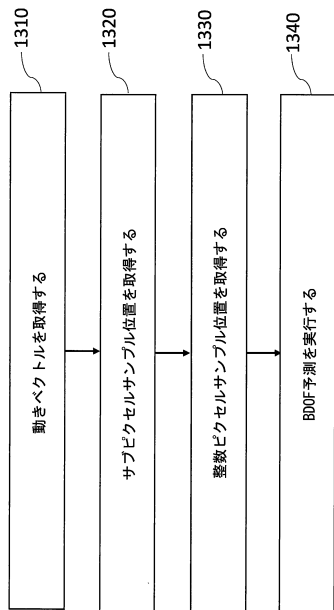
【図 7】



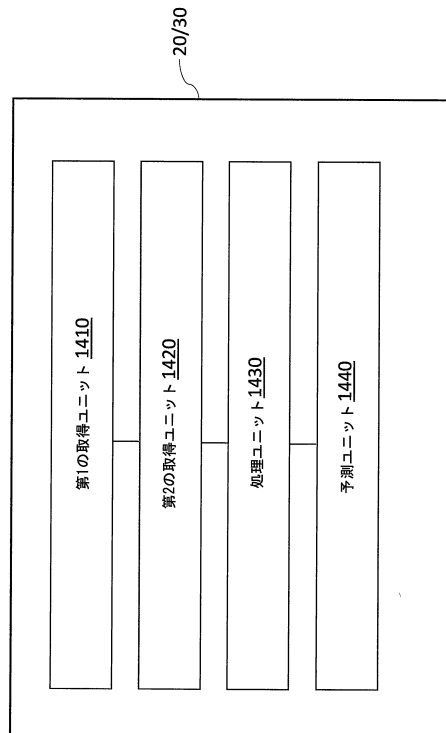
10

20

【図 8】



【図 9】



30

40

50

## フロントページの続き

- ブロック・アイ・クロス・1エー  
(72)発明者 ジーヴァ・ラジ・エー  
インド・637408・タミル・ナドゥ・ラシプラム・オーピーピー・トゥー・イー・ビー・コロニー・ラマサミー・トットム・10/17
- (72)発明者 サーガル・コテチャ  
インド・431126・マハーラーシュトラ・ビード・オーピーピー・トゥー・アウテ・マンガル・カリヤラヤ・ダクシュ・レジデンシー・フラット・ナンバー・101
- 審査官 富樫 明  
(56)参考文献 米国特許出願公開第2013/0272410 (US, A1)  
特表2013-543318 (JP, A)  
Sriram Sethuraman, CE9-related: A SIMD-friendly simplification for BDOF, JVET-N0312-v2, 2019年03月20日
- (58)調査した分野 (Int.Cl., DB名)  
H04N 19/00 - 19/98