

(19) 日本国特許庁 (JP)

(12) 公表特許公報 (A)

(11) 特許出願公表番号

特表2017-513332

(P2017-513332A)

(43) 公表日 平成29年5月25日 (2017.5.25)

(51) Int.Cl.	F I	テーマコード (参考)
HO 4 N 19/433 (2014.01)	HO 4 N 19/433	5 C 1 5 9
HO 4 N 19/593 (2014.01)	HO 4 N 19/593	
HO 4 N 19/70 (2014.01)	HO 4 N 19/70	

審査請求 未請求 予備審査請求 未請求 (全 56 頁)

(21) 出願番号	特願2016-557255 (P2016-557255)	(71) 出願人	507364838
(86) (22) 出願日	平成27年3月20日 (2015. 3. 20)		クアルコム, インコーポレイテッド
(85) 翻訳文提出日	平成28年9月14日 (2016. 9. 14)		アメリカ合衆国 カリフォルニア 921
(86) 国際出願番号	PCT/US2015/021866		21 サン ディエゴ モアハウス ドラ
(87) 国際公開番号	W02015/143395		イブ 5775
(87) 国際公開日	平成27年9月24日 (2015. 9. 24)	(74) 代理人	100108453
(31) 優先権主張番号	61/969, 022		弁理士 村山 靖彦
(32) 優先日	平成26年3月21日 (2014. 3. 21)	(74) 代理人	100163522
(33) 優先権主張国	米国 (US)		弁理士 黒田 晋平
(31) 優先権主張番号	62/000, 437	(72) 発明者	シアン・リ
(32) 優先日	平成26年5月19日 (2014. 5. 19)		アメリカ合衆国・カリフォルニア・921
(33) 優先権主張国	米国 (US)		21-1714・サン・ディエゴ・モアハ
(31) 優先権主張番号	14/663, 155		ウス・ドライブ・5775
(32) 優先日	平成27年3月19日 (2015. 3. 19)		
(33) 優先権主張国	米国 (US)		

最終頁に続く

(54) 【発明の名称】 ビデオコーディングの参照として現在のピクチャを使用すること

(57) 【要約】

ビデオデータを符号化または復号するための例示的な方法は、ビデオコードによって、参照ピクチャバッファ内に、ビデオデータの現在のピクチャのバージョンを記憶するステップと、現在のピクチャを予測するために使用される参照ピクチャリスト (RPL) 内に現在のピクチャを含めるステップと、ビデオコードによって、RPLに基づいて、参照ピクチャバッファ内に記憶された現在のピクチャのバージョン内に含まれるビデオデータの予測子ブロックに基づいて、現在のピクチャ内のビデオデータのブロックをコーディングするステップとを含む。

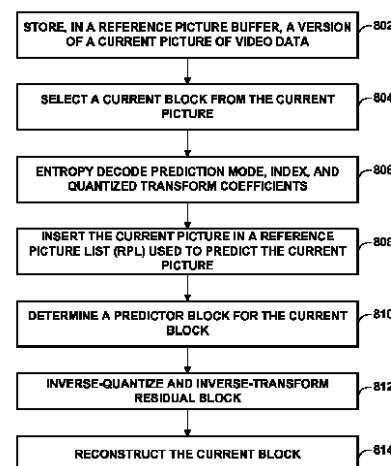


FIG. 8

【特許請求の範囲】**【請求項 1】**

ビデオデータを符号化または復号する方法であって、前記方法が、
ビデオコードによって、参照ピクチャバッファ内に、前記ビデオデータの現在のピクチャのバージョンを記憶するステップと、
前記現在のピクチャのブロックを予測する際に使用される参照ピクチャリスト(RPL)内に前記現在のピクチャの指標を挿入するステップと、
前記ビデオコードによって、前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記バージョン内に含まれるビデオデータの予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第1のブロックをコーディングするステップと
を備える方法。

10

【請求項 2】

前記ビデオコードによって、前記第1のブロックをコーディングした後に、前記参照ピクチャバッファ内の前記現在のピクチャの前記バージョンを、前記コーディングされた第1のブロックを含む前記現在のピクチャの更新されたバージョンで更新するステップと、
前記ビデオコードによって、前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記更新されたバージョン内に含まれる予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第2のブロックをコーディングするステップと
をさらに備える、請求項1に記載の方法。

20

【請求項 3】

前記ビデオコードによって、前記ビデオデータのピクチャが前記ピクチャ自体を予測するために使用されるRPL内に存在し得るのかどうかを示す構文要素をコーディングするステップと、
前記構文要素に基づいて、前記ビデオデータの前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に含めることを決定するステップと
をさらに備える、請求項1に記載の方法。

【請求項 4】

前記構文要素をコーディングするステップが、前記現在のピクチャによって参照されるビデオパラメータセット(VPS)、前記現在のピクチャによって参照されるシーケンスパラメータセット(SPS)、または前記現在のピクチャによって参照されるピクチャパラメータセット(PPS)内に前記構文要素をコーディングするステップを備える、請求項3に記載の方法。

30

【請求項 5】

前記方法が、前記第1のブロックがイントラブロックコピー(イントラBC)を使用してコーディングされるかどうかを示す構文要素をコーディングするステップを含まない、請求項4に記載の方法。

【請求項 6】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記スライスのための並置参照インデックスを示す構文要素が前記現在のピクチャではないピクチャを示す、請求項1に記載の方法。

40

【請求項 7】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記予測子ブロックが前記現在のスライス内に含まれる、請求項1に記載の方法。

【請求項 8】

前記RPL内の各エントリがインデックス値を有し、前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に挿入するステップが、少なくとも、
長期RPS内のピクチャのインデックス値よりも小さいインデックス値を用いて前記RPL内に前記現在のピクチャを挿入するか、

50

長期RPS内のピクチャのインデックス値よりも大きいインデックス値を用いて前記RPL内に前記現在のピクチャを挿入するか、

または固定されたインデックス値を用いて前記RPL内に前記現在のピクチャを挿入することによって、

1つまたは複数の参照ピクチャセット(RPS)に基づいて前記現在のピクチャを予測するために使用される前記RPLを構成するステップを備える、請求項1に記載の方法。

【請求項9】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスがイントラスライスであり、前記現在のピクチャを、前記現在のピクチャを予測するために使用されるRPL内に挿入するステップが、前記現在のピクチャを予測するために使用される前記RPLが前記現在のピクチャのみを含むように、前記現在のピクチャを予測するために使用される前記RPLを構成するステップを備える、請求項1に記載の方法。

10

【請求項10】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスがインタースライスであり、前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に挿入するステップが、前記現在のピクチャを予測するために使用される前記RPLが、前記現在のピクチャと、ビデオデータの1つまたは複数の他のピクチャとを含むように、前記現在のピクチャを予測するために使用される前記RPLを構成するステップを備える、請求項1に記載の方法。

20

【請求項11】

前記ビデオコードによって、前記現在のピクチャの前記ブロックをコーディングする前に、前記ビデオデータの前記現在のピクチャを長期参照ピクチャとしてマーキングするステップと、

前記ビデオデータによって、前記現在のピクチャの前記ブロックをコーディングした後に、前記ビデオデータの前記現在のピクチャを短期参照ピクチャとしてマーキングするステップと

をさらに備える、請求項1に記載の方法。

【請求項12】

前記ブロックをコーディングするステップが前記ブロックを符号化するステップを備え、前記方法が、符号化ビデオビットストリーム内に、ビデオデータの前記ブロックとビデオデータの前記予測子ブロックとの間の変位を表すベクトルの表現を符号化するステップをさらに備える、請求項1に記載の方法。

30

【請求項13】

前記ブロックをコーディングするステップが前記ブロックを復号するステップを備え、前記方法が、符号化ビデオビットストリームに基づいて、ビデオデータの前記ブロックとビデオデータの前記予測子ブロックとの間の変位を表すベクトルを決定するステップをさらに備える、請求項1に記載の方法。

【請求項14】

ビデオデータを符号化または復号するためのデバイスであって、前記デバイスが、

前記ビデオデータの1つまたは複数のピクチャを記憶するように構成された参照ピクチャバッファと、

40

1つまたは複数のプロセッサとを備え、前記1つまたは複数のプロセッサが、

参照ピクチャバッファ内に、前記ビデオデータの現在のピクチャのバージョンを記憶し、

前記現在のピクチャのブロックを予測する際に使用される参照ピクチャリスト(RPL)内に前記現在のピクチャの指標を挿入し、

ビデオコードによって、前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記バージョン内に含まれるビデオデータの予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第1のブロックをコーディングするように構成された、デバイス。

50

【請求項 15】

前記1つまたは複数のプロセッサが、

前記第1のブロックをコーディングした後に、前記参照ピクチャバッファ内の前記現在のピクチャの前記バージョンを、前記コーディングされた第1のブロックを含む前記現在のピクチャの更新されたバージョンで更新し、

前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記更新されたバージョン内に含まれる予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第2のブロックをコーディングするようにさらに構成された、請求項14に記載のデバイス。

【請求項 16】

10

前記1つまたは複数のプロセッサが、

前記ビデオデータのピクチャが前記ピクチャ自体を予測するために使用されるRPL内に存在し得るのかどうかを示す構文要素をコーディングし、

前記構文要素に基づいて、前記ビデオデータの前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に含めることを決定するようにさらに構成された、請求項14に記載のデバイス。

【請求項 17】

前記1つまたは複数のプロセッサが、前記現在のピクチャによって参照されるビデオパラメータセット(VPS)、前記現在のピクチャによって参照されるシーケンスパラメータセット(SPS)、または前記現在のピクチャによって参照されるピクチャパラメータセット(PPS)内に前記構文要素をコーディングするようにさらに構成された、請求項16に記載のデバイス。

20

【請求項 18】

前記1つまたは複数のプロセッサが、前記第1のブロックがイントラブロックコピー(イントラBC)を使用してコーディングされるかどうかを示す構文要素をコーディングするように構成されていない、請求項17に記載のデバイス。

【請求項 19】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスのための並置参照インデックスを示す構文要素が前記現在のピクチャではないピクチャを示す、請求項14に記載のデバイス。

30

【請求項 20】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記予測子ブロックが前記現在のスライス内に含まれる、請求項14に記載のデバイス。

【請求項 21】

前記RPL内の各エントリがインデックス値を有し、前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に挿入するために、前記1つまたは複数のプロセッサが、少なくとも、

長期RPS内のピクチャのインデックス値よりも小さいインデックス値を用いて前記RPL内に前記現在のピクチャを挿入するか、

長期RPS内のピクチャのインデックス値よりも大きいインデックス値を用いて前記RPL内に前記現在のピクチャを挿入するか、

40

または固定されたインデックス値を用いて前記RPL内に前記現在のピクチャを挿入することによって、

1つまたは複数の参照ピクチャセット(RPS)に基づいて前記現在のピクチャを予測するために使用される前記RPLを構成するように構成された、請求項14に記載のデバイス。

【請求項 22】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスがイントラスライスであり、前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に挿入するために、前記1つまたは複数のプロセッサが、前記現在のピクチャを予測するために使用される前記RPLが前記現在のピクチャのみを含むように

50

、前記現在のピクチャを予測するために使用される前記RPLを構成するように構成された、請求項14に記載のデバイス。

【請求項 2 3】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスがインタースライスであり、前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に挿入するために、前記1つまたは複数のプロセッサが、前記現在のピクチャを予測するために使用される前記RPLが、前記現在のピクチャと、ビデオデータの1つまたは複数の他のピクチャとを含むように、前記現在のピクチャを予測するために使用される前記RPLを構成するように構成された、請求項14に記載のデバイス。

【請求項 2 4】

前記1つまたは複数のプロセッサが、
前記現在のピクチャの前記ブロックをコーディングする前に、前記ビデオデータの前記現在のピクチャを長期参照ピクチャとしてマーキングし、
前記現在のピクチャの前記ブロックをコーディングした後に、前記ビデオデータの前記現在のピクチャを短期参照ピクチャとしてマーキングするようにさらに構成された、請求項14に記載のデバイス。

【請求項 2 5】

前記ビデオコーダがビデオエンコーダであり、前記1つまたは複数のプロセッサが、符号化ビデオビットストリーム内に、ビデオデータの前記ブロックとビデオデータの前記予測子ブロックとの間の変位を表すベクトルの表現を符号化するようにさらに構成された、請求項14に記載のデバイス。

【請求項 2 6】

前記ビデオコーダがビデオデコーダであり、前記1つまたは複数のプロセッサが、符号化ビデオビットストリームに基づいて、ビデオデータの前記ブロックとビデオデータの前記予測子ブロックとの間の変位を表すベクトルを決定するようにさらに構成された、請求項14に記載のデバイス。

【請求項 2 7】

ビデオデータを符号化または復号するためのデバイスであって、
参照ピクチャバッファ内に、前記ビデオデータの現在のピクチャのバージョンを記憶するための手段と、
前記現在のピクチャのブロックを予測する際に使用される参照ピクチャリスト(RPL)内に前記現在のピクチャの指標を挿入するための手段と、
前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記バージョン内に含まれるビデオデータの予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第1のブロックをコーディングするための手段と
を備えるデバイス。

【請求項 2 8】

前記ビデオデータのピクチャがそれ自体を予測するために使用されるRPL内に存在し得るかどうかを示す構文要素をコーディングするための手段と、
前記構文要素に基づいて、前記ビデオデータの前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に含めるかどうかを決定するための手段と
をさらに備える、請求項27に記載のデバイス。

【請求項 2 9】

実行されたとき、ビデオコーダの1つまたは複数のプロセッサに、
参照ピクチャバッファ内に、前記ビデオデータの現在のピクチャのバージョンを記憶させ、
前記現在のピクチャのブロックを予測する際に使用される参照ピクチャリスト(RPL)内に前記現在のピクチャの指標を挿入させ、
前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記バージョン内に含まれるビデオデータの予測子ブロックに基づいて、前記現在のピク

10

20

30

40

50

チャ内のビデオデータの第1のブロックをコーディングさせる命令を記憶するコンピュータ可読記憶媒体。

【請求項30】

実行されたとき、ビデオコードの前記1つまたは複数のプロセッサに、

前記ビデオデータのピクチャがそれ自体を予測するために使用されるRPL内に存在し得るのかどうかを示す構文要素をコーディングさせ、

前記構文要素に基づいて、前記ビデオデータの前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に含めるかどうかを決定させる命令をさらに記憶する、請求項29に記載のコンピュータ可読記憶媒体。

【発明の詳細な説明】

10

【技術分野】

【0001】

本出願は、2014年3月21日に出願した、米国仮出願第61/969,022号、および、2014年5月19日に出願した、米国仮出願第62/000,437号の利益を主張するものであり、これらの各々の全内容は、参照によりその全体が本明細書に組み込まれている。

【0002】

本開示は、ビデオコーディングに関する。

【背景技術】

【0003】

デジタルビデオ機能は、デジタルテレビジョン、デジタル直接放送システム、ワイヤレス放送システム、携帯情報端末(PDA)、ラップトップまたはデスクトップコンピュータ、タブレットコンピュータ、電子書籍リーダー、デジタルカメラ、デジタル記録デバイス、デジタルメディアプレーヤ、ビデオゲームデバイス、ビデオゲームコンソール、セルラーまたは衛星無線電話、いわゆる「スマートフォン」ビデオ会議デバイス、ビデオストリーミングデバイスなどを含む、広範囲のデバイスに組み込まれることが可能である。デジタルビデオデバイスは、MPEG-2、MPEG-4、ITU-T H.263、ITU-T H.264/MPEG-4 Part 10、高度ビデオコーディング(AVC:Advanced Video Coding)、現在開発中の高効率ビデオコーディング(HEVC:High Efficiency Video Coding)規格、およびそのような規格の拡張によって定義される規格に記載のものなどの、ビデオ圧縮技法を実装する。ビデオデバイスは、そのようなビデオ圧縮技法を実装することによって、デジタルビデオ情報をより効率的に送信、受信、符号化、復号、および/または記憶することがある。

20

30

【0004】

ビデオ圧縮技法は、ビデオシーケンスに固有の冗長性を低減または除去するために、空間(ピクチャ内)予測および/または時間(ピクチャ間)予測を実行する。ブロックベースのビデオコーディングに関して、ビデオスライス(すなわち、ビデオピクチャ、またはビデオピクチャの一部)は、ビデオブロックに分割されることがあり、ビデオブロックはまた、ツリーブロック、コーディング単位(CU:coding unit)、および/またはコーディングノードと呼ばれることもある。ピクチャのイントラコーディング(I)スライス内のビデオブロックは、同じピクチャ内の隣接ブロック内の参照サンプルに対する空間予測を用いて符号化される。ピクチャのインターコーディング(PまたはB)スライス内のビデオブロックは、同じピクチャ内の隣接ブロック内の参照サンプルに対する空間予測、または他の参照ピクチャ内の参照サンプルに対する時間予測を使用してもよい。

40

【0005】

空間または時間予測は、コーディングされるべきブロックのための予測ブロックをもたず。残差データは、コーディングされるべき元のブロックと予測ブロックとの間のピクセル差を表す。インターコーディングブロックは、予測ブロックを形成する参照サンプルのブロックを指す動きベクトルと、コーディングブロックと予測ブロックとの間の差を示す残差データとに従って符号化される。イントラコーディングブロックは、イントラコーディングモードと残差データとに従って符号化される。さらなる圧縮のために、残差データは、ピクセル領域から変換領域に変換されてもよく、次いで量子化されてもよい残差変

50

換係数をもたらす。

【先行技術文献】

【非特許文献】

【0006】

【非特許文献1】Pangら、「Non-RCE3: Intra Motion Compensation with 2-D MVs」、Document: JCTVC-N0256、JCT-VC of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11、第14回ミーティング:ウィーン、オーストリア、2013年7月25日～8月2日

【非特許文献2】Rapakaら、「On parallel processing capability of intra block copy」、Document: JCTVC-S0220、JCT-VC of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11、第19回ミーティング:ストラスブール、フランス、2014年10月17～24日

10

【発明の概要】

【課題を解決するための手段】

【0007】

全体的には、本開示は、ビデオコーディングのためのイントラ予測を実行するための技法について説明する。より具体的には、本開示は、現在のピクチャの1つまたは複数のブロックをコーディングするとき、参照ピクチャとして現在のピクチャを使用するための例示的な技法について説明する。たとえば、現在のピクチャは、イントラブロックコピーイング(イントラBC:Block Copying)を使用して現在のピクチャの1つまたは複数のブロックをコーディングするとき、参照ピクチャとして使用されてもよい。

【0008】

20

一例では、ビデオデータを符号化または復号する方法は、ビデオコードによって、参照ピクチャバッファ内に、ビデオデータの現在のピクチャのバージョンを記憶するステップと、現在のピクチャのブロックを予測する際に使用される参照ピクチャリスト(RPL:reference picture list)内に現在のピクチャの指標を挿入するステップと、ビデオコードによって、RPLに基づいて、参照ピクチャバッファ内に記憶された現在のピクチャのバージョン内に含まれるビデオデータの予測子ブロックに基づいて、現在のピクチャ内のビデオデータの第1のブロックをコーディングするステップとを含む。いくつかの例では、予測子ブロックは、代替的に予測ブロックと呼ばれることもある。

【0009】

別の例では、ビデオデータを符号化または復号するためのデバイスは、ビデオデータの1つまたは複数のピクチャを記憶するように構成された参照ピクチャバッファと、1つまたは複数のプロセッサとを含む。この例では、1つまたは複数のプロセッサは、参照ピクチャバッファ内に、ビデオデータの現在のピクチャのバージョンを記憶し、現在のピクチャのブロックを予測する際に使用される参照ピクチャリスト(RPL)内に現在のピクチャの指標を挿入し、ビデオコードによって、RPLに基づいて、参照ピクチャバッファ内に記憶された現在のピクチャのバージョン内に含まれるビデオデータの予測子ブロックに基づいて、現在のピクチャ内のビデオデータの第1のブロックをコーディングするように構成される。

30

【0010】

別の例では、ビデオデータを符号化または復号するためのデバイスは、参照ピクチャバッファ内に、ビデオデータの現在のピクチャのバージョンを記憶するための手段と、現在のピクチャのブロックを予測する際に使用される参照ピクチャリスト(RPL)内に現在のピクチャの指標を挿入するための手段と、RPLに基づいて、参照ピクチャバッファ内に記憶された現在のピクチャのバージョン内に含まれるビデオデータの予測子ブロックに基づいて、現在のピクチャ内のビデオデータの第1のブロックをコーディングするための手段とを含む。

40

【0011】

別の例では、コンピュータ可読記憶媒体は、実行されたとき、ビデオコードの1つまたは複数のプロセッサに、参照ピクチャバッファ内に、ビデオデータの現在のピクチャのバージョンを記憶させ、現在のピクチャのブロックを予測する際に使用される参照ピクチャ

50

リスト(RPL)内に現在のピクチャの指標を挿入させ、RPLに基づいて、参照ピクチャバッファ内に記憶された現在のピクチャのバージョン内に含まれるビデオデータの予測子ブロックに基づいて、現在のピクチャ内のビデオデータの第1のブロックをコーディングさせる命令を記憶する。

【0012】

本開示の1つまたは複数の態様の詳細を添付の図面および以下の説明に記載する。本開示で説明する技法の他の特徴、目的、および利点は、これらの説明および図面、ならびに特許請求の範囲から明らかになる。

【図面の簡単な説明】

【0013】

【図1】本開示の技法を実装することができる例示的なビデオ符号化および復号システムを示すブロック図である。

【図2】本開示の1つまたは複数の技法による、ピクチャの例示的なビデオシーケンスを示す概念図である。

【図3】本開示に記載のイントラブロックコピーのための技法を使用することができるビデオエンコーダの一例を示すブロック図である。

【図4】本開示の1つまたは複数の技法に従って使用され得る角度イントラ予測モードの一例を示す図である。

【図5】本開示に記載の技法を実装することができるビデオデコーダ30の一例を示すブロック図である。

【図6】本開示の1つまたは複数の技法による、イントラブロックコピーイングプロセスの一例を示す図である。

【図7】本開示の1つまたは複数の技法による、同じピクチャ内に含まれる予測子ブロックに基づいてピクチャのビデオデータのブロックを符号化するビデオエンコーダの例示的な動作を示すフロー図である。

【図8】本開示の1つまたは複数の技法による、同じピクチャ内に含まれる予測子ブロックに基づいてピクチャのビデオデータのブロックを復号するビデオデコーダの例示的な動作を示すフロー図である。

【発明を実施するための形態】

【0014】

ビデオシーケンスは、一般に、ピクチャのシーケンスとして表される。典型的には、ブロックベースのコーディング技法は、個々のピクチャの各々をコーディングするために使用される。すなわち、各ピクチャは、ブロックに分割され、各々のブロックは、個別にコーディングされる。ビデオデータのブロックをコーディングすることは、一般に、ブロック内のピクセルのための予測値を形成することと、残差値をコーディングすることを含む。予測値は、1つまたは複数の予測ブロック内のピクセルサンプルを使用して形成される。残差値は、元のブロックのピクセルと予測ピクセル値との間の差を表す。具体的には、ビデオデータの元のブロックは、ピクセル値のアレイを含み、予測ブロックは、予測ピクセル値のアレイを含む。残差値は、元のブロックのピクセル値と予測ピクセル値との間のピクセルごとの差を表す。

【0015】

ビデオデータのブロックのための予測技法は、一般に、イントラ予測およびインター予測に分類される。イントラ予測または空間予測は、任意の参照ピクチャからの予測を含まず、代わりにブロックは、隣接する以前にコーディングされたブロックのピクセル値から予測される。インター予測または時間予測は、一般に、1つまたは複数の参照ピクチャリスト(RPL)から選択された1つまたは複数の以前にコーディングされた参照ピクチャ(たとえば、フレームまたはスライス)のピクセル値からブロックを予測することを含む。ビデオコーダは、RPLに含まれるピクチャを記憶するように構成された1つまたは複数の参照ピクチャバッファを含んでもよい。

【0016】

リモートデスクトップ、リモートゲーム、ワイヤレスディスプレイ、車載インフォテインメント、クラウドコンピューティングなどの多くの用途は、日々の生活の中で日常的になってきている。これらの用途でのビデオコンテンツは、通常、自然なコンテンツ、テキスト、人工グラフィックスなどの組合せである。テキストおよび人工グラフィックス領域では、繰り返されるパターン(文字、アイコン、シンボルなど)がしばしば存在する。イントラブロックコピーイング(イントラBC)は、ビデオコードがそのような冗長性を除去し、イントラピクチャコーディング効率を改善することを可能にし得る技法である。いくつかの例では、イントラBCは、代替的に、イントラ動き補償(MC:motion compensation)と呼ばれることもある。

【0017】

いくつかのイントラBC技法によれば、ビデオコードは、現在のブロックの予測のための同じピクチャ内のビデオデータの(コーディングされるべき)現在のブロックの直上のまたは直接水平方向に並んだ、ビデオデータの現在のブロックと同じピクチャ内の以前にコーディングされたビデオデータのブロックを使用することができる。言い換えれば、ビデオデータのピクチャが2-Dグリッドに押し付けられた場合、ビデオデータの各ブロックは、x値およびy値の固有の範囲を占めることになる。したがって、いくつかのビデオコードは、x値の同じセット(すなわち、現在のブロックと垂直方向に並んだ)またはy値の同じセットのみを共有する(すなわち、現在のブロックと水平方向に並んだ)、以前にコーディングされたビデオデータのブロックに基づいて、ビデオデータの現在のブロックを予測することができる。

【0018】

他のイントラBC技法は、Pangら、「Non-RCE3: Intra Motion Compensation with 2-D MVs」、Document: JCTVC-N0256、JCT-VC of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11、第14回ミーティング:ウィーン、オーストリア、2013年7月25日~8月2日(以後、「JCTVC-N0256」)に記載されている。ウィーンでのJCT-VCミーティング(2013年7月)では、イントラBCは、高効率ビデオコーディング(HEVC)範囲拡張規格で採用された。JCTVC-N0256によれば、ビデオコードは、現在のブロックと同じピクチャ内の予測ブロックを識別する二次元動きベクトルを決定することができる。いくつかの例では、動きベクトルはまた、ブロックベクトル、オフセットベクトル、または変位ベクトルと呼ばれることもある。いずれの場合でも、二次元動きベクトルは、水平変位成分と垂直変位成分とを有し、これらの各々は、ゼロまたは非ゼロであり得る。水平変位成分は、ビデオデータの予測ブロック、または予測ブロックと、ビデオデータの現在のブロックとの間の水平変位を表し、垂直変位成分は、ビデオデータの予測ブロックと、ビデオデータの現在のブロックとの間の垂直変位を表す。イントラBCに関して、予測ブロックのピクセルは、コーディングされているブロック(すなわち、現在のブロック)内の対応するピクセルのための予測サンプルとして使用され得る。ビデオコードは、加えて、ビデオデータの現在のブロックおよび予測ブロックに基づいて、ビデオデータの残差ブロックを決定することができ、二次元動きベクトルとビデオデータの残差ブロックとをコーディングすることができる。

【0019】

いくつかの例では、イントラBCは、特に画面コンテンツコーディングのための効率的なコーディングツールである。たとえば、いくつかの例では、イントラBCを使用してブロックをコーディングすることは、インターまたはイントラコーディングを使用してブロックをコーディングするよりも小さいビットストリームをもたらすことができる。上記で説明したように、イントラBCは、インターに似たコーディングツール(ピクチャのピクセル値がピクチャ内の他のピクセル値から予測されることを意味する)である。しかしながら、いくつかの例では、実用的な設計では好ましくない可能性がある、イントラBCに適用される1つまたは複数の制約のため、イントラBCを従来のイントラピクチャに統合することは困難である可能性がある。いくつかの例示的な制約は、限定はしないが、予測ブロックが、コーディングされるべき現在のブロックとして、同じスライスまたはタイル内にあること、予測子ブロックがコーディングされるべき現在のブロックと重複しないこと、予測子

ブロック内のすべてのピクセルが再構成されること、(たとえば、Rapakaら、「On parallel processing capability of intra block copy」、Document: JCTVC-S0220、JCT-VC of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11、第19回ミーティング: ストラスブル、フランス、2014年10月17~24日に記載のように並列化実装に関する考慮事項のために) 予測子ブロックが特定の領域内にあること、および、制約されたイントラ予測が有効になっているとき、予測子ブロックが従来のインターモードを使用してコーディングされたいかなるピクセルをも含まないことを含む。加えて、いくつかの例では、従来のイントラおよびインターフレームのためのハードウェアアーキテクチャは、(たとえば、イントラBCがピクチャ内部のブロックコピーをもたらすため) 変更なしでイントラBSに再利用することはできない可能性がある。そのように、イントラBCに現在適用されている制約の一部またはすべてを維持しながら、ハードウェアアーキテクチャへの(重大な)変更なしに、ビデオコーダがイントラBCによって提供される効率を増すことができるようにすることが望ましい可能性がある。

【0020】

本開示の1つまたは複数の技法によれば、従来のイントラ予測技法を使用して現在のピクチャ内のサンプルに基づいて現在のピクチャのブロックを予測することとは対照的に、ビデオコーダは、従来のインター予測と同様の技法を使用して現在のピクチャ内のサンプルに基づいて現在のピクチャ内のブロックを予測するためにイントラBCを実行することができる。たとえば、ビデオコーダは、現在のピクチャを予測するために使用される参照ピクチャリスト(RPL)内に現在のピクチャを含み、参照ピクチャバッファ内に現在のピクチャのバージョンを記憶し、参照ピクチャバッファ内に記憶された現在のピクチャのバージョン内に含まれるビデオデータの予測子ブロックに基づいて、現在のピクチャ内のビデオデータのブロックをコーディングすることができる。このようにして、ビデオコーダは、イントラBCに現在適用されている制約の一部またはすべてを維持しながら、イントラBCによって提供される効率を増すことができる。また、このようにして、ビデオコーダは、重大な変更なしに、イントラBCのために従来のイントラおよびインターフレームのためのハードウェアアーキテクチャを再利用することができる。

【0021】

本開示は、現在のピクチャの一部を予測するとき、参照ピクチャとして現在のピクチャを利用することに関連する例示的な技法について説明する。理解を支援するために、例示的な技法は、4:4:4および4:2:2を含む、できる限り高いビット深度の(8ビットよりも高い)高彩度サンプリングフォーマットのサポートを含む、高効率ビデオコーディング(HEVC)ビデオコーディング規格への範囲拡張(RExt: range extension)に関して説明される。技法はまた、画面コンテンツコーディングのために適用可能であり得る。技法は、範囲拡張または画面コンテンツコーディングに限定されず、一般的に、規格ベースまたは非規格ベースのビデオコーディングを含むビデオコーディング技法に適用可能であり得ることが理解されるべきである。また、本開示に記載の技法は、将来的に開発される規格の一部になり得る。言い換えれば、本開示に記載の技法は、以前に開発されたビデオコーディング規格、現在開発中のビデオコーディング規格、および今後のビデオコーディング規格に適用可能であり得る。

【0022】

図1は、本開示の技法を実装することができる例示的なビデオ符号化および復号システム10を示すブロック図である。図1に示すように、システム10は、宛先デバイス14によって後で復号されるべき符号化ビデオデータを提供するソースデバイス12を含む。具体的には、ソースデバイス12は、コンピュータ可読媒体16を介して宛先デバイス14にビデオデータを提供する。ソースデバイス12および宛先デバイス14は、デスクトップコンピュータ、ノートブック(すなわち、ラップトップ)コンピュータ、タブレットコンピュータ、セットトップボックス、いわゆる「スマート」フォンなどの電話ハンドセット、いわゆる「スマート」パッド、テレビジョン、カメラ、表示デバイス、デジタルメディアプレーヤ、ビデオゲームコンソール、ビデオストリーミングデバイス、などを含む、広い範囲のデバイス

のうちのいずれかを含んでもよい。いくつかの場合には、ソースデバイス12および宛先デバイス14は、ワイヤレス通信のために装備されてもよい。

【0023】

宛先デバイス14は、コンピュータ可読媒体16を介して、復号されるべき符号化ビデオデータを受信してもよい。コンピュータ可読媒体16は、ソースデバイス12から宛先デバイス14に符号化ビデオデータを移動させることができる任意のタイプの媒体またはデバイスを備えてもよい。一例では、コンピュータ可読媒体16は、ソースデバイス12がリアルタイムで宛先デバイス14に符号化ビデオデータを直接送信することができるようにする通信媒体を備えてもよい。符号化ビデオデータは、ワイヤレス通信プロトコルなどの通信規格に従って変調され、宛先デバイス14に送信されてもよい。通信媒体は、無線周波数(RF)スペクトルなどの任意のワイヤレスもしくはワイヤード通信媒体、または、1つもしくは複数の物理的伝送線を備えてもよい。通信媒体は、ローカルエリアネットワーク、ワイドエリアネットワーク、またはインターネットなどのグローバルネットワークなどの、パケットベースのネットワークの一部を形成してもよい。通信媒体は、ルータ、スイッチ、基地局、または、ソースデバイス12から宛先デバイス14への通信を容易にするために有用であってもよい任意の他の機器を含んでもよい。

10

【0024】

いくつかの例では、符号化データは、ソースデバイス12の出力インターフェース22から記憶デバイス32に出力されてもよい。同様に、符号化データは、宛先デバイス14の入力インターフェース28によって記憶デバイス32からアクセスされてもよい。記憶デバイス32は、ハードドライブ、Blu-ray（登録商標）ディスク、DVD、CD-ROM、フラッシュメモリ、揮発性もしくは不揮発性メモリ、または符号化ビデオデータを記憶するための任意の他の適切なデジタル記憶媒体などの、様々な分散されたまたはローカルにアクセスされるデータ記憶媒体のいずれかを含んでもよい。さらなる例では、記憶デバイス32は、ソースデバイス12によって生成された符号化ビデオを記憶することができるファイルサーバまたは別の中間記憶デバイスに対応してもよい。

20

【0025】

宛先デバイス14は、ストリーミングまたはダウンロードを介して記憶デバイス32から記憶されたビデオデータにアクセスしてもよい。ファイルサーバは、符号化ビデオデータを記憶し、宛先デバイス14にその符号化ビデオデータを送信することが可能な任意のタイプのサーバであってもよい。例示的なファイルサーバは、(たとえば、ウェブサイトのための)ウェブサーバ、FTPサーバ、ネットワーク接続ストレージ(NAS: network attached storage)デバイス、またはローカルディスクドライブを含む。宛先デバイス14は、インターネット接続を含む任意の標準的なデータ接続を介して符号化ビデオデータにアクセスしてもよい。これは、ファイルサーバに記憶された符号化ビデオデータにアクセスするのに適した、ワイヤレスチャネル(たとえば、Wi-Fi接続)、ワイヤード接続(たとえば、DSL、ケーブルモデムなど)、または両方の組合せを含んでもよい。記憶デバイスからの符号化ビデオデータの送信は、ストリーミング送信、ダウンロード送信、またはそれらの組合せであってもよい。

30

【0026】

本開示の技法は、必ずしもワイヤレス用途または設定に限定されない。技法は、無線(over-the-air)テレビジョン放送、ケーブルテレビジョン送信、衛星テレビジョン送信、インターネットストリーミングビデオ送信、動的適応型HTTPストリーミング(DASH: dynamic adaptive streaming over HTTP)、データ記憶媒体上で符号化されたデジタルビデオ、データ記憶媒体上に記憶されたデジタルビデオの復号、または他の用途などの、様々なマルチメディア用途のいずれかを支持してビデオコーディングに適用されてもよい。いくつかの例では、システム10は、ビデオストリーミング、ビデオ再生、ビデオ放送、および/またはビデオ電話などの用途をサポートするために、一方向または双方向ビデオ送信をサポートするように構成されてもよい。

40

【0027】

50

図1の例では、ソースデバイス12は、ビデオソース18と、ビデオエンコーダ20と、出力インターフェース22とを含む。宛先デバイス14は、入力インターフェース28と、ビデオデコーダ30と、表示デバイス31とを含む。本開示によれば、ソースデバイス12のビデオエンコーダ20は、ビデオコーディングにおける変換を実行するための技法を適用するように構成され得る。他の例では、ソースデバイスおよび宛先デバイスは、他の構成要素または配置を含んでもよい。たとえば、ソースデバイス12は、外部カメラなどの外部ビデオソース18からビデオデータを受信してもよい。同様に、宛先デバイス14は、一体化された表示デバイスを含むのではなく、外部表示デバイスとインターフェースしてもよい。

【0028】

図1の図示のシステム10は、単なる一例である。ビデオコーディングにおける改善されたイントラブロックコピー合図(signaling)のための技法は、任意のデジタルビデオ符号化および/または復号デバイスによって実行されてもよい。一般に、本開示の技法は、ビデオ符号化または復号デバイスによって実行されるが、技法はまた、ビデオコーデックによって実行されてもよい。さらに、本開示の技法はまた、ビデオプリプロセッサによって実行されてもよい。ソースデバイス12および宛先デバイス14は、単に、ソースデバイス12が宛先デバイス14に送信するためのコーディングされたビデオデータを生成するそのようなコーディングデバイスの例である。いくつかの例では、デバイス12、14は、デバイス12、14の各々がビデオ符号化および復号構成要素を含むように実質的に対称的な方法で動作してもよい。したがって、システム10は、たとえば、ビデオストリーミング、ビデオ再生、ビデオ放送、またはビデオ電話のための、ビデオデバイス12、14間の一方向または双方向ビデオ送信をサポートしてもよい。

【0029】

ビデオデバイス12のビデオソース18は、ビデオカメラ、以前にキャプチャされたビデオを含むビデオアーカイブ、および/またはビデオコンテンツプロバイダからビデオを受信するビデオ供給インターフェースなどの、ビデオキャプチャデバイスを含んでもよい。さらなる代替として、ビデオソース18は、ソースビデオとしてコンピュータグラフィックスベースのデータを、または、ライブビデオ、アーカイブされたビデオ、およびコンピュータ生成ビデオの組合せを生成してもよい。いくつかの場合では、ビデオソース18がビデオカメラである場合、ソースデバイス12および宛先デバイス14は、いわゆるカメラ電話またはビデオ電話を形成してもよい。上述したように、しかしながら、本開示に記載の技法は、一般に、ビデオコーディングに適用可能であり得、ワイヤレスおよび/またはワイヤード用途に適用されてもよい。各々の場合で、キャプチャされた、事前にキャプチャされた、またはコンピュータ生成されたビデオは、ビデオエンコーダ20によって符号化されてもよい。符号化ビデオ情報は、次いで、出力インターフェース22によって、コンピュータ可読媒体16上に出力されてもよい。

【0030】

コンピュータ可読媒体16は、ワイヤレス放送もしくはワイヤードネットワーク送信などの一時的媒体、または、ハードディスク、フラッシュドライブ、コンパクトディスク、デジタルビデオディスク、Blu-ray（登録商標）ディスク、もしくは他のコンピュータ可読媒体などの記憶媒体(すなわち、非一時的記憶媒体)を含んでもよい。いくつかの例では、ネットワークサーバ(図示せず)は、ソースデバイス12から符号化ビデオデータを受信してもよく、たとえば、ネットワーク送信を介して、宛先デバイス14に符号化ビデオデータを提供してもよい。同様に、ディスクスタンプ設備などの媒体製造設備のコンピューティングデバイスは、ソースデバイス12から符号化ビデオデータを受信してもよく、符号化ビデオデータを含むディスクを製造してもよい。したがって、コンピュータ可読媒体16は、様々な例において、様々な形態の1つまたは複数のコンピュータ可読媒体を含むと理解されてもよい。

【0031】

宛先デバイス14の入力インターフェース28は、コンピュータ可読媒体16または記憶デバイス32から情報を受信してもよい。コンピュータ可読媒体16または記憶デバイス32の情報

は、ビデオデコーダ30によっても使用される、ビデオエンコーダ20によって定義された構文情報を含んでもよく、構文情報は、ブロックおよび他のコーディング単位、たとえばGOPの特性および/または処理を記述する構文要素を含む。表示デバイス31は、復号ビデオデータをユーザに表示し、陰極線管(CRT)、液晶ディスプレイ(LCD)、プラズマディスプレイ、有機発光ダイオード(OLED)ディスプレイ、または別のタイプの表示デバイスなどの、様々な表示デバイスのいずれかを備えてもよい。

【0032】

ビデオエンコーダ20およびビデオデコーダ30は、該当する場合は、各々、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回路、ソフトウェア、ハードウェア、ファームウェア、またはそれらの組合せなどの、様々な適切なエンコーダまたはデコーダ回路のいずれかとして実装されてもよい。技法が部分的にソフトウェアにおいて実装されているとき、デバイスは、適切な非一時的コンピュータ可読媒体内にソフトウェアのための命令を記憶してもよく、本開示の技法を実行するために、1つまたは複数のプロセッサを使用してハードウェアにおいて命令を実行してもよい。ビデオエンコーダ20およびビデオデコーダ30の各々は、1つまたは複数のエンコーダまたはデコーダ内に含まれてもよく、それらのいずれかは、組み合わされたビデオエンコーダ/デコーダ(コーデック)の一部として組み込まれてもよい。ビデオエンコーダ20および/またはビデオデコーダ30を含むデバイスは、集積回路、マイクロプロセッサ、および/または、セルラー電話などのワイヤレス通信デバイスを備えてもよい。

【0033】

図1には示していないが、いくつかの態様では、ビデオエンコーダ20およびビデオデコーダ30は、各々、オーディオエンコーダおよびデコーダと一体化されてもよく、共通データストリームまたは別々のデータストリームにおけるオーディオとビデオの両方の符号化を処理するために、適切なMUX-DEMUXユニットまたは他のハードウェアおよびソフトウェアを含んでもよい。適用可能な場合、MUX-DEMUXユニットは、ITU H.223マルチプレクサプロトコル、または、ユーザデータグラムプロトコル(UDP:user datagram protocol)などの他のプロトコルに準拠してもよい。

【0034】

本開示は、一般に、ビデオデコーダ30などの別のデバイスに特定の情報を「合図する」ビデオエンコーダ20を参照することがある。しかしながら、ビデオエンコーダ20は、特定の構文要素をビデオデータの様々な符号化部分と関連付けることによって、情報を合図してもよいことが理解されるべきである。すなわち、ビデオエンコーダ20は、ビデオデータの様々な符号化部分のヘッダに特定の構文要素を格納することによって、データを「合図」してもよい。いくつかの場合では、そのような構文要素は、ビデオデコーダ30によって受信および復号される前に、符号化および記憶(たとえば、記憶デバイス32に記憶)されてもよい。したがって、「合図する」という用語は、一般に、そのような通信がリアルタイムもしくはほぼリアルタイムで、またはある時間の期間にわたって生じるかにかかわらず、符号化の時点で、媒体に構文要素を記憶するときに生じてもよいような、圧縮されたビデオデータを復号するための構文または他のデータの通信を指してもよく、構文要素は、次いで、この媒体に記憶された後の任意の時間に、復号デバイスによって取得されてもよい。

【0035】

ビデオエンコーダ20およびビデオデコーダ30は、HEVC規格などのビデオ圧縮規格に従って動作してもよい。本開示の技法は、どのような特定のコーディング規格にも限定されないが、技法は、HEVC規格と、特に、RExt拡張などのHEVC規格の拡張とに関連してもよい。HEVC規格化の努力は、HEVCテストモデル(HM)と呼ばれるビデオコーディングデバイスのモデルに基づいている。HMは、たとえば、ITU-T H.264/AVCによる既存のデバイスと比較してビデオコーディングデバイスのいくつかの追加の能力を想定する。たとえば、H.264は、9のイントラ予測符号化モードを提供するが、HMは、35ものイントラ予測符号化モード

を提供する

【0036】

一般に、HMの作業モデルは、ビデオピクチャが輝度サンプルと彩度サンプルの両方を含むツリーブロックまたは最大コーディング単位(LCU)のシーケンスに分割されてもよいことを記載している。ビットストリーム内の構文データは、ピクセル数の点から最大コーディング単位であるLCUのサイズを定義してもよい。スライスは、連続したコーディングツリー単位(CTU: coding tree unit)の数を含む。CTUの各々は、輝度サンプルのコーディングツリーブロックと、彩度サンプルの2つの対応するコーディングツリーブロックと、コーディングツリーブロックのサンプルをコーディングするために使用される構文構造とを備えてもよい。白黒ピクチャ、または3つの別個の色平面を有するピクチャでは、CTUは、単一のコーディングツリーブロックと、コーディングツリーブロックのサンプルをコーディングするために使用される構文構造とを備えてもよい。

10

【0037】

ビデオピクチャは、1つまたは複数のスライスに分割されてもよい。各ツリーブロックは、4分木に従ってコーディング単位(CU: coding unit)に分割されてもよい。一般に、4分木データ構造は、CUごとに1つのノードを含み、ルートノードは、ツリーブロックに対応する。CUが4つのサブCUに分割されたとき、CUに対応するノードは、4つのリーフノードを含み、各々のリーフノードは、サブCUのうちの1つに対応する。CUは、輝度サンプルアレイと、CbサンプルアレイおよびCrサンプルアレイと、コーディングブロックのサンプルをコーディングするために使用される構文構造とを有するピクチャの輝度サンプルのコーディングブロックと、彩度サンプルの2つの対応するコーディングブロックとを備えてもよい。白黒ピクチャ、または3つの別個の色平面を有するピクチャでは、CUは、単一のコーディングブロックと、コーディングブロックのサンプルをコーディングするために使用される構文構造とを備えてもよい。コーディングブロックは、サンプルの $N \times N$ ブロックである。

20

【0038】

4分木データ構造の各ノードは、対応するCUのための構文データを提供してもよい。たとえば、4分木内のノードは、ノードに対応するCUがサブCUに分割されているかどうかを示す分割フラグを含んでもよい。CUのための構文要素は、再帰的に定義されてもよく、CUがサブCUに分割されているかどうか依存してもよい。CUがさらに分割されていない場合、それは、リーフCUと呼ばれる。本開示では、リーフCUの4つのサブCUはまた、元のリーフCUの明示的な分割が存在しない場合でも、リーフCUと呼ばれることになる。たとえば、 16×16 サイズのCUがさらに分割されていない場合、 16×16 CUは、決して分割されていないが、4つの 8×8 サブCUはまた、リーフCUと呼ばれることになる。

30

【0039】

CUは、CUがサイズの区別を持たないことを除いて、H.264規格のマクロブロックと同様の目的を有する。たとえば、ツリーブロックは、4つの子ノード(サブCUとも呼ばれる)に分割されてもよく、各子ノードは、今度は、親ノードであってもよく、別の4つの子ノードに分割されてもよい。4分木のリーフノードと呼ばれる最後の分割されていない子ノードは、リーフCUとも呼ばれるコーディングノードを備える。コーディングされたビットストリームに関連付けられた構文データは、最大CU深度と呼ばれる、ツリーブロックが分割され得る最大回数を定義してもよく、また、コーディングノードの最小サイズを定義してもよい。したがって、ビットストリームはまた、最小コーディング単位(SCU)を定義してもよい。本開示は、HEVCの文脈におけるCU、PU、もしくはTUのいずれか、または、他の規格の文脈における同様のデータ構造(たとえば、H.264/AVCにおけるマクロブロックおよびそのサブブロック)を指すために、「ブロック」という用語を使用する。

40

【0040】

CUは、コーディングノードと、コーディングノードに関連付けられた予測単位(PU: prediction unit)および変換単位(TU: transform unit)とを含む。CUのサイズは、コーディングノードのサイズに対応し、形状において正方形でなければならない。CUのサイズは、8

50

×8ピクセルから、最大64×64ピクセルまたはそれ以上のツリーブロックのサイズまでの範囲であってもよい。各CUは、1つまたは複数のPUと1つまたは複数のTUとを含んでもよい。

【0041】

一般に、PUは、対応するCUのすべてまたは一部に対応する空間領域を表し、PUのための参照サンプルを取得するためのデータを含んでもよい。さらに、PUは、予測に関連するデータを含む。たとえば、PUがイントラモード符号化されているとき、PUのためのデータは、残差4分木(RQT:residual quadtree)内に含まれてもよく、PUに対応するTUのためのイントラ予測モードを記述するデータを含んでもよい。別の例として、PUがインターモード符号化されているとき、PUは、PUのための1つまたは複数の動きベクトルを定義するデータを含んでもよい。予測ブロックは、同じ予測が適用されるサンプルの長方形(すなわち、正方形または非正方形)ブロックであってもよい。CUのPUは、ピクチャの輝度サンプルの予測ブロックと、彩度サンプルの2つの対応する予測ブロックと、予測ブロックサンプルを予測するために使用される構文構造とを備えてもよい。白黒ピクチャ、または3つの別個の色平面を有するピクチャでは、PUは、単一の予測ブロックと、予測ブロックサンプルを予測するために使用される構文構造とを備えてもよい。

10

【0042】

TUは、変換、たとえば、離散コサイン変換(DCT)、整数変換、ウェーブレット変換、または残差ビデオデータへの概念的に類似の変換の適用に続く変換領域における係数を含んでもよい。残差データは、符号化されていないピクチャのピクセルと、PUに対応する予測値との間のピクセル差に対応してもよい。ビデオエンコーダ20は、CUのための残差データを含むTUを形成し、次いで、CUのための変換係数を生成するためにTUを変換してもよい。変換ブロックは、同じ変換が適用されるサンプルの長方形ブロックであってもよい。CUの変換単位(TU:transform unit)は、輝度サンプルの変換ブロックと、彩度サンプルの2つの対応する変換ブロックと、変換ブロックサンプルを変換するために使用される構文構造とを備えてもよい。白黒ピクチャ、または3つの別個の色平面を有するピクチャでは、TUは、単一の変換ブロックと、変換ブロックサンプルを変換するために使用される構文構造とを備えてもよい。

20

【0043】

変換に続いて、ビデオエンコーダ20は、変換係数の量子化を実行してもよい。量子化は、一般に、係数を表すために使用されるデータの量をできる限り低減するために変換係数が量子化され、さらなる圧縮を提供するプロセスを指す。量子化プロセスは、係数の一部またはすべてに関連付けられたビット深度を低減してもよい。たとえば、nビット値は、量子化の間にmビット値に切り捨てられてもよく、ここでnは、mよりも大きい。

30

【0044】

ビデオエンコーダ20は、量子化変換係数を含む2次元マトリクスから1次元ベクトルを生成して、変換係数を走査してもよい。走査は、アレイの前方により高いエネルギー(したがって、より低い周波数)係数を配置し、アレイの後方により低いエネルギー(したがって、より高い周波数)係数を配置するように設計されてもよい。いくつかの例では、ビデオエンコーダ20は、エントロピー符号化されることが可能なシリアル化ベクトルを生成するために、量子化変換係数を走査するために事前定義された走査順を利用してもよい。他の例では、ビデオエンコーダ20は、適応走査を実行してもよい。

40

【0045】

1次元ベクトルを形成するために、量子化変換係数を走査した後、ビデオエンコーダ20は、たとえば、コンテキスト適応型可変長コーディング(CAVLC:context-adaptive variable length coding)、コンテキスト適合型二値算術コーディング(CABAC:context-adaptive binary arithmetic coding)、構文ベースのコンテキスト適合型二値算術コーディング(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)コーディング、または別のエントロピー符号化方法に従って、1次元ベクトルをエントロピー符号化してもよい。ビデ

50

オエンコーダ20はまた、ビデオデータを復号する上でビデオデコーダ30によって使用する
ための符号化ビデオデータに関連付けられた構文要素をエントロピー符号化してもよい。

【0046】

ビデオエンコーダ20は、さらに、ブロックベースの構文データ、ピクチャベースの構文
データ、およびピクチャのグループ(GOP:group of pictures)ベースの構文データなどの
構文データを、たとえば、ピクチャヘッダ、ブロックヘッダ、スライスヘッダ、またはGO
Pヘッダ内で、ビデオデコーダ30に送信してもよい。GOP構文データは、それぞれのGOP内
のピクチャの数を記述してもよく、ピクチャ構文データは、対応するピクチャを符号化す
るために使用される符号化/予測モードを示してもよい。

【0047】

ビデオデコーダ30は、コーディングされたビデオデータを取得すると、ビデオエンコー
ダ20に関して説明した符号化パスと全体的に相互的な復号パスを実行してもよい。たとえ
ば、ビデオデコーダ30は、ビデオエンコーダ20から、符号化ビデオスライスのビデオブロ
ックと関連する構文要素とを表す符号化ビデオビットストリームを取得してもよい。ビデ
オデコーダ30は、ビットストリーム内に含まれたデータを使用して、元の符号化されてい
ないビデオシーケンスを再構成してもよい。

【0048】

ビデオエンコーダ20およびビデオデコーダ30は、ビデオスライス内のビデオブロックの
イントラおよびインターコーディングを実行してもよい。イントラコーディングは、所与
のビデオピクチャ内のビデオにおける空間的冗長性を低減または除去するために、空間予
測に依存する。インターコーディングは、ビデオシーケンスの隣接ピクチャ内のビデオに
おける時間的冗長性を低減もしくは除去するために、または、他のビューにおけるビデオ
で冗長性を低減もしくは除去するために、時間予測またはインタービュー予測に依存する
。イントラモード(Iモード)は、いくつかの空間ベースの圧縮モードのうちのいずれかを
指してもよい。片方向予測(Pモード)または双方向予測(Bモード)などのインターモードは
、いくつかの時間ベースの圧縮モードのうちのいずれかを指してもよい。

【0049】

いくつかの例では、画面コンテンツをコーディングするときなど、ビデオエンコーダ20
および/またはビデオデコーダ30は、ビデオスライス内のブロックをコーディングするた
めに、イントラブロックコピーイング(イントラBC)を実行してもよい。イントラBCを実行
するために、ビデオエンコーダ20および/またはビデオデコーダ30は、現在のブロックの
予測のために現在のブロックと同じピクチャ内の以前にコーディングされたビデオデー
タのブロックを使用してもよい。上記で説明したように、イントラBCは、インターに似たコ
ーディングツール(ピクチャのピクセル値がベクトルによって示される予測子ブロックに
基づいて予測されることを意味する)である。しかしながら、従来のイントラ予測ピク
チャにおいてイントラBCを使用することは、複雑さを増加させる可能性があり、および/ま
たはイントラBCのコーディング効率を低下させる可能性がある。加えて、本開示の技法の
前に、イントラBCを使用するビデオコードの実装は、従来のイントラおよびインター予
測を実行するために使用されるものに比べて、(たとえば、イントラBCがピクチャ内部のブ
ロックコピーをもたらすため)イントラBCを実行するために追加のツールを用いて構成さ
れることを必要とする。そのように、イントラBCに現在適用されている制約の一部または
すべてを維持しながら、ハードウェアアーキテクチャへの(重大な)変更なしに、ビデオコ
ーダがイントラBCによって提供される効率を増すことができるようにすることが望ましい
可能性がある。

【0050】

本開示の1つまたは複数の技法によれば、従来のイントラ予測技法を使用して現在のピ
クチャのブロックを予測するためにイントラBCを実行することとは対称的に、ビデオエン
コーダ20および/またはビデオデコーダ30は、従来のインター予測と同様の技法を使用し
てイントラBCを実行することができる。たとえば、ビデオエンコーダ20および/またはビ
デオデコーダ30は、現在のピクチャのブロックを予測するために使用される参照ピクチャ

10

20

30

40

50

リスト(RPL)内に現在のピクチャの指標を挿入し、参照ピクチャバッファ内に現在のピクチャのバージョン(すなわち、インスタンス)を記憶し、参照ピクチャバッファ内に記憶された現在のピクチャのバージョン内に含まれるビデオデータの予測子ブロックに基づいて、現在のピクチャ内のビデオデータのブロックをコーティングすることができる。このようにして、ビデオエンコーダ20および/またはビデオデコーダ30は、イントラBCに現在適用されている制約の一部またはすべてを維持しながら、イントラBCによって提供される効率を増すことができる。また、このようにして、ビデオエンコーダ20および/またはビデオデコーダ30は、重大な変更なしに、イントラBCのために従来のイントラおよびインターフレームのためのハードウェアアーキテクチャを再利用することができる。

【0051】

図2は、表示順で、ピクチャ34、35A、36A、38A、35B、36B、38B、および35Cを含む例示的なビデオシーケンス33を示す概念図である。これらのピクチャの1つまたは複数は、Pスライス、Bスライス、またはIスライスを含んでもよい。いくつかの場合では、ビデオシーケンス33は、ピクチャのグループ(GOP)と呼ばれることがある。ピクチャ39は、ビデオシーケンス33の後に発生するシーケンスの表示順で第1のピクチャである。図2は、全体的に、ビデオシーケンスのための例示的な予測構造を表し、異なるインター予測されたスライスタイプを符号化するために使用されるピクチャ参照を説明することのみを意図している。実際のビデオシーケンスは、異なるスライスタイプを異なる表示順で含むより多いまたはより少ないビデオピクチャを含んでもよい。

【0052】

ブロックベースのビデオコーディングのために、ビデオシーケンス33内に含まれるビデオピクチャの各々は、ビデオブロックまたはコーディング単位(CU)に分割され得る。ビデオピクチャの各CUは、1つまたは複数の予測単位(PU)を含んでもよい。いくつかの例では、ピクチャ内のPUを予測するために利用可能な予測方法は、ピクチャタイプに依存し得る。一例として、イントラ予測されたピクチャ(Iピクチャ)のスライス内のビデオブロックまたはPUは、イントラ予測モード(すなわち、同じピクチャ内の隣接ブロックに対する空間予測)を使用して予測されてもよい。別の例として、インター予測されたピクチャ(BピクチャまたはPピクチャ)のスライス内のビデオブロックまたはPUは、インターまたはイントラ予測モード(すなわち、同じピクチャ内の隣接ブロックに対する空間予測、または他の参照ピクチャに対する時間予測)を使用して予測されてもよい。言い換えれば、Iピクチャは、Iスライスを含んでもよく、Pピクチャは、IスライスとPスライスの両方を含んでもよく、Bピクチャは、Iスライスと、Pスライスと、Bスライスとを含んでもよい。

【0053】

Pスライスのビデオブロックは、参照ピクチャリスト内で特定された参照ピクチャから一方向予測コーディングを使用して符号化されてもよい。Bスライスのビデオブロックは、複数の参照ピクチャリスト内で特定された複数の参照ピクチャから双方向予測コーディングを用いて符号化されてもよい。

【0054】

図2の例では、第1のピクチャ34は、Iピクチャとしてイントラモードコーディングのために指定される。他の例では、第1のピクチャ34は、たとえば、前のシーケンスの第1のピクチャに対してPピクチャまたはBピクチャとしてインターモードコーディングでコーディングされてもよい。ビデオピクチャ35A~35C(総称して「ビデオピクチャ35」)は、過去のピクチャおよび将来のピクチャに対する双予測を使用してBピクチャとしてコーディングするために指定される。図2の例に示すように、ピクチャ35Aは、ピクチャ34およびピクチャ36Aからビデオピクチャ35Aへの矢印によって示されているように、第1のピクチャ34およびピクチャ36Aに対してBピクチャとして符号化されてもよい。図2の例では、第1のピクチャ34およびピクチャ36Aは、ピクチャ35Aのブロックの予測の際に使用される参照ピクチャリスト内に含まれてもよい。ピクチャ35Bおよび35Cは、同様に符号化される。

【0055】

ビデオピクチャ36A~36B(総称して「ビデオピクチャ36」)は、過去のピクチャに対する

一方向予測を使用してPピクチャまたはBピクチャとしてコーディングするために指定されてもよい。図2の例に示すように、ピクチャ36Aは、ピクチャ34からビデオピクチャ36Aへの矢印によって示されているように、第1のピクチャ34に対してPピクチャまたはBピクチャとして符号化される。ピクチャ36Bは、ピクチャ38Aからビデオピクチャ36Bへの矢印によって示されているように、ピクチャ38Aに対してPピクチャまたはBピクチャとして同様に符号化される。

【0056】

ビデオピクチャ38A~38B(総称して「ビデオピクチャ38」)は、同じ過去のピクチャに対する一方向予測を使用してPピクチャまたはBピクチャとしてコーディングするために指定されてもよい。図2の例に示すように、ピクチャ38Aは、ピクチャ36Aからビデオピクチャ38Aへの2本の矢印によって示されているように、ピクチャ36Aへの2つの参照を用いて符号化される。ピクチャ38Bは、同様に符号化される。

【0057】

いくつかの例では、ピクチャの各々は、ピクチャが出力されるべき順序を示す一意の値(すなわち、特定のビデオシーケンス、たとえば、復号順序で瞬間デコーダリフレッシュ(IDR:instantaneous decoder refresh)ピクチャに続くピクチャのシーケンスに対して一意である)を割り当てられてもよい。この一意の値は、ピクチャ順序カウンタ(POC:picture order count)と呼ばれることがある。いくつかの例では、ピクチャが出力されるべき順序は、ピクチャがコーディングされる順序と異なってもよい。たとえば、ピクチャ35Aは、ピクチャ36Aの前に出力されてもよいが、ピクチャ36Aは、ピクチャ35Aの前にコーディングされてもよい。

【0058】

本開示の1つまたは複数の技法によれば、ビデオコーダ(たとえば、ビデオエンコーダ20またはビデオデコーダ30)は、現在のピクチャ内のブロックを予測するために使用される参照ピクチャリスト(RPL)内に現在のピクチャを挿入することによってイントラBCを実行してもよい。たとえば、図2の例では、ビデオコーダは、ピクチャ35A内のブロックを予測するために使用されるRPL内に、ピクチャ34およびピクチャ36Aの指標とともに、ピクチャ35Aの指標を挿入してもよい。ビデオコーダは、次いで、ピクチャ35Aのブロックをコーディングするとき、参照ピクチャとしてピクチャ35Aを使用してもよい。

【0059】

図3は、本開示に記載のイントラブロックコピーのための技法を使用することができるビデオエンコーダ20の一例を示すブロック図である。ビデオエンコーダ20を、他のコーディング規格に関する本開示の限定なしで、説明の目的のためのHEVCコーディングの文脈で説明する。さらに、ビデオエンコーダ20は、HEVCの範囲拡張に従って技法を実装するように構成されてもよい。

【0060】

ビデオエンコーダ20は、ビデオスライス内のビデオブロックのイントラコーディングとインターコーディングとを実行してもよい。イントラコーディングは、所与のビデオピクチャ内のビデオにおける空間的冗長性を低減または除去するために、空間予測に依存する。インターコーディングは、ビデオシーケンスの隣接ピクチャ内のビデオにおける時間的冗長性を低減もしくは除去するために、または、他のビューにおけるビデオで冗長性を低減もしくは除去するために、時間予測またはインタビュー予測に依存する。

【0061】

図3の例では、ビデオエンコーダ20は、ビデオデータメモリ40と、予測処理ユニット42と、参照ピクチャメモリ64と、加算器50と、変換処理ユニット52と、量子化処理ユニット54と、エン트로ピー処理ユニット56とを含んでもよい。予測処理ユニット42は、次に、動き推定ユニット44と、動き補償ユニット46と、イントラ予測ユニット48とを含む。ブロック再構成のために、ビデオエンコーダ20はまた、逆量子化処理ユニット58と、逆変換処理ユニット60と、加算器62とを含む。デブロックフィルタ(図3に示さず)はまた、再構成されたビデオからブロックネスアーティファクトを除去するためにブロック境界をフィ

ルタリングするために含まれてもよい。所望ならば、デブロッキングフィルタは、典型的には、加算器62の出力をフィルタリングすることになる。追加のループフィルタ(ループ内またはループ後)はまた、デブロッキングフィルタに加えて使用されてもよい。

【0062】

ビデオデータメモリ40は、ビデオエンコーダ20の構成要素によって符号化されるべきビデオデータを記憶してもよい。ビデオデータメモリ40内に記憶されるビデオデータは、たとえば、ビデオソース18から取得されてもよい。参照ピクチャメモリ64は、(たとえば、イントラまたはインター予測コーディングモードとも呼ばれるイントラまたはインターコーディングモードで)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際に使用するための参照ビデオデータを記憶する復号ピクチャバッファ(DPB:decoding picture buffer)の一例である。ビデオデータメモリ40および参照ピクチャメモリ64は、シンクロナスDRAM(SDRAM)を含むダイナミックランダムアクセスメモリ(DRAM)、磁気抵抗性RAM(MRAM)、抵抗性RAM(RRAM(登録商標))、または他のタイプ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てもよい。ビデオデータメモリ40および参照ピクチャメモリ64は、同じメモリデバイスまたは別個のメモリデバイスによって提供されてもよい。様々な例では、ビデオデータメモリ40は、ビデオエンコーダ20の他の構成要素とともにオンチップであってもよく、または、これらの構成要素に対してオフチップであってもよい。

【0063】

符号化プロセス中、ビデオエンコーダ20は、コーディングされるべきビデオピクチャまたはスライスを受信する。ピクチャまたはスライスは、複数のビデオブロックに分割されてもよい。動き推定ユニット44および動き補償ユニット46は、時間的圧縮を提供するため、またはインタービュー圧縮を提供するために、1つまたは複数の参照ピクチャ内の1つまたは複数のブロックに対する受信ビデオブロックのインター予測コーディングを実行する。イントラ予測ユニット48は、代替的には、空間的圧縮を提供するために、コーディングされるべきブロックと同じピクチャまたはスライス内の1つまたは複数の隣接ブロックに対して受信ビデオブロックのイントラ予測コーディングを実行してもよい。ビデオエンコーダ20は、(たとえば、ビデオデータの各ブロックのための適切なコーディングモードを選択するために)複数のコーディングパスを実行してもよい。

【0064】

さらに、分割ユニット(図示せず)は、以前のコーディングプロセスにおける以前に分割方式の評価に基づいて、ビデオデータのブロックをサブブロックに分割してもよい。たとえば、分割ユニットは、最初にスライスまたはピクチャをLCUに分割し、レート-歪み分析(たとえば、レート-歪み最適化)に基づいて、LCUの各々をサブCUに分割してもよい。予測処理ユニット42は、さらに、サブCUへのLCUの分割を示す4分木データ構造を生成してもよい。4分木のリーフノードCUは、1つまたは複数のPUと、1つまたは複数のTUとを含んでもよい。

【0065】

予測処理ユニット42は、たとえば、エラー結果に基づいて、コーディングモードの1つ、イントラまたはインターを選択し、得られたイントラまたはインターコーディングされたブロックを、残差ブロックデータを生成するために加算器50に、参照ピクチャとして使用するための符号化ブロックを再構成するために加算器62に提供してもよい。予測処理ユニット42はまた、動きベクトル、イントラモードインジケータ、分割情報などの構文要素、および他のそのような構文情報を、エントロピー符号化ユニット56に提供する。

【0066】

動き推定ユニット44および動き補償ユニット46は、高度に統合されてもよいが、概念的な目的のために別々に示されている。動き推定ユニット44によって実行される動き推定は、ビデオブロックに関する動きを推定する動きベクトルを生成するプロセスである。動きベクトルは、たとえば、現在のピクチャ(または他のコーディング単位)内のコーディングされた現在のブロックに対する参照ピクチャ(または他のコーディング単位)内の予測プロ

ックに対する現在のビデオピクチャ内のビデオブロックのPUの変位を示してもよい。予測ブロックは、絶対差合計(SAD:sum of absolute difference)、自乗差合計(SSD:sum of square difference)、または他の差分メトリックによって決定され得るピクセル差の観点から、コーディングされるべきブロックに密接に一致することが見出されたブロックである。いくつかの例では、ビデオエンコーダ20は、参照ピクチャメモリ64内に記憶された参照ピクチャのサブ整数ピクセル位置の値を計算してもよい。たとえば、ビデオエンコーダ20は、参照ピクチャの4分の1ピクセル位置の値、8分の1ピクセル位置の値、または他の分数ピクセル位置の値を補間してもよい。したがって、動き推定ユニット44は、フルピクセル位置および分数ピクセル位置に対する動き探索を実行し、分数ピクセル精度で動きベクトルを出力してもよい。

10

【0067】

動き推定ユニット44は、参照ピクチャの予測ブロックの位置とPUの位置とを比較することによって、インターコーディングスライス内のビデオブロックのPUのための動きベクトルを計算する。参照ピクチャは、参照ピクチャメモリ64内に記憶された1つまたは複数の参照ピクチャを識別する1つまたは複数の参照ピクチャリスト(RPL)から選択されてもよい。動き推定ユニット44は、エントロピー符号化ユニット56および動き補償ユニット46に計算された動きベクトルを送信する。いくつかの例では、動き推定ユニット44は、選択された参照ピクチャの指標をエントロピー符号化ユニット56に送ってもよい。

【0068】

いくつかの例では、動き推定ユニット44は、1つまたは複数のRPLを生成してもよい。たとえば、動き推定ユニット44は、出力順で現在のピクチャの前にある(すなわち、現在のピクチャよりも低いPOC値を有する)ビデオデータのピクチャを含み得る第1のRPL(リスト0)と、出力順で現在のピクチャの後にある(すなわち、現在のピクチャよりも高いPOC値を有する)ビデオデータのピクチャを含み得る第2のRPL(リスト1)とを生成してもよい。いくつかの例では、RPLは、短期ピクチャと長期ピクチャとを含んでもよく、これらは、長期ピクチャが短期ピクチャよりも長い間、参照ピクチャメモリ64に記憶され得る点で異なってもよい。いくつかの例では、参照ピクチャメモリ64内のピクチャは、たとえば、長期、短期、などとしてマークされてもよい。

20

【0069】

いくつかの例では、動き推定ユニット44は、1つまたは複数の参照ピクチャセット(RPS: reference picture set)に基づいて1つまたは複数のRPLを生成してもよい。1つまたは複数のRPSは、1)短期RPS前もしくはRefPicSetStCurrBeforeと呼ばれる、現在のピクチャを予測するために利用可能であり、出力順で現在のピクチャの前にある短期ピクチャの1つもしくは複数のセット、2)短期RPS後もしくはRefPicSetStCurrAfterと呼ばれる、現在のピクチャを予測するために利用可能であり、出力順で現在のピクチャの後にある短期ピクチャのセットと、3)短期RPS利用不可もしくはRefPicSetStFollと呼ばれる、現在のピクチャを予測するために利用可能ではないが、コーディング順で次のピクチャをコーディングするために使用され得る短期ピクチャのセット、4)ロングタームRPS利用可能もしくはRefPicSetLtCurrと呼ばれる、現在のピクチャを予測するために利用可能である長期ピクチャのセット、および/または、5)長期RPS利用不可またはRefPicSetLtFollと呼ばれる、現在のピクチャを予測するために利用不可能である長期ピクチャのセットを含んでもよい。

30

40

【0070】

いくつかの例では、動き推定ユニット44は、特定の順序で1つまたは複数のRPL内の1つまたは複数のRPS内のピクチャを含んでもよい。たとえば、RPLを生成するために、動き推定ユニット44は、最初に、現在のピクチャを予測するために利用可能な短期RPS(たとえば、RefPicSetStCurrBeforeおよび/またはRefPicSetStCurrAfter)内のピクチャを含んでもよく、続いて、現在のピクチャを予測するために利用可能な長期RPS(たとえば、RefPicSetLtCurr)内のピクチャを含んでもよい。いくつかの例では、RPL内の各エントリは、より以前にRPLに含まれたピクチャ(たとえば、短期RPL内のピクチャ)が、より後にRPLに含まれたピクチャ(たとえば、長期RPS内のピクチャ)よりも低いインデックス値を有するよう

50

に、インデックス値を有してもよい。

【0071】

上記で説明したように、動き推定ユニット44は、選択された参照ピクチャの指標をエントロピー符号化ユニット56に送ってもよい。いくつかの例では、動き推定ユニット44は、RPL内の選択された参照ピクチャのインデックス値を送ることによって指標を送ってもよい。

【0072】

いくつかの例では、動き推定ユニット44は、ビデオデコーダが1つまたは複数の以前のRPLから1つまたは複数の現在のRPLを予測できるようにするために情報を出力してもよい。たとえば、動き推定ユニット44は、ビデオデコーダが現在のスライスのための1つまたは複数のRPLを生成するために以前のスライスのための1つまたは複数のRPLを変更できるようにする1つまたは複数の構文要素の出力を生じさせてもよい。

【0073】

本開示の1つまたは複数の技法によれば、参照ピクチャとして他のピクチャを使用するためにインター予測を制限することとは対照的に、動き推定ユニット44は、現在のピクチャ内に含まれるビデオデータのブロックを予測するために、参照ピクチャとして現在のピクチャを使用してもよい。たとえば、動き推定ユニット44は、参照ピクチャメモリ64内に現在のピクチャのバージョンを記憶してもよい。いくつかの例では、動き推定ユニット44は、固定された値に初期化されたピクセル値を有する現在のピクチャの初期化されたバージョンを記憶してもよい。いくつかの例では、固定された値は、現在のピクチャのサンプルのビット深度に基づいてもよい。たとえば、固定された値は、 $1 \ll (\text{bitDepth} - 1)$ であってもよい。いくつかの例では、動き推定ユニット44は、現在のピクチャの任意のブロックを符号化する前に、現在のブロックの初期化バージョンを記憶してもよい。現在のピクチャの初期化バージョンを記憶することによって、動き推定ユニット44は、予測ブロックの探索(すなわち探索範囲)をすでに再構成されたブロックに限定することを必要とされなくてもよい。対照的に、動き推定ユニット44が現在のピクチャの初期化バージョンを記憶しない場合、たとえば、予測ブロックの探索は、デコーダ/エンコーダのミスマッチを回避するために、すでに再構成されたブロックに限定されてもよい。

【0074】

予測処理ユニット42は、現在のピクチャのための1つまたは複数のRPLを生成してもよい。たとえば、予測処理ユニット42は、現在のピクチャのためのRPLに現在のピクチャを含めてもよい。いくつかの例では、予測処理ユニット42は、RPL内の特定の位置に現在のピクチャを含めてもよい。一例として、予測処理ユニット42は、長期RPS内のピクチャの前にRPL内に現在のピクチャを挿入してもよい。たとえば、予測処理ユニット42は、長期RPSからのピクチャのインデックス値よりも小さいインデックス値を用いてRPL内に現在のピクチャを挿入してもよい。いくつかの例では、予測処理ユニット42は、長期RPS内のピクチャの直前にRPL内に現在のピクチャを挿入してもよい。

【0075】

別の例として、予測処理ユニット42は、長期RPSからのピクチャを挿入した後に、RPL内に現在のピクチャを挿入してもよい。たとえば、予測処理ユニット42は、長期RPS内のピクチャのインデックス値よりも大きいインデックス値を用いてRPL内に現在のピクチャを挿入してもよい。いくつかの例では、動き推定ユニット44は、長期RPSからのピクチャを挿入した直後に、RPL内に現在のピクチャを挿入してもよい。

【0076】

別の例として、予測処理ユニット42は、固定された位置においてRPL内に現在のピクチャを挿入してもよい。たとえば、予測処理ユニット42は、固定されたインデックス値を用いてRPL内に現在のピクチャを挿入してもよい。いくつかの例では、固定されたインデックス値は、-1、または $\text{num_ref_idx_l1_active_minus1} + 1$ であってもよい。そのような例のいくつかでは、動き推定ユニット44は、現在のブロックがイントラBCを使用してコーディングされたことを示すフラグ(すなわち、 intra_bc_flag)をコーディングしなくてもよい

。

【0077】

動きベクトルが時間的動きベクトル予測(TMVP:temporal motion vector prediction)を使用して予測される場合などのいくつかの例では、動き推定ユニット44は、現在のピクチャがそれ自体のための並置されたピクチャとして使用されないように、1つまたは複数の制約を適用してもよい。たとえば、動き推定ユニット44は、Xがcollocated_from_l0_flagに等しいとき、RefPicListX[collocated_ref_idx]が現在のピクチャではないように、TMVP(たとえば、collocated_ref_idx)のために使用される並置されたピクチャの参照インデックスを指定する構文要素をコーディングしてもよい。

【0078】

上記で説明したように、ビデオデータの現在のピクチャのビデオデータのブロックを符号化するとき、動き推定ユニット44は、現在のブロックに密接に一致する予測ブロックを選択してもよい。本開示の1つまたは複数の技法によれば、他のピクチャのブロックを探索するのはとは対照的に(または、それに加えて)、動き推定ユニット44は、現在のピクチャの現在のブロックのための予測ブロックとして使用するために現在のピクチャ内に位置するブロックを選択してもよい。たとえば、動き推定ユニット44は、現在のピクチャを含む1つまたは複数の参照ピクチャを含むピクチャに対して探索を実行してもよい。各ピクチャについて、動き推定ユニット44は、たとえば、ピクセルごとの絶対差合計(SAD)、自乗差合計(SSD)、平均絶対差(MAD:mean absolute difference)、平均自乗差(MSD:mean squared difference)、などを使用して、予測ブロックが現在のブロックに一致する程度を反映する探索結果を計算してもよい。次いで、動き推定ユニット44は、現在のブロックに最も一致したピクチャ内のブロックを識別し、ブロックおよびピクチャ(現在のピクチャであってもよい)の位置を予測処理ユニット42に示してもよい。このようにして、動き推定ユニット44は、たとえば、動き推定ユニット44が、予測ブロックが現在のピクチャ、すなわち、予測されている現在のブロックと同じピクチャ内に含まれていると決定したとき、イントラBCを実行してもよい。

【0079】

いくつかの例では、動き推定ユニット44は、現在のピクチャ内の予測ブロックの探索を制限してもよい。たとえば、現在のブロックが現在のスライス内に位置する場合、動き推定ユニット44は、予測ブロックの探索を、現在のスライスの以前にコーディングされた領域(たとえば、現在のスライス内の現在のブロックの上および/または左の領域)に制限してもよい。そのように、いくつかの例では、現在のピクチャの以前のスライスの以前にコーディングされた領域は、イントラBCを実行するための予測ブロックとして使用されなくてもよい。しかしながら、上記で説明したように、いくつかの例では、現在のピクチャの以前のスライスの以前にコーディングされた領域は、イントラBCを実行するための予測ブロックとして使用されてもよい。いくつかの例では、動き推定ユニット44は、(たとえば、動き推定の探索範囲のように)参照ピクチャ間の動き探索と同様の制約を使用して、予測ブロックの探索を制限してもよい。

【0080】

いくつかの例では、予測処理ユニット42は、現在のピクチャが現在のピクチャを予測するために使用されるRPL内に存在し得るかどうかを示すために、エントロピー符号化ユニット56に1つまたは複数の構文要素を符号化させてもよい。一例として、予測処理ユニット42は、ビデオデータのピクチャがそれら自体を予測するために使用されるRPL内に存在し得るかどうかを示す単一の構文要素を、エントロピー符号化ユニット56に符号化させてもよい。いくつかの例では、予測処理ユニット42は、エントロピー符号化ユニット56に、現在のピクチャによって参照されるビデオパラメータセット(VPS:video parameter set)、現在のピクチャによって参照されるシーケンスパラメータセット(SPS:sequence parameter set)、または現在のピクチャによって参照されるピクチャパラメータセット(PPS:picture parameter set)内に、単一の構文要素を含めさせてもよい。

【0081】

別の例として、予測処理ユニット42は、現在のピクチャが現在のピクチャを予測するために使用されるRPL内に存在し得るかどうかを示すために、エントロピー符号化ユニット56に複数の構文要素を符号化させてもよい。たとえば、予測処理ユニット42は、ビデオデータのピクチャがビデオデータのそれぞれのピクチャを予測するために使用される(すなわち、それら自体を予測するために使用される)RPL内に存在し得るかどうかを示す第1の構文要素を、エントロピー符号化ユニット56に符号化させてもよい。いくつかの例では、予測処理ユニット42は、エントロピー符号化ユニット56に、現在のピクチャによって参照されるVPS、現在のピクチャによって参照されるSPS、または現在のピクチャ(もしくは現在のピクチャのスライス)によって参照されるPPS内に第1の構文要素を含めさせてもよい。構文要素が、ビデオデータのピクチャがそれ自体を予測するために使用されるRPL内に存在し得ることを示す場合などの、いくつかの例では、予測処理ユニット42は、ビデオデータの現在のピクチャが現在のスライスを予測するために使用されるRPL内に存在し得るかどうかを示す第2の構文要素を、エントロピー符号化ユニット56に符号化させてもよい。いくつかの例では、予測処理ユニット42は、エントロピー符号化ユニット56に、現在のスライスのスライスヘッダ内に第2の構文要素を含めさせてもよい。

10

【0082】

いくつかの例では、予測処理ユニット42は、ブロックがイントラブロックコピー(イントラBC)を使用してコーディングされているかどうかを示す構文要素を、エントロピー符号化ユニット56に符号化させなくてもよい。たとえば、予測処理ユニット42は、本開示の技法に従ってイントラBCを使用して予測されるブロックのコーディング単位構文内にintr_a_bc_flagを、エントロピー符号化ユニット56に符号化させなくてもよい。

20

【0083】

いくつかの例では、イントラBCを使用してBスライスおよびPスライス内のブロックをコーディングすることに加えて、予測処理ユニット42は、現在のピクチャのIスライスのブロックを符号化するために現在のピクチャを含む1つまたは複数のRPLを構成してもよい。いくつかのそのような例では、1つまたは複数のRPLが現在のピクチャのみを含むと仮定され得る。いくつかの例では、予測処理ユニット42は、現在のピクチャが現在のピクチャ内に含まれるIスライスの参照ピクチャとして使用され得るかどうかを示すために、エントロピー符号化ユニット56に構文要素を符号化させてもよい。いくつかの例では、予測処理ユニット42は、エントロピー符号化ユニット56に、現在のピクチャによって参照されるVPS、現在のピクチャによって参照されるSPS、現在のピクチャによって参照されるPPS、または現在のIスライスのスライスヘッダ内に構文要素を含めさせてもよい。いくつかの例では、予測処理ユニット42は、AMVPおよびマージの一方または両方を依然として使用してもよい。いくつかの例では、予測処理ユニット42は、デコーダが固定値、たとえば0として目標参照インデックスを導出するように、エントロピー符号化ユニット56に、IスライスのためのAMVPのための目標参照インデックスを合図させなくてもよい。いくつかの例では、予測処理ユニット42は、エントロピー符号化ユニット56に、IスライスのためのAMVPのための目標参照インデックスを合図させてもよいが、目標参照インデックスの値は、固定値、たとえば0に限定されてもよい。

30

【0084】

上記で説明したように、参照ピクチャメモリ64によって記憶されたピクチャは、短期、長期、別のマーキングとしてマークされてもよく、および/またはマークされなくてもよい。現在のスライスがIスライスであり、イントラBCが有効である場合などのいくつかの例では、予測処理ユニット42は、現在のピクチャを、長期または短期のいずれかとしてマークしてもよい。現在のスライスがIスライスであり、イントラBCが有効である場合などのいくつかの例では、予測処理ユニット42は、現在のピクチャを、長期または短期のいずれかとしてマークしなくてもよい。

40

【0085】

いくつかの例では、予測処理ユニット42は、現在のピクチャをコーディングする前に、現在のピクチャを長期としてマークしてもよく、現在のピクチャをコーディングした後に

50

、現在のピクチャを短期としてマークしてもよい。現在のスライスがIスライスである場合などの、そのような例のいくつかでは、予測処理ユニット42は、(「長期」としてマークされた)イントラBC参照を純粹に指す候補、または(「短期」としてマークされた)インター参照を指す他の候補を含むように、マージ候補リストを生成してもよい。このようにして、予測処理ユニット42は、イントラBC候補と通常の(インター予測)候補の両方を含むように候補リストを生成してもよい。いくつかの例では、予測処理ユニット42は、イントラBC参照およびインター予測参照からマージ候補を双方向予測してもよい。現在のスライスがインターコーディングスライス(たとえば、BスライスまたはPスライス)である場合などの、そのような例のいくつかでは、予測処理ユニット42は、現在のブロックがイントラBCでコーディングされているかどうかを示すフラグ(intra_bc_flag)を示す追加のフラグを省略してもよい。

10

【0086】

時間的動きベクトル予測(TMVP)が有効であり、現在のフレームが追加の参照ピクチャであると仮定することによってイントラBCが有効である場合などのいくつかの例では、予測処理ユニット42は、目標マージ参照ピクチャが現在のピクチャまたはRPL内の第1のピクチャのどちらであるのか(たとえば、RefPicList0[0]またはRefPicList1[0]のどちらであるのか)を示す構文要素を、エントロピー符号化ユニット56にコーディングさせてもよい。いくつかの例では、予測処理ユニット42は、参照の長期(イントラBC)カテゴリまたは参照の短期カテゴリのための目標インデックスを導出または合図し、同一位置のブロックの参照ピクチャのカテゴリに基づいて異なる目標マージインデックスを適用することによって、参照ピクチャを識別してもよい。たとえば、予測処理ユニット42は、参照の長期(イントラBC)カテゴリまたは参照の短期カテゴリのための目標インデックスを合図または導出することによって参照ピクチャを識別してもよく、同一位置のブロックの参照ピクチャのカテゴリに基づいて、異なる目標マージインデックスが適用される。

20

【0087】

上記で説明したように、予測処理ユニット42は、ビデオデータの現在のブロックとビデオデータの予測子ブロックとの間の変位を表す動きベクトルを決定し、決定されたベクトルをエントロピー符号化ユニット56および動き補償ユニット46に出力してもよい。いくつかの例では、予測処理ユニット42は、整数精度の動きベクトルを決定してもよい。現在のピクチャが長期参照ピクチャとしてマークされている場合などの、そのような例では、予測処理ユニット42は、現在のピクチャを予測するために通常の長期参照ピクチャ(すなわち、現在のピクチャではない長期参照ピクチャ)を使用しなくてもよい。また、そのような例では、予測処理ユニット42は、たとえば、イントラBC動きベクトルが空間的および時間的に隣接するブロックのイントラBC動きベクトルのみに基づいて予測される場合、どのような変更もなしに、高度動きベクトル予測(AMVP:advanced motion vector prediction)またはHEVCバージョン1におけるようなマージ関連復号プロセスを利用してもよい。

30

【0088】

いくつかの例では、予測処理ユニット42は、異なるレベルの精度で動きベクトルを決定してもよい。たとえば、予測処理ユニット42は、整数精度、デフォルトの精度、または最も詳細な動き精度(たとえば、HEVCにおける1/4ピクセル(「pel」)精度)で動きベクトルを決定してもよい。いくつかの例では、予測処理ユニット42は、たとえば、現在のピクチャによって参照されるSPSまたはVPS内で、コーディングされたイントラBC動きベクトルの精度を示す構文要素を符号化してもよい。いくつかの例では、イントラBC動きベクトルの精度は、ピクチャレベルで適応されてもよく、予測処理ユニット42は、たとえば、現在のブロックによって参照されるPPSまたはスライスにおいて、コーディングされたイントラBC動きベクトルの精度を示す構文要素を、エントロピー符号化ユニット56に符号化させてもよい。

40

【0089】

いくつかの例では、予測処理ユニット42は、イントラBC動きベクトルの精度レベルを補償するために、1つまたは複数の動作を実行してもよい。一例として、ブロックを参照ピ

50

クチャメモリ64内に記憶する前に、予測処理ユニット42は、最高精度が1/4pelであるとき、2などだけ、または、任意の他の丸め手段によって、左シフト後に±2などだけ、イントラBCでコーディングされた各ブロックの動きベクトルのxおよびy座標を左シフトしてもよい。別の例として、整数の精度を有するイントラBC動きベクトルで現在のスライスをコーディングするとき、予測処理ユニット42は、最高精度が1/4pelであるとき、2などだけ、各イントラBCコーディングブロックの動きベクトルが右シフトされるように、同一位置のピクチャを処理してもよい。現在のスライスが最高の動き精度を有するイントラBC動きベクトルでコーディングされるときなどの、いくつかの例では、予測処理ユニット42は、上記の動きベクトル右シフトの処理を適用しなくてもよい。

【0090】

現在のピクチャが長期参照ピクチャとしてマークされているいくつかの例では、予測処理ユニット42は、現在のピクチャを予測するために、通常長期参照ピクチャを依然として使用してもよい。通常長期参照ピクチャを参照する動きベクトルおよび現在のピクチャを参照する動きベクトルが、マージまたはAMVPの間、互いから予測されるのを防止するために、予測処理ユニット42は、マージまたはAMVP処理の間、通常長期参照ピクチャと現在のピクチャとを区別してもよい。たとえば、予測処理ユニット42は、参照ピクチャのピクチャ順序カウント(POC:picture order count)値が現在のピクチャのPOCと同じであるかどうかを決定することによって、通常長期参照ピクチャを現在のピクチャから区別してもよい。

【0091】

動き補償ユニット46によって実行される動き補償は、動き推定ユニット44によって決定された動きベクトルに基づいて、予測ブロックをフェッチまたは生成することを含んでもよい。再び、動き推定ユニット44および動き補償ユニット46は、いくつかの例では、機能的に統合されてもよい。現在のブロックのPUのための動きベクトルを受信すると、動き補償ユニット46は、動きベクトルが参照ピクチャリスト(RPL)のうちの1つにおいて指し示す予測ブロックの位置を突き止めてもよい。加算器50は、以下に説明するように、コーディングされている現在のブロックのピクセル値から予測ブロックのピクセル値を除算することによって残差ビデオブロックを形成し、ピクセル差分値を形成する。一般に、動き推定ユニット44は、輝度成分に対する動き推定を実行し、動き補償ユニット46は、彩度成分と輝度成分の両方のための輝度成分に基づいて計算された動きベクトルを使用する。予測処理ユニット42はまた、ビデオスライスのビデオブロックを復号する際にビデオデコーダ30によって使用するためのビデオブロックおよびビデオスライスに関連付けられた構文要素を生成してもよい。

【0092】

イントラ予測ユニット48は、上記で説明したように、動き推定ユニット44および動き補償ユニット46によって実行されるインター予測の代替として、現在のブロックをイントラ予測してもよい。具体的には、イントラ予測ユニット48は、現在のブロックを符号化するために使用するイントラ予測モードを決定してもよい。いくつかの例では、イントラ予測ユニット48は、たとえば、別個の符号化パスの間、様々なイントラ予測モードを使用してブロックを符号化してもよく、イントラ予測ユニット48は、複数のイントラ予測モードから使用するための適切なイントラ予測モードを選択してもよい。

【0093】

たとえば、イントラ予測ユニット48は、様々な試験されたイントラ予測モードに対してレート-歪み分析を使用してレート-歪み値を計算し、試験されたモードの中から最良のレート-歪み特性を有するイントラ予測モードを選択してもよい。レート-歪み分析は、一般に、符号化ブロックと、符号化ブロックを生成するために符号化された元の未符号化ブロックとの間の歪み(または誤差)の量、ならびに、符号化ブロックを生成するために使用されたビットレート(すなわち、ビット数)を決定する。イントラ予測ユニット48は、どのイントラ予測モードがブロックのための最良のレート-歪み値を示すのかを決定するために、様々な符号化ブロックに関する歪みおよびレートから比を計算してもよい。

【0094】

いくつかの例では、イントラ予測ユニット48によって使用するために利用可能な複数のイントラ予測モードは、平面予測モードと、DC予測モードと、1つまたは複数の角度予測モードとを含んでもよい。選択されたモードに関係なく、イントラ予測ユニット48は、常に、現在のブロックに隣接する再構成されたブロックに基づいて現在のブロックを予測してもよい。一例として、平面予測モードを使用するとき、イントラ予測ユニット48は、決定された水平および垂直予測を平均することによって現在のブロックを予測してもよい。いくつかの例では、イントラ予測ユニット48は、(現在のブロックを予測するとき、右隣接ブロックのサンプルが再構成されない可能性があるとき)左隣接ブロックおよび右上隣接ブロックに基づいて水平予測を決定してもよく、(現在のブロックを予測するとき、下隣接ブロックのサンプルが再構成されない可能性があるとき)上隣接ブロックおよび左下隣接ブロックに基づいて水平予測を決定してもよい。

10

【0095】

別の例として、DC予測モードを使用するとき、イントラ予測ユニット48は、一定値で現在のブロックのサンプルを予測してもよい。いくつかの例では、一定値は、左隣接ブロック内のサンプルと上隣接ブロック内のサンプルの平均を表してもよい。別の例として、1つまたは複数の角度予測モードのうちの1つを使用するとき、イントラ予測ユニット48は、予測方向によって示される隣接ブロックからのサンプルに基づいて、現在のブロックのサンプルを予測してもよい。図4は、イントラ予測ユニット48によって使用され得る角度イントラ予測モードの一例を示す。図4によって示された矢印は、(すなわち、現在のブロックから離れる方向に延びる)予測方向を表す。

20

【0096】

いくつかの例では、複数のイントラ予測モードの各々は、イントラ予測ユニット48によって(すなわち、ビデオデコーダに)合図され得る対応するモードインデックスを有してもよい。図4に示す数字は、上記で説明した角度イントラ予測モードに対応するモードインデックスである。図4によって示されたモードインデックスに加えて、平面モードは、0のインデックスを有してもよく、DCモードは、1のインデックスを有してもよい。上記で説明したように、かつ、本開示の1つまたは複数の技法によれば、イントラモードインデックスに基づいて現在のフレーム内のブロックをコーディングするのとは対照的に、動き推定ユニット44は、動きベクトルによって識別された現在のフレーム内の予測子ブロックに基づいて、現在のフレーム内のブロックを予測してもよい。

30

【0097】

ビデオエンコーダ20は、コーディングされている元のビデオブロックから、予測処理ユニット42からの予測データを減算することによって、残差ビデオブロックを形成する。加算器50は、この減算演算を実行する1つまたは複数の構成要素を表す。

【0098】

変換処理ユニット52は、離散コサイン変換(DCT)または概念的に同様の変換などの変換を残差ブロックに適用し、残差変換係数値を含むビデオブロックを生成する。変換処理ユニット52は、DCTと概念的に同様の他の変換を実行してもよい。ウェーブレット変換、整数変換、サブバンド変換、または他のタイプの変換はまた、使用され得る。いずれの場合にも、変換処理ユニット52は、残差ブロックに変換を適用し、残差変換係数のブロックを生成する。変換は、残差情報を、ピクセル値領域から、周波数領域などの変換領域に変換してもよい。

40

【0099】

変換処理ユニット52は、得られた変換係数を量子化処理ユニット54に送ってもよい。量子化処理ユニット54は、ビットレートをさらに低減するために、変換係数を量子化する。量子化処理は、係数の一部またはすべてに関連するビット深度を低減してもよい。量子化の程度は、量子化パラメータを調整することによって変更されてもよい。いくつかの例では、量子化処理ユニット54は、次いで、量子化変換係数を含む行列の走査を実行してもよい。代替的には、エントロピー符号化ユニット56は、走査を実行してもよい。

50

【0100】

量子化に続いて、エントロピー符号化ユニット56は、量子化変換係数をエントロピーコーディングする。たとえば、エントロピー符号化ユニット56は、コンテキスト適応型可変長コーディング(CAVLC)、コンテキスト適合型二値算術コーディング(CABAC)、構文ベースのコンテキスト適合型二値算術コーディング(SBAC)、確率間隔区分エントロピー(PIPE)コーディング、または別のエントロピー符号化技法を実行してもよい。コンテキストベースのエントロピーコーディングの場合には、コンテキストは、隣接ブロックに基づいてもよい。エントロピー符号化ユニット56によるエントロピーコーディングに続いて、符号化ビットストリームは、別のデバイス(たとえば、ビデオデコーダ30)に送信されてもよく、または後の送信もしくは検索のためにアーカイブされてもよい。

10

【0101】

逆量子化処理ユニット58および逆変換処理ユニット60は、たとえば、参照ブロックとして後で使用するために、ピクセル領域における残差ブロックを再構成するために、それぞれ、逆量子化および逆変換を適用する。

【0102】

動き補償ユニット46はまた、動き推定の際に使用するためのサブ整数ピクセル値を計算するために、参照ブロックに1つまたは複数の補間フィルタを適用してもよい。加算器62は、参照ピクチャメモリ64内に記憶するための再構成されたビデオブロックを生成するために、動き補償ユニット46によって生成された動き補償された予測ブロックに、再構成された残差ブロックを加算する。再構成されたビデオブロックは、後続のビデオピクチャ内のブロックをインターコーディングするために、参照ブロックとして動き推定ユニット44および動き補償ユニット46によって使用されてもよい。現在のピクチャが現在のピクチャを予測するために参照ピクチャとして使用される場合などのいくつかの例では、動き補償ユニット46および/または加算器62は、現在のピクチャをコーディングしながら、規則的な間隔で、参照ピクチャ64によって記憶された現在のピクチャのバージョンを更新してもよい。一例として、動き補償ユニット46および/または加算器62は、現在のピクチャの各ブロックをコーディングした後に、参照ピクチャメモリ64によって記憶された現在のピクチャのバージョンを更新してもよい。たとえば、現在のブロックのサンプルが初期化値として参照ピクチャメモリ64内に記憶される場合、動き補償ユニット46および/または加算器62は、現在のブロックのための再構成されたサンプルで、参照ピクチャメモリ64によって記憶された現在のピクチャの現在のサンプルを更新してもよい。

20

30

【0103】

フィルタリングユニット(図示せず)は、様々なフィルタリングプロセスを実行してもよい。たとえば、フィルタリングユニットは、デブロッキングを実行してもよい。すなわち、フィルタリングユニットは、再構成されたビデオのスライスまたはフレームを形成する複数の再構成されたビデオブロックを受信し、スライスまたはフレームからブロックネスアーティファクトを除去するために、ブロック境界をフィルタリングしてもよい。一例では、フィルタリングユニットは、ビデオブロックのいわゆる「境界強度」を評価する。ビデオブロックの境界強度に基づいて、ビデオブロックのエッジピクセルは、1つのビデオブロックからの遷移が、観察者が知覚するのがより困難になるように、隣接ビデオブロックのエッジピクセルに対してフィルタリングされてもよい。

40

【0104】

いくつかの例では、動き補償ユニット46および/または加算器62は、フィルタリングがサンプルへのフィルタリング(たとえば、デブロッキングおよび/またはSAO)を実行する前に、参照ピクチャメモリ64によって記憶された現在のピクチャのバージョンを更新してもよい。たとえば、フィルタリングユニットは、フィルタリングを適用する前に、ピクチャ全体がコーディングされるまで待機してもよい。このようにして、動き推定ユニット44は、フィルタリングを適用する前に、参照として現在のピクチャを使用してもよい。いくつかの例では、フィルタリングユニットは、参照ピクチャメモリ64によって記憶された現在のピクチャのバージョンが更新されたとき、フィルタリングを実行してもよい。たとえば

50

、フィルタリングユニットは、各ブロックが更新されたとき、フィルタリングを適用してもよい。このようにして、動き推定ユニット44は、フィルタリングを適用した後、参照として現在のピクチャを使用してもよい。

【0105】

技法のいくつかの異なる態様および例が本開示で説明されているが、技法の様々な態様および例は、互いに一緒にまたは別々に実行されてもよい。言い換えれば、技法は、上記で説明した様々な態様および例に厳密に限定されるべきではないが、組み合わせて使用されても、または一緒におよび/もしくは別々に実行されてもよい。加えて、特定の技法は、(イントラ予測ユニット48、動き補償ユニット46、またはエントロピー符号化ユニット56などの)ビデオエンコーダ20の特定のユニットに起因することがあるが、ビデオエンコーダ20の1つまたは複数の他のユニットはまた、そのような技法を実行することを担当してもよいことが理解されるべきである。

【0106】

このようにして、ビデオエンコーダ20は、本開示に記載の1つまたは複数の例示的な技法を実装するように構成されてもよい。たとえば、ビデオエンコーダ20は、現在のフレーム内に含まれる予測子ブロックを使用して、現在のフレーム内のビデオデータのブロックをコーディングするように構成されてもよい。ビデオエンコーダ20は、さらに、たとえば、イントラBCを使用して現在のピクチャの1つまたは複数のブロックをコーディングする目的のため、VPS/SPS/PPSを参照するピクチャがピクチャ自体の参照ピクチャリスト内に存在し得るかどうかを示す構文要素を含むビットストリームを出力するように構成されてもよい。すなわち、ブロックがイントラBCモードを使用してコーディングされるとき、ビデオエンコーダ20は、(構文要素が、現在のピクチャがそれ自体のための参照ピクチャリスト内に含まれ得ることを示すと仮定して)たとえば、インデックス値がピクチャ自体に対応するように参照ピクチャリスト内へのインデックス値を使用して、ブロックのための参照ピクチャがブロックを含むピクチャであることを合図してもよい。ビデオエンコーダ20は、イントラBCモードを使用してコーディングされるブロックの動き情報内にこのインデックス値を含めてもよい。いくつかの例では、ビデオエンコーダ20のハードウェアアーキテクチャは、現在のピクチャの現在のブロックを予測するために、参照ピクチャとして現在のピクチャを使用するために特に適合されなくてもよい。

【0107】

図5は、本開示に記載の技法を実装することができるビデオデコーダ30の一例を示すブロック図である。再び、ビデオデコーダ30を、他のコーディング規格に関する本開示の限定なしで、説明の目的のためのHEVCコーディングの文脈で説明する。さらに、ビデオデコーダ30は、範囲拡張に従って技法を実装するように構成されてもよい。

【0108】

図5の例では、ビデオデコーダ30は、ビデオデータメモリ69と、エントロピー復号ユニット70と、予測処理ユニット71と、逆量子化処理ユニット76と、逆変換処理ユニット78と、加算器80と、参照ピクチャメモリ82とを含んでもよい。予測処理ユニット71は、動き補償ユニット72と、イントラ予測ユニット74とを含む。ビデオデコーダ30は、いくつかの例では、図3からのビデオエンコーダ20に関連して説明した符号化パスと全体的に相互的な復号パスを実行してもよい。

【0109】

ビデオデータメモリ69は、ビデオデコーダ30の構成要素によって復号されるべき、符号化ビデオビットストリームなどのビデオデータを記憶してもよい。ビデオデータメモリ69内に記憶されたビデオデータは、たとえば、記憶デバイス34から、カメラなどのローカルビデオソースから、ビデオデータのワイヤードもしくはワイヤレスネットワーク通信を介して、または物理データ記憶媒体にアクセスすることによって取得されてもよい。ビデオデータメモリ69は、符号化ビデオビットストリームからの符号化ビデオデータを記憶するコーディングピクチャバッファ(CPB:coded picture buffer)を形成してもよい。

【0110】

10

20

30

40

50

参照ピクチャメモリ82は、(たとえば、イントラまたはインターコーディングモードにおいて)ビデオデコーダ30によってビデオデータを復号する際に使用するための参照ビデオデータを記憶する復号ピクチャバッファ(DPB:decoded picture buffer)の一例である。ビデオデータメモリ69および参照ピクチャメモリ82は、シンクロナスDRAM(SDRAM)を含むダイナミックランダムアクセスメモリ(DRAM)、磁気抵抗性RAM(MRAM)、抵抗性RAM(RRAM(登録商標))、または他のタイプ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てもよい。ビデオデータメモリ69および参照ピクチャメモリ82は、同じメモリデバイスまたは別個のメモリデバイスによって提供されてもよい。様々な例では、ビデオデータメモリ69は、ビデオデコーダ30の他の構成要素とともにオンチップであってもよく、または、これらの構成要素に対してオフチップであってもよい。

10

【0111】

復号プロセスの間、ビデオデコーダ30は、ビデオエンコーダ20から、符号化ビデオスライスのビデオブロックと関連する構文要素とを表す符号化されたビデオビットストリームを受信する。ビデオデコーダ30のエントロピー復号ユニット70は、量子化係数、動きベクトルまたはイントラ予測モードインジケータ、および他の構文要素を生成するために、ビットストリームをエントロピー復号する。エントロピー復号ユニット70は、動きベクトルおよび他の構文要素を動き補償ユニット72に転送する。ビデオデコーダ30は、ビデオスライスレベルおよび/またはビデオブロックレベルにおいて構文要素を受信してもよい。

【0112】

いくつかの例では、ビデオスライスがイントラコーディング(I)スライスとしてコーディングされているとき、イントラ予測ユニット74は、合図されたイントラ予測モードと、現在のピクチャの以前に復号されたブロックからのデータとに基づいて、現在のビデオスライスのビデオブロックのための予測データを生成してもよい。いくつかの例では、ビデオピクチャがインターコーディング(すなわち、BまたはP)スライスとしてコーディングされているとき、動き補償ユニット72は、エントロピー復号ユニット70から受信した動きベクトルおよび他の構文要素に基づいて、現在のビデオスライスのビデオブロックのための予測ブロックを生成する。予測ブロックは、参照ピクチャリスト(RPL)のうちの1つの中の参照ピクチャのうちの1つから生成されてもよい。予測処理ユニット71は、参照ピクチャメモリ82内に記憶された参照ピクチャに基づいて、構成技法を使用して、RPL、たとえば、List0およびList1を構成してもよい。

20

30

【0113】

本開示の1つまたは複数の技法によれば、参照ピクチャとして他のピクチャを使用することにインター予測を限定するのとは対照的に、ビデオデコーダ30は、現在のピクチャ内に含まれたビデオデータのブロックを予測するために、参照ピクチャとして現在のピクチャを使用してもよい。たとえば、予測処理ユニット71は、予測処理ユニット71内に現在のピクチャのバージョンを記憶してもよい。いくつかの例では、予測処理ユニット71は、固定値に初期化されたピクセル値を有する現在のフレームの初期化バージョンを記憶してもよい。いくつかの例では、固定値は、現在のピクチャのサンプルのビット深度に基づいてもよい。たとえば、固定値は、 $1 \ll (\text{bitDepth} - 1)$ であってもよい。いくつかの例では、予測処理ユニット71は、現在のピクチャの任意のブロックを符号化する前に、現在のピクチャの初期化バージョンを記憶してもよい。現在のピクチャの初期化バージョンを記憶することによって、予測処理ユニット71は、まだ再構成されていない予測ブロックを使用してもよい。対照的に、予測処理ユニット71が現在のピクチャの初期化バージョンを記憶していない場合、すでに再構成されたブロックのみが、(すなわち、デコーダ/エンコーダのミスマッチを回避するために)予測子ブロックとして使用されてもよい。

40

【0114】

上記で説明したように、予測処理ユニット71は、現在のピクチャのための1つまたは複数のRPLを生成してもよい。たとえば、予測処理ユニット71は、現在のピクチャのためのRPLに現在のピクチャを含めてもよい。いくつかの例では、予測処理ユニット71は、RPL内の特定の位置に現在のピクチャを含めてもよい。一例として、予測処理ユニット71は、長

50

期RPS内のピクチャの前にRPL内に現在のピクチャを挿入してもよい。たとえば、予測処理ユニット71は、長期RPSからのピクチャのインデックス値よりも小さいインデックス値を用いてRPL内に現在のピクチャを挿入してもよい。いくつかの例では、予測処理ユニット71は、長期RPS内のピクチャの直前にRPL内に現在のピクチャを挿入してもよい。

【0115】

別の例として、予測処理ユニット71は、長期RPSからのピクチャを挿入した後に、RPL内に現在のピクチャを挿入してもよい。たとえば、予測処理ユニット71は、長期RPS内のピクチャのインデックス値よりも大きいインデックス値を用いてRPL内に現在のピクチャを挿入してもよい。いくつかの例では、予測処理ユニット71は、長期RPSからのピクチャを挿入した直後に、RPL内に現在のピクチャを挿入してもよい。

10

【0116】

別の例として、予測処理ユニット71は、固定された位置においてRPL内に現在のピクチャを挿入してもよい。たとえば、予測処理ユニット71は、固定されたインデックス値を用いてRPL内に現在のピクチャを挿入してもよい。いくつかの例では、固定されたインデックス値は、-1、またはnum_ref_idx_l1_active_minus1+1であってもよい。そのような例のいくつかでは予測処理ユニット71は、エントロピー復号ユニット70から、現在のブロックがイントラBCを使用してコーディングされたことを示すフラグ(すなわち、intra_bc_flag)を受信しなくてもよい。

【0117】

動きベクトルが時間的動きベクトル予測(TMVP: temporal motion vector prediction)を使用して予測される場合などのいくつかの例では、予測処理ユニット71は、現在のピクチャがそれ自体のための並置されたピクチャとして使用されないように、1つまたは複数の制約を適用してもよい。たとえば、予測処理ユニット71は、Xがcollocated_from_l0_flagに等しいとき、RefPicListX[collocated_ref_idx]が現在のピクチャではないように、TMVP(たとえば、collocated_ref_idx)のために使用される並置されたピクチャの参照インデックスを指定する構文要素を、エントロピー復号ユニット70から受信してもよい。

20

【0118】

上記で説明したように、ビデオデコーダ30は、予測ブロックに基づいて、ビデオデータの現在のピクチャのビデオデータのブロックを復号してもよい。本開示の1つまたは複数の技法によれば、動き補償ユニット72は、現在のピクチャの現在のブロックのための予測ブロックとして使用するために、現在のピクチャ内に位置するブロックを選択してもよい。具体的には、予測処理ユニット71は、現在のブロックのために、現在のピクチャを含むRPLを構成してもよく、動き補償ユニット72は、RPL内のインデックスを示す現在のブロックのための動きパラメータを受信してもよい。いくつかの例では、インデックスは、RPL内の現在のピクチャを識別してもよい。これが生じたとき、動き補償ユニット72は、現在のブロックに対する動きベクトルによって識別された位置において現在のピクチャ自体から予測ブロックを抽出するために、動きパラメータ内に含まれる動きベクトルを使用してもよい。このようにして、動き補償ユニット72は、イントラBCを実行してもよい。

30

【0119】

いくつかの例では、予測処理ユニット71は、現在のピクチャが現在のピクチャを予測するために使用されるRPL内に存在し得るかどうかを示すために、エントロピー復号ユニット70から1つまたは複数の構文要素を受信してもよい。一例として、予測処理ユニット71は、ビデオデータのピクチャがそれら自体を予測するために使用されるRPL内に存在し得るかどうかを示す単一の構文要素を受信してもよい。いくつかの例では、予測処理ユニット71は、現在のピクチャによって参照されるビデオパラメータセット(VPS)、現在のピクチャによって参照されるシーケンスパラメータセット(SPS)、または現在のピクチャによって参照されるピクチャパラメータセット(PPS)からの単一の構文要素を受信してもよい。

40

【0120】

別の例として、予測処理ユニット71は、現在のピクチャが現在のピクチャを予測するた

50

めに使用されるRPL内に存在し得るかどうかを示すために、エントロピー復号ユニット70から複数の構文要素を受信してもよい。たとえば、予測処理ユニット71は、ビデオデータのピクチャがそれら自体を予測するために使用されるRPL内に存在し得るかどうかを示す第1の構文要素を受信してもよい。いくつかの例では、予測処理ユニット71は、現在のピクチャによって参照されるVPS、現在のピクチャによって参照されるSPS、または現在のピクチャによって参照されるPPSからの第1の構文要素を受信してもよい。構文要素が、ビデオデータのピクチャがそれら自体を予測するために使用されるRPL内に存在し得ることを示す場合などの、いくつかの例では、予測処理ユニット71は、ビデオデータの現在のピクチャが現在のスライスを予測するために使用されるRPL内に存在し得るかどうかを示す第2の構文要素を受信してもよい。いくつかの例では、予測処理ユニット71は、現在のスライスのスライスヘッダから第2の構文要素を受信してもよい。

10

【0121】

いくつかの例では、予測処理ユニット71は、ブロックがイントラブロックコピー(イントラBC)を使用してコーディングされているかどうかを示す構文要素を受信しなくてもよい。たとえば、予測処理ユニット71は、本開示の技法に従ってイントラBCを使用して予測されるブロックのコーディング単位構文内のintra_bc_flagを受信しなくてもよい。

【0122】

いくつかの例では、イントラBCを使用してBスライスおよびPスライス内のブロックをコーディングすることに加えて、予測処理ユニット71は、現在のピクチャのIスライスのブロックを復号するために現在のピクチャを含む1つまたは複数のRPLを構成してもよい。いくつかのそのような例では、1つまたは複数のRPLが現在のピクチャのみを含むと仮定してもよい。いくつかの例では、予測処理ユニット71は、現在のピクチャが現在のピクチャ内に含まれるIスライスの参照ピクチャとして使用され得るかどうかを示す構文要素を受信してもよい。いくつかの例では、予測処理ユニット71は、現在のピクチャによって参照されるVPS、現在のピクチャによって参照されるSPS、現在のピクチャによって参照されるPPS、または現在のIスライスのスライスヘッダからの構文要素を復号してもよい。いくつかの例では、予測処理ユニット71は、AMVPとマージのうちの一方または両方を依然として使用してもよい。いくつかの例では、予測処理ユニット71は、エンコーダからIスライスのためのAMVPのための目標参照インデックスを受信しなくてもよく、固定値、たとえばゼロとして目標参照インデックスを導出してもよい。いくつかの例では、予測処理ユニット71は、IスライスのためのAMVPのための目標参照インデックスを受信してもよいが、目標参照インデックスの値は、固定値、たとえば0であるように制限されてもよい。

20

30

【0123】

参照ピクチャメモリ82によって記憶されたピクチャは、短期、長期、別のマーキングとしてマークされてもよく、および/またはマークされなくてもよい。現在のスライスがIスライスであり、イントラBCが有効である場合などのいくつかの例では、予測処理ユニット71は、現在のピクチャを、長期または短期のいずれかとしてマークしてもよい。現在のスライスがIスライスであり、イントラBCが有効である場合などのいくつかの例では、予測処理ユニット71は、現在のピクチャを、長期または短期のいずれかとしてマークしなくてもよい。

40

【0124】

いくつかの例では、予測処理ユニット71は、現在のピクチャを復号する前に、現在のピクチャを長期としてマークしてもよく、現在のピクチャを復号した後に、現在のピクチャを短期としてマークしてもよい。現在のスライスがIスライスである場合などの、そのような例のいくつかでは、予測処理ユニット71は、(「長期」としてマークされた)イントラBC参照を純粋に指す候補と、(「短期」としてマークされた)インター参照を指す他の候補とを含むように、マージ候補リストを生成してもよい。このようにして、予測処理ユニット71は、イントラBC候補と通常の(インター予測)候補の両方を含むように候補リストを生成してもよい。いくつかの例では、予測処理ユニット71は、イントラBC参照およびインター予測参照からマージ候補を双方向予測してもよい。現在のスライスがインターコーディ

50

ングスライス(たとえば、BスライスまたはPスライス)である場合などの、そのような例のいくつかでは、予測処理ユニット71は、現在のブロックがイントラBCでコーディングされているかどうかを示すフラグ(intra_bc_flag)を示す追加のフラグを受信しなくてもよい。

【0125】

時間的動きベクトル予測(TMVP)が有効であり、現在のフレームが追加の参照ピクチャであると仮定することによってイントラBCが有効である場合などのいくつかの例では、予測処理ユニット71は、目標マージ参照ピクチャが現在のピクチャまたはRPL内の第1のピクチャのどちらであるのか(たとえば、RefPicList0[0]またはRefPicList1[0]のどちらであるのか)を示す構文要素を受信してもよい。いくつかの例では、予測処理ユニット71は、参照の長期(イントラBC)カテゴリまたは参照の短期カテゴリのための目標インデックスを導出または受信し、同一位置のブロックの参照ピクチャのカテゴリに基づいて異なる目標マージインデックスを適用することによって、参照ピクチャを識別してもよい。たとえば、予測処理ユニット71は、参照の長期(イントラBC)カテゴリまたは参照の短期カテゴリのための目標インデックスを合図または導出することによって参照ピクチャを識別してもよく、同一位置のブロックの参照ピクチャのカテゴリに基づいて、異なる目標マージインデックスが適用される。

【0126】

予測処理ユニット71は、ビデオデータの現在のブロックとビデオデータの予測子ブロックとの間の変位を表す動きベクトルを決定してもよい。いくつかの例では、予測処理ユニット71は、符号化ビデオビットストリームにおいて受信された1つまたは複数の構文要素に基づいて動きベクトルを決定してもよい。いくつかの例では、予測処理ユニット71は、整数精度の動きベクトルを決定してもよい。現在のピクチャが長期参照ピクチャとしてマークされている場合などの、そのような例では、予測処理ユニット71は、現在のピクチャを予測するために通常の長期参照ピクチャ(すなわち、現在のピクチャではない長期参照ピクチャ)を使用しなくてもよい。また、そのような例では、予測処理ユニット71は、たとえば、イントラBC動きベクトルが空間的および時間的に隣接するブロックのイントラBC動きベクトルのみに基づいて予測される場合、どのような変更もなしに、高度動きベクトル予測(AMVP)またはHEVCバージョン1におけるようなマージ関連復号プロセスを利用して

【0127】

いくつかの例では、予測処理ユニット71は、異なるレベルの精度で動きベクトルを決定してもよい。たとえば、予測処理ユニット71は、整数精度、デフォルトの精度、または最も詳細な動き精度(たとえば、HEVCにおける1/4pel精度)で動きベクトルを決定してもよい。いくつかの例では、予測処理ユニット71は、たとえば、現在のピクチャによって参照されるSPSまたはVPS内で、コーディングされたイントラBC動きベクトルの精度を示す構文要素を受信してもよい。いくつかの例では、イントラBC動きベクトルの精度は、ピクチャレベルで適応されてもよく、予測処理ユニット71は、たとえば、現在のブロックによって参照されるPPSまたはスライスにおいて、コーディングされたイントラBC動きベクトルの精度を示す構文要素を受信してもよい。

【0128】

いくつかの例では、予測処理ユニット71は、イントラBC動きベクトルの精度レベルを補償するために、1つまたは複数の動作を実行してもよい。一例として、ブロックを参照ピクチャメモリ82内に記憶する前に、予測処理ユニット71は、最高精度が1/4pelであるとき、2などだけ、または、任意の他の丸め手段によって、左シフト後に±2などだけ、イントラBCでコーディングされた各ブロックの動きベクトルを左シフトしてもよい。別の例として、整数の精度を有するイントラBC動きベクトルで現在のスライスをコーディングするとき、予測処理ユニット71は、最高精度が1/4pelであるとき、2などだけ、各イントラBCコーディングブロックの動きベクトルが右シフトされるように、同一位置のピクチャを処理してもよい。現在のスライスが最高の動き精度を有するイントラBC動きベクトルでコーデ

イングされるときなどの、いくつかの例では、予測処理ユニット71は、上記の動きベクトル右シフトの処理を適用しなくてもよい。

【0129】

現在のピクチャが長期参照ピクチャとしてマークされているいくつかの例では、予測処理ユニット71は、現在のピクチャを予測するために、通常の長期参照ピクチャを依然として使用してもよい。通常の長期参照ピクチャを参照する動きベクトルおよび現在のピクチャを参照する動きベクトルが、マージまたはAMVPの間、互いから予測されるのを防止するために、予測処理ユニット71は、マージまたはAMVPの間、通常の長期参照ピクチャと現在のピクチャとを区別してもよい。

【0130】

動き補償ユニット72は、動きベクトルおよび他の構文要素を構文解析することによって、現在のビデオスライスのビデオブロックのための予測情報を決定し、復号されている現在のブロックのための予測ブロックを生成するために予測情報を使用する。たとえば、動き補償ユニット72は、ビデオスライスのビデオブロックをコーディングするために使用される予測モード(たとえば、イントラまたはインター予測)と、インター予測スライスタイプ(たとえば、BスライスまたはPスライス)と、スライスのための1つまたは複数の参照ピクチャリストのための構成情報と、スライスの各インター符号化ビデオブロックのための動きベクトルと、スライスの各インターコーディングビデオブロックのためのインター予測状態と、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とを決定するために、受信した構文要素のうちのいくつかを使用する。

【0131】

動き補償ユニット72はまた、補間フィルタに基づいて補間を実行してもよい。動き補償ユニット72は、参照ブロックのサブ整数ピクセルに関する補間値を計算するために、ビデオブロックの符号化の間にビデオエンコーダ20によって使用されるように補間フィルタを使用してもよい。この場合には、動き補償ユニット72は、受信した構文要素からビデオエンコーダ20によって使用された補間フィルタを決定し、予測ブロックを生成するために補間フィルタを使用してもよい。

【0132】

逆量子化処理ユニット76は、ビットストリーム内で提供され、エントロピー復号ユニット70によって復号された量子化変換係数を逆量子化する(inverse quantize)、すなわち逆量子化する(de-quantize)。逆量子化プロセスは、量子化の程度と、同様に、適用されるべき逆量子化の程度とを決定するために、ビデオスライス内の各ビデオブロックのための、ビデオデコーダ30によって計算された量子化パラメータ QP_Y の使用を含んでもよい。

【0133】

逆変換処理ユニット78は、ピクセル領域における残差ブロックを生成するために、変換係数に逆変換、たとえば、逆DCT、逆整数変換、または概念的に同様の逆変換プロセスを適用する。ビデオデコーダ30は、逆変換プロセスユニット78からの残差ブロックを動き補償ユニット72によって生成された対応する予測ブロックと加算することによって、復号ビデオブロックを形成する。加算器80は、この加算演算を実行する1つまたは複数の成分を表す。

【0134】

ビデオデコーダ30は、フィルタリングユニットを含んでもよく、フィルタリングユニットは、いくつかの例では、上記で説明したビデオエンコーダ20のフィルタリングユニットと同様に構成されてもよい。たとえば、フィルタリングユニットは、符号化ビットストリームからのビデオデータを復号し、再構成するとき、デブロッキング、SAO、または他のフィルタリング動作を実行するように構成されてもよい。

【0135】

技法のいくつかの異なる態様および例が本開示で説明されているが、技法の様々な態様および例は、互いに一緒にまたは別々に実行されてもよい。言い換えれば、技法は、上記で説明した様々な態様および例に厳密に限定されるべきではないが、組み合わせて使用さ

10

20

30

40

50

れてもよいし、または一緒におよび/もしくは別々に実行されてもよい。加えて、特定の技法は、ビデオデコーダ30の特定のユニットに起因することがあるが、ビデオデコーダ30の1つまたは複数の他のユニットはまた、そのような技法を実行することを担当してもよいことが理解されるべきである。

【0136】

このようにして、ビデオデコーダ30は、本開示に記載の1つまたは複数の例示的な技法を実装するように構成されてもよい。たとえば、ビデオデコーダ30は、たとえば、イントラBCモードを使用して現在のピクチャの1つまたは複数のブロックをコーディングする目的のため、PPSを参照するピクチャがピクチャ自体の参照ピクチャリスト内に存在し得るかどうかを示す構文要素を含むビットストリームを受信するように構成されてもよい。すなわち、ビデオデコーダ30は、現在のピクチャがそれ自体のための参照ピクチャリスト内に含まれ得ることを示す構文要素のための値を復号してもよい。したがって、ブロックがイントラBCモードを使用してコーディングされるとき、ビデオデコーダ30は、たとえば、インデックス値がピクチャ自体に対応するように参照ピクチャリスト内へのインデックス値を使用して、ブロックのための参照ピクチャがブロックを含むピクチャであることを決定してもよい。ビデオデコーダ30は、イントラBCモードを使用してコーディングされるブロックの動き情報内からこのインデックス値を復号してもよい。いくつかの例では、ビデオデコーダ30のハードウェアアーキテクチャは、現在のピクチャの現在のブロックを予測するために、参照ピクチャとして現在のピクチャを使用するために特に適合されなくてもよい。

【0137】

図6は、本開示の1つまたは複数の技法による、イントラブロックコピーイングプロセスの一例を示す図である。1つの例示的なイントラ予測プロセスによれば、ビデオエンコーダ20は、たとえば、以前にコーディングされ、再構成されたビデオデータのブロックのセットから、予測子ビデオブロックを選択してもよい。図6の例では、再構成された領域108は、以前にコーディングされ、再構成されたビデオブロックのセットを含む。再構成された領域108内のブロックは、ビデオデコーダ30によって復号され、再構成され、再構成領域メモリ92内に記憶されているブロック、または、ビデオエンコーダ20の再構成ループ内で復号および再構成され、再構成メモリ領域64内に記憶されているブロックを表してもよい。現在のブロック102は、コーディングされるべきビデオデータの現在のブロックを表す。予測子ブロック104は、現在のブロック102のイントラBC予測のために使用される、現在のブロック102と同じピクチャ内の再構成されたビデオブロックを表す。

【0138】

例示的なイントラ予測プロセスでは、ビデオエンコーダ20は、動きベクトル106を決定し、符号化してもよく、動きベクトル106は、残差信号と一緒に、現在のブロック102に対する予測子ブロック104の位置を示す。たとえば、図6によって示すように、動きベクトル106は、現在のブロック102の左上隅に対する予測子ブロック104の左上隅の位置を示してもよい。上記で説明したように、動きベクトル106はまた、オフセットベクトル、変位ベクトル、またはブロックベクトル(BV)と呼ばれることもある。ビデオデコーダ30は、現在のブロックを復号するための符号化された情報を利用してもよい。

【0139】

1つの例示的な例として、現在のピクチャを復号する前に、ビデオデコーダ30は、現在のピクチャの再構成されたサンプルを $1 \ll (\text{bitDepth} - 1)$ に初期化してもよい。ビデオデコーダ30は、次いで、参照ピクチャメモリ82などの参照ピクチャバッファ内に現在のピクチャのバージョンを記憶してもよく、長期参照として現在のピクチャをマークしてもよい。ビデオデコーダ30は、次いで、参照ピクチャリスト(RPL)内に現在のピクチャを含めてもよく、RPL内の現在のピクチャのための参照インデックス(たとえば、(参照リストListX内の)IdxCur)を割り当ててもよい。

【0140】

ビデオデコーダ30は、RPLに基づいて、参照ピクチャメモリ82内に記憶された現在のピ

クチャのバージョン内に含まれる予測子ブロックに基づいて、現在のピクチャ内のビデオデータのブロックを復号してもよい。言い換えれば、現在のピクチャのブロックを復号するとき、ビデオデコーダ30は、現在のピクチャ、すなわち、参照インデックスIdxCur(ListX内)による参照からブロックを予測してもよい。この場合、ビデオデコーダ30は、HEVCバージョン1におけるものと同じMVコーディング方法を利用してもよい。いくつかの例では、たとえば、ビデオデコーダは固定値で現在のピクチャを初期化したので、ビデオデコーダ30は、MVの値を制限しなくてもよい。ビデオデコーダ30は、(たとえば、ビデオデコーダがブロックの復号を終了した後)初期化された値を置換するために、現在のピクチャのバッファ(たとえば、参照ピクチャメモリ82)にブロックの再構成されたサンプルを書き込んでもよい。この例では、ビデオデコーダ30は、ブロックを復号した後、再構成されたサンプルにデブロッキング、SAO、またはなにか他のフィルタリング動作を適用しないことに留意されたい。言い換えれば、ビデオデコーダ30は、デブロッキングおよびSAOを適用する前に、参照として現在のピクチャを使用してもよい。

10

【0141】

ピクチャ全体をコーディングした後、ビデオデコーダ30は、デブロッキング、SAO、および、HEVCバージョン1に記載のものと同様の方法におけるピクチャマーキングなどの他の動作を適用してもよい。いくつかの例では、ビデオデコーダ30は、現在のピクチャ内のブロックを参照するとき、MVの精度を従来の参照ピクチャと同じ(たとえば、1/4ピクセル精度)に保ってもよい。そのような例では、ビデオデコーダ30は、HEVCバージョン1において定義された補間フィルタを使用してもよい。いくつかの例では、ビデオデコーダ30は、(HEVCバージョン1において定義された補間フィルタに加えて、またはその代わりに)双線形補間フィルタなどの他の補間フィルタを使用してもよい。MVが現在のピクチャ内のブロックを参照しているときなどの、いくつかの例では、ビデオデコーダ30は、MVの精度を整数ピクセルに制限してもよい。いくつかの例では、ビデオデコーダ30は、上記で説明したように、現在のピクチャを含む1つまたは複数の参照ピクチャ管理技法を実行してもよい。

20

【0142】

上記で説明したように、いくつかの例では、ビデオエンコーダ20は、現在のピクチャが現在のピクチャを予測するために使用されるRPL内に存在し得るのかどうかを示すために、1つまたは複数の構文要素を符号化し、構文要素を合図してもよい。たとえば、ビデオデコーダ30は、PPSを参照するピクチャがピクチャ自体の参照ピクチャリスト内に存在し得るのかどうかをビデオデコーダ30に示すためにcurr_pic_as_ref_enabled_flagを合図してもよい。一例として、ビデオエンコーダ20は、PPSを参照するピクチャがピクチャ自体の参照ピクチャリスト内に存在し得ることを示すために、1に等しいようにcurr_pic_as_ref_enabled_flagを合図してもよい。別の例として、ビデオエンコーダ20は、PPSを参照するピクチャがピクチャ自体の参照ピクチャリスト内に存在しないことを示すために、0に等しいようにcurr_pic_as_ref_enabled_flagを合図してもよい。さらに別の例として、ビデオエンコーダ20は、curr_pic_as_ref_enabled_flagを合図しなくてもよい。いくつかの例では、構文要素curr_pic_as_ref_enabled_flagが存在しないとき、ビデオデコーダ30は、curr_pic_as_ref_enabled_flagの値が0に等しくなるように推定してもよい(すなわち、明示的な指示または合図なしで決定してもよい)。いくつかの例では、ビデオエンコーダ20は、可変NumAddRefPicを(curr_pic_as_ref_enabled_flag?1:0)に等しく設定してもよい。以下の表では、curr_pic_as_ref_enabledは、この構文要素がピクチャパラメータセットに加えられていることを示すために、斜体のテキストで示されている。他の例では、この構文要素は、追加で、または代替的に、他のデータ構造、たとえば、シーケンスパラメータセット(SPS)、ビデオパラメータセット(VPS)、スライスヘッダ、などに加えられてもよい。

30

40

【0143】

【表 1 A】

pic_parameter_set_rbsp() {	記述子
pps_pic_parameter_set_id	ue(v)
pps_seq_parameter_set_id	ue(v)
dependent_slice_segments_enabled_flag	u(1)
output_flag_present_flag	u(1)
num_extra_slice_header_bits	u(3)
sign_data_hiding_enabled_flag	u(1)
cabac_init_present_flag	u(1)
num_ref_idx_l0_default_active_minus1	ue(v)
num_ref_idx_l1_default_active_minus1	ue(v)
init_qp_minus26	se(v)
constrained_intra_pred_flag	u(1)
transform_skip_enabled_flag	u(1)
cu_qp_delta_enabled_flag	u(1)
if(cu_qp_delta_enabled_flag)	
diff_cu_qp_delta_depth	ue(v)
pps_cb_qp_offset	se(v)
pps_cr_qp_offset	se(v)
pps_slice_chroma_qp_offsets_present_flag	u(1)
weighted_pred_flag	u(1)
weighted_bipred_flag	u(1)
transquant_bypass_enabled_flag	u(1)
tiles_enabled_flag	u(1)
entropy_coding_sync_enabled_flag	u(1)
if(tiles_enabled_flag) {	
num_tile_columns_minus1	ue(v)
num_tile_rows_minus1	ue(v)
uniform_spacing_flag	u(1)
if(!uniform_spacing_flag) {	
for(i = 0; i < num_tile_columns_minus1; i++)	
column_width_minus1[i]	ue(v)
for(i = 0; i < num_tile_rows_minus1; i++)	
row_height_minus1[i]	ue(v)
}	
loop_filter_across_tiles_enabled_flag	u(1)
}	
pps_loop_filter_across_slices_enabled_flag	u(1)
deblocking_filter_control_present_flag	u(1)

【表 1 B】

if(deblocking_filter_control_present_flag) {	
deblocking_filter_override_enabled_flag	u(1)
pps_deblocking_filter_disabled_flag	u(1)
if(!pps_deblocking_filter_disabled_flag) {	
pps_beta_offset_div2	se(v)
pps_tc_offset_div2	se(v)
}	
}	
pps_scaling_list_data_present_flag	u(1)
if(pps_scaling_list_data_present_flag)	
scaling_list_data()	
lists_modification_present_flag	u(1)
log2_parallel_merge_level_minus2	ue(v)
slice_segment_header_extension_present_flag	u(1)
pps_extension_present_flag	u(1)
if(pps_extension_present_flag) {	
for(i = 0; i < 1; i++)	
pps_extension_flag[i]	u(1)
pps_extension_7bits	u(7)
}	
if(pps_extension_flag[0]) {	
if(transform_skip_enabled_flag)	
log2_max_transform_skip_block_size_minus2	ue(v)
cross_component_prediction_enabled_flag	u(1)
chroma_qp_adjustment_enabled_flag	u(1)
if(chroma_qp_adjustment_enabled_flag) {	
diff_cu_chroma_qp_adjustment_depth	ue(v)
chroma_qp_adjustment_table_size_minus1	ue(v)
for(i = 0; i <= chroma_qp_adjustment_table_size_minus1; i++) {	
cb_qp_adjustment[i]	se(v)
cr_qp_adjustment[i]	se(v)
}	
}	
}	
log2_sao_offset_scale_luma	ue(v)
log2_sao_offset_scale_chroma	ue(v)
curr_pic_as_ref_enabled_flag	u(1)
}	
if(pps_extension_7bits)	

10

20

30

40

【表 1 C】

while(more_rbsp_data())	
pps_extension_data_flag	u(1)
rbsp_trailing_bits()	
}	

【 0 1 4 6 】

上記で説明したように、ビデオデコーダ30は、現在のピクチャを含んでもよい1つまたは複数の参照ピクチャリストを構成してもよい。たとえば、いくつかの例では、ビデオデコーダ30は、各インターコーディングスライスのための復号プロセスの開始時に、以下のプロセスを呼び出してもよい。Pスライスを復号するときなどのいくつかの例では、ビデオデコーダ30は、単一の参照ピクチャリストRefPicList0を構成してもよい。Bスライスを復号するときなどのいくつかの例では、ビデオデコーダ30は、RefPicList0に加えて、第2の独立した参照ピクチャリストRefPicList1をさらに構成してもよい。

10

【 0 1 4 7 】

各スライスのための復号処理の開始時に、ビデオデコーダ30は、参照ピクチャリストRefPicList0を導出してもよく、Bスライスのために、以下のようにRefPicList1を導出してもよい(ここで、斜体テキストは、標準の現在のセマンティクスに対する追加を表す)。

20

【 0 1 4 8 】

ビデオデコーダ30は、可変NumRpsCurrTempList0をMax(num_ref_idx_l0_active_minus1+1, NumPocTotalCurr+NumAddRefPic)に等しいように設定してもよく、リストRefPicListTemp0を以下のように構成してもよい。ここで、currPicは、現在のピクチャである。

rIdx=0

while(rIdx<NumRpsCurrTempList0){

 for(i=0; i<NumPocStCurrBefore&&rIdx<NumRpsCurrTempList0; rIdx++, i++)

 RefPicListTemp0[rIdx]=RefPicSetStCurrBefore[i]

 if(curr_pic_as_ref_enabled_flag)

 RefPicListTemp0[rIdx]=currPic

30

 for(i=0; i<NumPocStCurrAfter&&rIdx<NumRpsCurrTempList0; rIdx++, i++)

 RefPicListTemp0[rIdx]=RefPicSetStCurrAfter[i]

 for(i=0; i<NumPocLtCurr&&rIdx<NumRpsCurrTempList0; rIdx++, i++)

 RefPicListTemp0[rIdx]=RefPicSetLtCurr[i]

}

【 0 1 4 9 】

ビデオデコーダ30は、以下のようにリストRefPicList0を構成してもよく、ここで、currPicは、現在のピクチャである。

for(rIdx=0; rIdx <= num_ref_idx_l0_active_minus1; rIdx++)

 RefPicList0[rIdx]=ref_pic_list_modification_flag_l0?RefPicListTemp0[list_entry_l0[rIdx]]: RefPicListTemp0[rIdx]

40

【 0 1 5 0 】

スライスがBスライスであるときなどのいくつかの例では、ビデオデコーダ30は、可変NumRpsCurrTempList1がMax(num_ref_idx_l1_active_minus1+1, NumPocTotalCurr+NumAddRefPic)に等しいように設定してもよく、リストRefPicListTemp1を以下のように構成してもよい。

rIdx=0

while(rIdx<NumRpsCurrTempList1){

 for(i=0; i<NumPocStCurrAfter&&rIdx<NumRpsCurrTempList1; rIdx++, i++)

 RefPicListTemp1[rIdx]=RefPicSetStCurrAfter[i]

50

```

    if(curr_pic_as_ref_enabled_flag)
        RefPicListTemp0[rIdx]=currPic
    for(i=0; i<NumPocStCurrBefore&&rIdx<NumRpsCurrTempList1;rIdx++,i++)
        RefPicListTemp1[rIdx]=RefPicSetStCurrBefore[i]
    for(i=0;i<NumPocLtCurr&&rIdx<NumRpsCurrTempList1;rIdx++,i++)
        RefPicListTemp1[rIdx]=RefPicSetLtCurr[i]
}

```

【0151】

スライスがBスライスであるときなどのいくつかの例では、ビデオデコーダ30は、以下のようにリストRefPicList1を構成してもよい。

```
for(rIdx=0;rIdx<=num_ref_idx_l1_active_minus1;rIdx++)
```

```
    RefPicList1[rIdx]=ref_pic_list_modification_flag_l1?RefPicListTemp1[list_entry_l1[rIdx]]:RefPicListTemp1[rIdx]
```

【0152】

図7は、本開示の1つまたは複数の技法による、同じピクチャ内に含まれる予測子ブロックに基づいてピクチャのビデオデータのブロックを符号化するためのビデオエンコーダの例示的な動作を示すフロー図である。図7の技法は、図1および図3に示すビデオエンコーダ20などの、1つまたは複数のビデオエンコーダによって実行されてもよい。ビデオエンコーダ20の構成とは異なる構成を有するビデオエンコーダは、図7の技法を実行してもよいが、例示の目的のため、図7の技法は、ビデオエンコーダ20の文脈内で説明される。

【0153】

本開示の1つまたは複数の技法によれば、ビデオエンコーダ20は、参照ピクチャバッファ内に、ビデオデータの現在のピクチャのバージョンを記憶してもよい(702)。たとえば、予測処理ユニット42は、参照ピクチャメモリ64内に現在のピクチャの初期化バージョンを記憶してもよい。

【0154】

ビデオエンコーダ20は、現在のブロックを選択してもよく(704)、現在のブロックのための予測モードを決定してもよい(706)。ビデオエンコーダ20は、たとえば、BDレートを使用して試験され得る、様々な異なるモードの中から予測モードを選択してもよく、ビデオエンコーダ20は、最良のBDレート性能をもたらすモードを選択してもよい。図7の例では、ビデオエンコーダ20は、イントラBCを使用して現在のブロックを符号化することを決定する。

【0155】

ビデオエンコーダ20は、現在のブロックのための予測子ブロックを決定してもよい(708)。たとえば、ビデオエンコーダ20は、ピクセル差の点から現在のブロックに密接に一致することが見出された現在のピクチャ内のブロックとして予測ブロックを決定してもよく、これは、絶対差合計(SAD)、自乗差合計(SSD)、または他の差分メトリックによって決定されてもよい。

【0156】

ビデオエンコーダ20は、現在のピクチャを予測するために使用される参照ピクチャリスト(RPL)内に現在のピクチャを挿入してもよい(710)。いくつかの例では、ビデオエンコーダ20は、ビデオデータのピクチャがそれ自体を予測するために使用されるRPL内に存在し得るかどうかを示す構文要素(たとえば、curr_pic_as_ref_enabled_flag)を符号化してもよい。上記で説明したように、いくつかの例では、ビデオエンコーダ20は、長期RPS内のピクチャのインデックス値よりも小さいインデックス値、長期RPS内のピクチャのインデックス値よりも大きいインデックス値、または固定されたインデックス値を用いて、RPL内に現在のピクチャを挿入してもよい。いくつかの例では、ビデオエンコーダ20は、現在のピクチャを予測するために使用されるRPLが現在のピクチャのみを含むように、現在のピクチャを予測するために使用されるRPLを構成してもよい。たとえば、現在のブロックが現在のイントラスライス(たとえば、Iスライス)内に含まれている場合、ビデオエンコ

10

20

30

40

50

ーダ20は、現在のイントラスライスのブロックを予測するために使用されるRPLが現在のピクチャのみを含むように、現在のイントラスライスのブロックを予測するために使用されるRPLを構成してもよい。いくつかの例では、ビデオエンコーダ20は、現在のピクチャを予測するために使用されるRPLが、現在のピクチャと、ビデオデータの1つまたは複数の他のピクチャとを含むように、現在のピクチャを予測するために使用されるRPLを構成してもよい。たとえば、現在のブロックが現在のインタースライス(たとえば、BスライスまたはPスライス)内に含まれている場合、ビデオエンコーダ20は、現在のイントラスライスのブロックを予測するために使用されるRPLが、現在のピクチャと、ビデオデータの1つまたは複数の他のピクチャとを含むように、現在のインタースライスのブロックを予測するために使用されるRPLを構成してもよい。

10

【0157】

ビデオエンコーダ20は、RPL内の現在のピクチャのインデックスを決定してもよい(712)。たとえば、現在のピクチャが固定されたインデックス値を用いてRPL内に挿入されている場合、ビデオエンコーダ20は、RPL内の現在のピクチャのインデックスが固定されたインデックス値であることを決定してもよい。

【0158】

ビデオエンコーダ20は、現在のブロックのための残差ブロックを計算してもよい(714)。たとえば、加算器50は、残差ブロックを計算するために、決定された予測子ブロックのサンプルから現在のブロックのサンプルを減算してもよい。

【0159】

ビデオエンコーダ20は、残差ブロックを量子化および変換してもよい(716)。たとえば、ビデオエンコーダ20の変換処理ユニット52は、離散コサイン変換(DCT)、または概念的に同様の変換などの変換を残差ブロックに適用し、残差変換係数値を備えるビデオブロックを生成してもよい。ビデオエンコーダ20の量子化処理ユニット54は、ビットレートをさらに低減するために、変換係数を量子化してもよい。

20

【0160】

ビデオエンコーダ20は、予測モードと、インデックスと、量子化変換係数とをエントロピー符号化してもよい(718)。たとえば、エントロピー符号化ユニット56は、現在のブロックの予測モードと、現在のブロックのための参照ピクチャのインデックス(RPL内の現在のブロックのインデックスであってもよい)と、量子化変換係数とをエントロピー符号化

30

【0161】

いくつかの例では、ビデオエンコーダ20は、図7に示す順序で例示的な動作を実行してもよい。いくつかの例では、ビデオエンコーダ20は、図7に示す順序以外の順序で例示的な動作を実行してもよい。たとえば、いくつかの例では、ビデオエンコーダ20は、現在のピクチャから現在のブロックを選択する(704)前に、現在のピクチャを予測するために使用されるRPL内に現在のピクチャを挿入してもよい(710)。たとえば、現在のブロックが現在のスライス内に含まれている場合、ビデオエンコーダ20は、現在のスライスの任意のブロックを符号化する前に、現在のピクチャを予測するために使用されるRPL内に現在のピクチャを挿入してもよい。

40

【0162】

図8は、本開示の1つまたは複数の技法による、同じピクチャ内に含まれる予測子ブロックに基づいてピクチャのビデオデータのブロックを復号するためのビデオデコーダの例示的な動作を示すフロー図である。図8の技法は、図1および図5に示すビデオデコーダ30などの、1つまたは複数のビデオデコーダによって実行されてもよい。ビデオデコーダ30の構成とは異なる構成を有するビデオデコーダは、図8の技法を実行してもよいが、例示の目的のため、図8の技法は、ビデオデコーダ30の文脈内で説明される。

【0163】

本開示の1つまたは複数の技法によれば、ビデオデコーダ30は、参照ピクチャバッファ内にビデオデータの現在のピクチャのバージョンを記憶してもよい(802)。たとえば、予

50

測処理ユニット71は、参照ピクチャメモリ82内に現在のピクチャの初期化バージョンを記憶してもよい。

【0164】

ビデオデコーダ30は、現在のピクチャから現在のブロックを選択してもよく(804)、予測モードと、参照ピクチャのインデックスと、現在のブロックのための量子化変換係数とをエントロピー復号してもよい(806)。たとえば、エントロピー復号ユニット70は、現在のブロックのための予測モードがイントラBCであることを示す1つまたは複数の構文要素と、現在のブロックを予測するために使用される参照ピクチャのインデックス(たとえば、ref_idx_lx)とをエントロピー復号してもよい。図8の例では、現在のブロックを予測するために使用される参照ピクチャは、現在のピクチャであってもよい。

10

【0165】

ビデオデコーダ30は、現在のピクチャを予測するために使用される参照ピクチャリスト(RPL)内に現在のピクチャを挿入してもよい(808)。いくつかの例では、ビデオデコーダ30は、1つまたは複数の構文要素の存在/値に基づいて、現在のピクチャをRPL内に挿入するかどうかを決定してもよい。たとえば、ビデオデコーダ30は、ビデオデータのピクチャがそれ自体を予測するために使用されるRPL内に存在し得るかどうかを示す構文要素(たとえば、curr_pic_as_ref_enabled_flag)の値に基づいて、現在のピクチャをRPL内に挿入するかどうかを決定してもよい。上記で説明したように、いくつかの例では、ビデオデコーダ30は、長期RPS内のピクチャのインデックス値よりも小さいインデックス値、長期RPS内のピクチャのインデックス値よりも大きいインデックス値、または固定されたインデックス値を用いて、現在のピクチャをRPL内に挿入してもよい。いくつかの例では、ビデオデコーダ30は、現在のピクチャを予測するために使用されるRPLが現在のピクチャのみを含むように、現在のピクチャを予測するために使用されるRPLを構成してもよい。いくつかの例では、ビデオデコーダ30は、現在のピクチャを予測するために使用されるRPLが、現在のピクチャと、ビデオデータの1つまたは複数の他のピクチャとを含むように、現在のピクチャを予測するために使用されるRPLを構成してもよい。

20

【0166】

ビデオデコーダ30は、現在のブロックのための予測子ブロックを決定してもよい(810)。たとえば、予測処理ユニット71は、RPL内の現在のピクチャのインデックスを参照し得る参照ピクチャのインデックスと、現在のブロックと予測子ブロックとの間の変位を示す動きベクトルとに基づいて、現在のブロックのための予測子ブロックを決定してもよい。

30

【0167】

ビデオデコーダ30は、残差ブロックを逆量子化および逆変換してもよい(812)。たとえば、逆量子化処理ユニット76は、エントロピー復号ユニット70によって復号された量子化変換係数を逆量子化(inverse quantize)、すなわち逆量子化(de-quantize)してもよい。逆変換処理ユニット78は、ピクセル領域における残差ブロックを生成するために、係数を変換するために、逆変換、たとえば、逆DCT、逆整数変換、または概念的に同様な逆変換プロセスを適用してもよい。

【0168】

ビデオデコーダ30は、現在のブロックを再構成してもよい(814)。たとえば、加算器80は、現在のブロックを再構成するために、予測子ブロックに残差ブロックを追加してもよい。ビデオデコーダ30は、現在のブロックを再構成した後、参照ピクチャバッファ内の現在のピクチャのバージョンを、コーディングされた現在のブロックを含む現在のピクチャの更新されたバージョンで更新してもよい。たとえば、加算器80は、たとえば、後続のブロックが予測子ブロックの一部またはすべてとして現在のブロックの1つまたは複数の再構成されたサンプルを使用することができるようにするために、参照ピクチャメモリ82内に現在のブロックの再構成されたサンプルを記憶してもよい。このようにして、ビデオデコーダ30は、イントラBCを実行してもよい。

40

【0169】

いくつかの例では、ビデオデコーダ30は、図8に示す順序で例示的な動作を実行しても

50

よい。いくつかの例では、ビデオデコーダ30は、図8に示す順序以外の順序で例示的な動作を実行してもよい。たとえば、いくつかの例では、ビデオデコーダ30は、現在のピクチャから現在のブロックを選択する(804)前に、現在のピクチャを予測するために使用されるRPL内に現在のピクチャを挿入してもよい(808)。たとえば、現在のブロックが現在のスライス内に含まれている場合、ビデオデコーダ30は、現在のスライスの任意のブロックを符号化する前に、現在のピクチャを予測するために使用されるRPL内に現在のピクチャを挿入してもよい。

【0170】

以下の番号付けされた例は、本開示の1つまたは複数の態様を例示することができる。

【0171】

1. ビデオデータを符号化または復号する方法であって、前記方法が、ビデオコードによって、参照ピクチャバッファ内に、前記ビデオデータの現在のピクチャのバージョンを記憶するステップと、前記現在のピクチャのブロックを予測する際に使用される参照ピクチャリスト(RPL)内に前記現在のピクチャの指標を挿入するステップと、ビデオコードによって、前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記バージョン内に含まれるビデオデータの予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第1のブロックをコーディングするステップとを含む方法。

【0172】

2. 前記ビデオコードによって、前記第1のブロックをコーディングした後に、前記参照ピクチャバッファ内の前記現在のピクチャの前記バージョンを、前記コーディングされた第1のブロックを含む前記現在のピクチャの更新されたバージョンで更新するステップと、前記ビデオコードによって、前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記更新されたバージョン内に含まれる予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第2のブロックをコーディングするステップとをさらに備える、請求項1に記載の方法。

【0173】

3. 前記ビデオコードによって、前記ビデオデータのピクチャが前記ピクチャ自体を予測するために使用されるRPL内に存在し得るのかどうかを示す構文要素をコーディングするステップと、前記構文要素に基づいて、前記ビデオデータの前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に含めることを決定するステップとをさらに備える、請求項1に記載の方法。

【0174】

4. 前記構文要素をコーディングするステップが、前記現在のピクチャによって参照されるビデオパラメータセット(VPS)、前記現在のピクチャによって参照されるシーケンスパラメータセット(SPS)、または前記現在のピクチャによって参照されるピクチャパラメータセット(PPS)内に前記構文要素をコーディングするステップを備える、請求項3に記載の方法。

【0175】

5. 前記構文要素が第1の構文要素であり、前記第1のブロックが前記現在のピクチャの現在のスライス内に含まれ、前記方法が、前記ビデオコードによって、前記ビデオデータのピクチャが前記ピクチャ自体を予測するために使用されるRPL内に存在し得ることを示す前記第1の構文要素に基づいて、前記ビデオデータの前記現在のピクチャが前記現在のスライスを予測するために使用される前記RPL内に存在し得るのかどうかを示す第2の構文要素をコーディングするステップをさらに備え、前記第2の構文要素が前記現在のスライスのヘッダ内に含まれ、前記ビデオデータの前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に挿入することの前記決定が、さらに、前記第2の構文要素に基づく、請求項4に記載の方法。

【0176】

6. 前記第2の構文要素をコーディングするステップが、前記現在のスライスを予測するために使用されるRPLを変更することに関連する他の構文要素の前に前記現在のスライス

10

20

30

40

50

のヘッダ内に前記第2の構文要素をコーディングするステップを備える、請求項5に記載の方法。

【0177】

7. 前記方法が、前記第1のブロックがイントラブロックコピー(イントラBC)を使用してコーディングされるかどうかを示す構文要素をコーディングするステップを含まない、請求項4に記載の方法。

【0178】

8. 前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記スライスのための並置参照インデックスを示す構文要素が前記現在のピクチャではないピクチャを示す、請求項1に記載の方法。

【0179】

9. 前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記予測子ブロックが前記現在のスライス内に含まれる、請求項1に記載の方法。

【0180】

10. 前記RPL内の各エントリがインデックス値を有し、前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に挿入するステップが、少なくとも、長期RPS内のピクチャのインデックス値よりも小さいインデックス値を用いて前記RPL内に前記現在のピクチャを挿入するか、長期RPS内のピクチャのインデックス値よりも大きいインデックス値を用いて前記RPL内に前記現在のピクチャを挿入するか、または固定されたインデックス値を用いて前記RPL内に前記現在のピクチャを挿入することによって、1つまたは複数の参照ピクチャセット(RPS)に基づいて前記現在のピクチャを予測するために使用される前記RPLを構成するステップを備える、請求項1に記載の方法。

【0181】

11. 前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスがイントラスライスであり、前記現在のピクチャを、前記現在のピクチャを予測するために使用されるRPL内に挿入するステップが、前記現在のピクチャを予測するために使用される前記RPLが前記現在のピクチャのみを含むように、前記現在のピクチャを予測するために使用される前記RPLを構成するステップを備える、請求項1に記載の方法。

【0182】

12. 前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスがインタースライスであり、前記現在のピクチャを、前記現在のピクチャを予測するために使用される前記RPL内に挿入するステップが、前記現在のピクチャを予測するために使用される前記RPLが、前記現在のピクチャと、ビデオデータの1つまたは複数の他のピクチャとを含むように、前記現在のピクチャを予測するために使用される前記RPLを構成するステップを備える、請求項1に記載の方法。

【0183】

13. 前記ビデオコードによって、前記現在のピクチャの前記ブロックをコーディングする前に、前記ビデオデータの前記現在のピクチャを長期参照ピクチャとしてマーキングするステップと、前記ビデオデータによって、前記現在のピクチャの前記ブロックをコーディングした後に、前記ビデオデータの前記現在のピクチャを短期参照ピクチャとしてマーキングするステップとをさらに備える、請求項1に記載の方法。

【0184】

14. 前記ブロックをコーディングするステップが前記ブロックを符号化するステップを備え、前記方法が、符号化ビデオビットストリーム内に、ビデオデータの前記ブロックとビデオデータの前記予測子ブロックとの間の変位を表すベクトルの表現を符号化するステップをさらに備える、請求項1に記載の方法。

【0185】

15. 前記ブロックをコーディングするステップが前記ブロックを復号するステップを備え、前記方法が、符号化ビデオビットストリームに基づいて、ビデオデータの前記ブロックとビデオデータの前記予測子ブロックとの間の変位を表すベクトルを決定するステップ

10

20

30

40

50

をさらに備える、請求項1に記載の方法。

【0186】

16. 前記参照ピクチャバッファ内に前記現在のピクチャの前記バージョンを記憶するステップが、前記ビデオコードによって、前記参照ピクチャバッファ内に、固定された値に初期化されたサンプルピクセル値を有する前記現在のピクチャのバージョンを記憶するステップを備える、請求項1に記載の方法。

【0187】

17. ビデオデータを符号化または復号するためのデバイスであって、前記デバイスが、前記ビデオデータの1つまたは複数のピクチャを記憶するように構成された参照ピクチャバッファと、例1~16の任意の組合せの方法を実行するように構成された1つまたは複数のプロセッサとを備えるデバイス。

10

【0188】

18. ビデオデータを符号化または復号するためのデバイスであって、前記デバイスが、例1~16の任意の組合せの方法を実行するための手段を備えるデバイス。

【0189】

19. 実行されたとき、ビデオコードの1つまたは複数のプロセッサに例1~16の任意の組合せの方法を実行させる命令を記憶するコンピュータ可読記憶媒体。

【0190】

本開示の特定の態様は、例示の目的のためにHEVC規格に関して説明されている。しかしながら、本開示に記載の技法は、現在開発中のH.266ビデオコーディング規格などの、まだ開発されていない他の規格または独自のビデオコーディングプロセスを含む、他のビデオコーディングプロセスに有用であり得る。

20

【0191】

ビデオコードは、本開示に記載のように、ビデオエンコーダまたはビデオデコーダを指してもよい。同様に、ビデオコーディングユニットは、ビデオエンコーダまたはビデオデコーダを指してもよい。同様に、ビデオコーディングは、該当すれば、ビデオ符号化またはビデオ復号を指してもよい。

【0192】

例に応じて、本明細書で説明される技法のいずれかのいくつかの行為またはイベントは異なる順序で実行されてもよく、一緒に追加され、統合され、または省略されてもよい(たとえば、説明される行為またはイベントのすべてが技法の実施のために必要とは限らない)ことを認識されたい。さらに、いくつかの例では、行為またはイベントは、順次的ではなく、たとえばマルチスレッド処理、割り込み処理またはマルチプロセッサを通じて同時に実行され得る。

30

【0193】

1つまたは複数の例では、説明した機能は、ハードウェア、ソフトウェア、ファームウェア、またはそれらの任意の組合せにおいて実装されてもよい。ソフトウェアにおいて実装された場合、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に記憶またはコンピュータ可読媒体を介して送信されてもよく、ハードウェアベースの処理ユニットによって実行されてもよい。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、通信プロトコルに従う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んでもよい。

40

【0194】

このように、コンピュータ可読媒体は、一般に、(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してもよい。データ記憶媒体は、本開示に記載の技法の実装のための命令、コード、および/またはデータ構造を取得するために、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ってもよい。コンピュータプログラム製品は、コンピュータ可読媒体を含み得る。

50

【0195】

例として、限定はしないが、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デバイス、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得、コンピュータによってアクセスされ得る任意の他の媒体を備えることができる。また、任意の接続が、適切にコンピュータ可読媒体と呼ばれる。たとえば、命令が、同軸ケーブル、光ファイバケーブル、より対線、デジタル加入者線(DSL)、または、赤外線、無線、およびマイクロ波などのワイヤレス技術を使用して、ウェブサイト、サーバ、または他の遠隔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、より対線、DSL、または、赤外線、無線、およびマイクロ波などのワイヤレス技術は、媒体の定義内に含まれる。

【0196】

しかしながら、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、代わりに、非一時的な有形の記憶媒体に向けられていることが理解されるべきである。ディスク(disk)およびディスク(disc)は、本明細書で使用される場合、コンパクトディスク(CD)、レーザディスク、光ディスク、デジタル多用途ディスク(DVD)、フロッピー(登録商標)ディスク、およびBlu-ray(登録商標)ディスクを含み、ディスク(disk)は通常、磁氣的にデータを再生し、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。上記の組合せも、コンピュータ可読媒体の範囲内に含まれるべきである。

【0197】

命令は、1つもしく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論理アレイ(FPGA)、または他の同等の集積もしくはディスクリット論理回路などの、1つまたは複数のプロセッサによって実行されてもよい。したがって、本明細書で使用する「プロセッサ」という用語は、任意の前述の構造、または本明細書で説明する技法の実装に適した任意の他の構造のいずれかを指す場合がある。さらに、いくつかの態様では、本明細書で説明する機能は、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モジュールおよび/またはソフトウェアモジュール内に与えられてもよく、あるいは複合コーデックに組み込まれてもよい。また、技法は、1つまたは複数の回路または論理要素において完全に実装され得る。

【0198】

本開示の技法は、ワイヤレスハンドセット、集積回路(IC)、またはICのセット(たとえば、チップセット)を含む、多種多様なデバイスまたは装置において実装されてもよい。本開示では、開示される技法を実行するように構成されたデバイスの機能的態様を強調するために、様々なコンポーネント、モジュール、またはユニットが説明されたが、それらのコンポーネント、モジュール、またはユニットは、必ずしも異なるハードウェアユニットによる実現を必要とするとは限らない。そうではなくて、上で説明されたように、様々なユニットは、コーデックハードウェアユニットにおいて結合されてよく、または適切なソフトウェアおよび/もしくはファームウェアとともに、前述のような1つもしくは複数のプロセッサを含む、相互動作可能なハードウェアユニットの集合によって提供されてよい。

【0199】

様々な例が説明されている。これらおよび他の例は以下の特許請求の範囲内に入る。

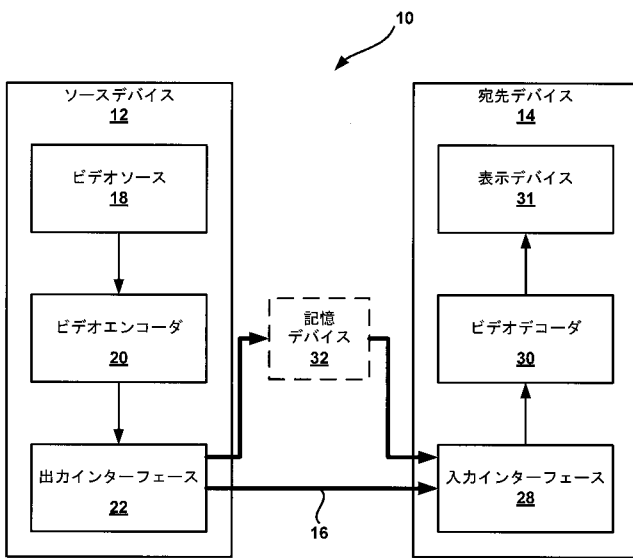
【符号の説明】

【0200】

- 10 ビデオ符号化および復号システム
- 12 ソースデバイス
- 14 宛先デバイス
- 16 コンピュータ可読媒体
- 18 ビデオソース

20	ビデオエンコーダ	
22	出力インターフェース	
28	入力インターフェース	
30	ビデオデコーダ	
31	表示デバイス	
32	記憶デバイス	
33	ビデオシーケンス	
34	ピクチャ	
35A	ピクチャ	
35B	ピクチャ	10
35C	ピクチャ	
36A	ピクチャ	
36B	ピクチャ	
38A	ピクチャ	
38B	ピクチャ	
39	ピクチャ	
40	ビデオデータメモリ	
42	予測処理ユニット	
44	動き推定ユニット	
46	動き補償ユニット	20
48	イントラ予測ユニット	
50	加算器	
52	変換処理ユニット	
54	量子化処理ユニット	
56	エントロピー符号化ユニット	
58	逆量子化処理ユニット	
60	逆変換処理ユニット	
62	加算器	
64	参照ピクチャメモリ	
69	ビデオデータメモリ	30
70	エントロピー復号ユニット	
71	予測処理ユニット	
72	動き補償ユニット	
74	イントラ予測ユニット	
76	逆量子化処理ユニット	
78	逆変換処理ユニット	
80	加算器	
82	参照ピクチャメモリ	
92	再構成領域	
102	現在のブロック	40
104	予測子ブロック	
106	動きベクトル	
108	再構成された領域	

【図 1】



【図 2】

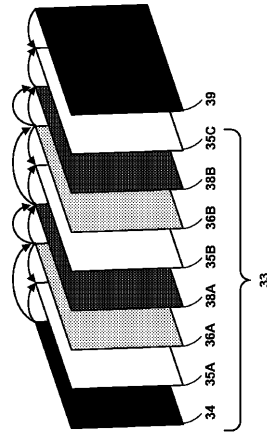
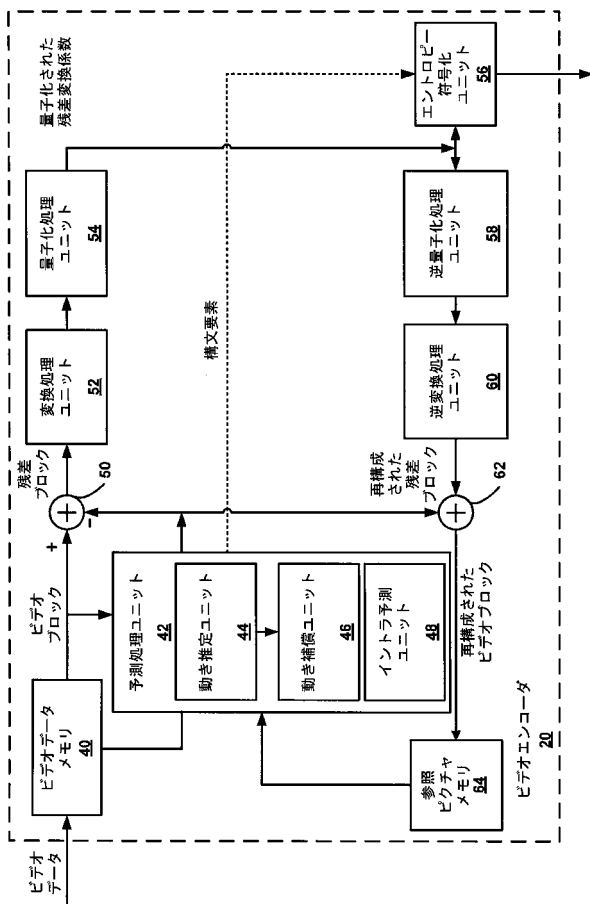


FIG. 2

【図 3】



【図 4】

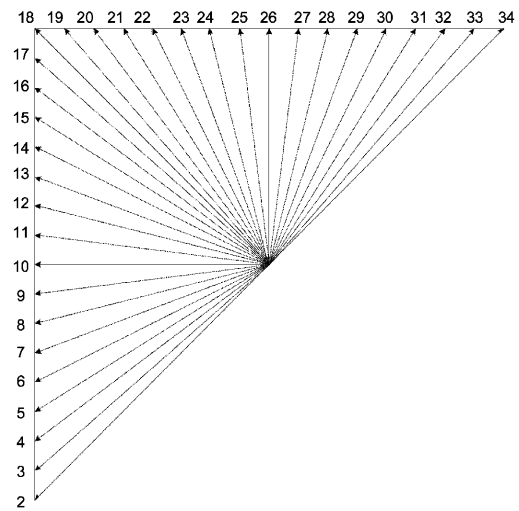
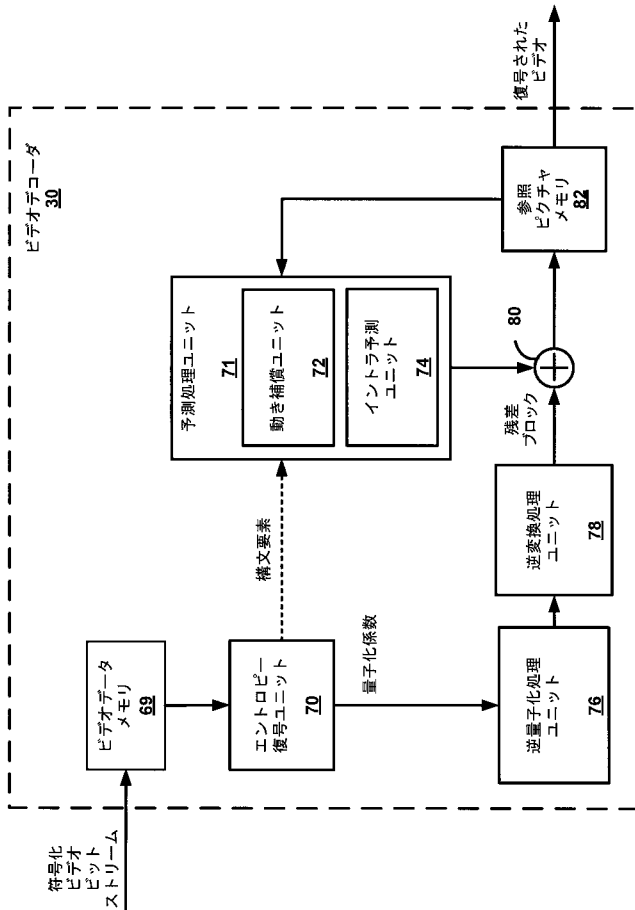


FIG. 4

【図 5】



【図 6】

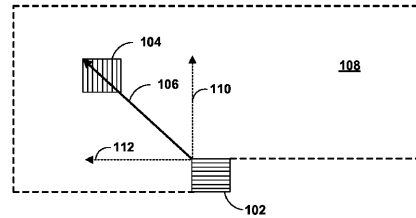
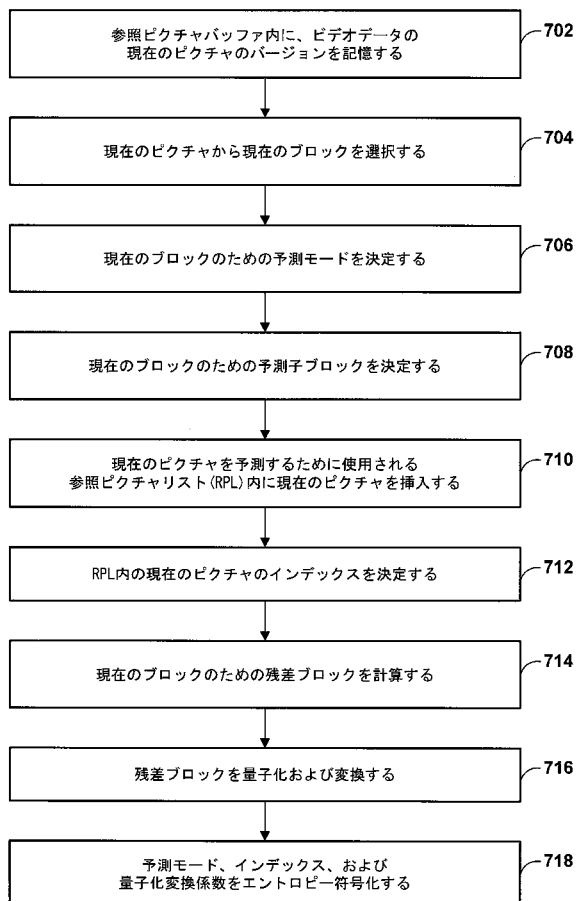
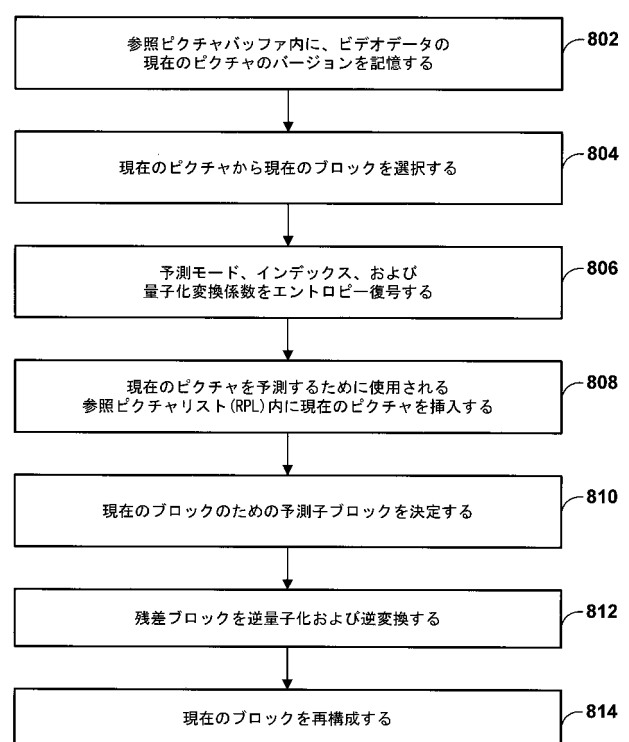


FIG. 6

【図 7】



【図 8】



【手続補正書】

【提出日】平成28年9月23日(2016.9.23)

【手続補正 1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項 1】

ビデオデータを符号化または復号する方法であって、前記方法が、

ビデオコードによって、前記ビデオデータのピクチャが前記ピクチャ自体を時間的に予測するために使用される参照ピクチャリスト(RPL)内に存在し得るかどうかを示す構文要素をコーディングするステップと、

前記ビデオコードによって、参照ピクチャバッファ内に、前記ビデオデータの現在のピクチャのバージョンを記憶するステップと、

前記構文要素が、前記ビデオデータのピクチャが前記ピクチャ自体を時間的に予測するために使用されるRPL内に存在し得ることを示すことに基づいて、前記現在のピクチャのブロックを時間的に予測する際に使用されるRPL内に前記現在のピクチャの指標を挿入するステップと、

前記ビデオコードによって、前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記バージョン内に含まれるビデオデータの予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第1のブロックをコーディングするステップと

を備える方法。

【請求項 2】

前記ビデオコードによって、前記第1のブロックをコーディングした後に、前記参照ピクチャバッファ内の前記現在のピクチャの前記バージョンを、前記コーディングされた第1のブロックを含む前記現在のピクチャの更新されたバージョンで更新するステップと、

前記ビデオコードによって、前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記更新されたバージョン内に含まれる予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第2のブロックをコーディングするステップと

をさらに備える、請求項1に記載の方法。

【請求項 3】

前記構文要素をコーディングするステップが、前記現在のピクチャによって参照されるビデオパラメータセット(VPS)、前記現在のピクチャによって参照されるシーケンスパラメータセット(SPS)、または前記現在のピクチャによって参照されるピクチャパラメータセット(PPS)内に前記構文要素をコーディングするステップを備える、請求項1に記載の方法。

【請求項 4】

前記方法が、前記第1のブロックがイントラブロックコピー(イントラBC)を使用してコーディングされるかどうかを示す構文要素をコーディングするステップを含まない、請求項3に記載の方法。

【請求項 5】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記スライスのための並置参照インデックスを示す構文要素が前記現在のピクチャではないピクチャを示す、請求項1に記載の方法。

【請求項 6】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記予測子ブロックが前記現在のスライス内に含まれる、請求項1に記載の方法。

【請求項 7】

前記RPL内の各エントリがインデックス値を有し、前記現在のピクチャを、前記現在のピクチャを時間的に予測するために使用される前記RPL内に挿入するステップが、少なくとも、

長期RPS内のピクチャのインデックス値よりも小さいインデックス値を用いて前記RPL内に前記現在のピクチャを挿入するか、

長期RPS内のピクチャのインデックス値よりも大きいインデックス値を用いて前記RPL内に前記現在のピクチャを挿入するか、

または固定されたインデックス値を用いて前記RPL内に前記現在のピクチャを挿入することによって、

1つまたは複数の参照ピクチャセット(RPS)に基づいて前記現在のピクチャを時間的に予測するために使用される前記RPLを構成するステップを備える、請求項1に記載の方法。

【請求項 8】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスがイントラスライスであり、前記現在のピクチャを、前記現在のピクチャを時間的に予測するために使用されるRPL内に挿入するステップが、前記現在のピクチャを時間的に予測するために使用される前記RPLが前記現在のピクチャのみを含むように、前記現在のピクチャを時間的に予測するために使用される前記RPLを構成するステップを備える、請求項1に記載の方法。

【請求項 9】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスがインタースライスであり、前記現在のピクチャを、前記現在のピクチャを時間的に予測するために使用される前記RPL内に挿入するステップが、前記現在のピクチャを時間的に予測するために使用される前記RPLが、前記現在のピクチャと、ビデオデータの1つまたは複数の他のピクチャとを含むように、前記現在のピクチャを時間的に予測するために使用される前記RPLを構成するステップを備える、請求項1に記載の方法。

【請求項 10】

前記ビデオコードによって、前記現在のピクチャの前記ブロックをコーディングする前に、前記ビデオデータの前記現在のピクチャを長期参照ピクチャとしてマーキングするステップと、

前記ビデオデータによって、前記現在のピクチャの前記ブロックをコーディングした後に、前記ビデオデータの前記現在のピクチャを短期参照ピクチャとしてマーキングするステップと

をさらに備える、請求項1に記載の方法。

【請求項 11】

前記ブロックをコーディングするステップが前記ブロックを符号化するステップを備え、前記方法が、符号化ビデオビットストリーム内に、ビデオデータの前記ブロックとビデオデータの前記予測子ブロックとの間の変位を表すベクトルの表現を符号化するステップをさらに備える、請求項1に記載の方法。

【請求項 12】

前記ブロックをコーディングするステップが前記ブロックを復号するステップを備え、前記方法が、符号化ビデオビットストリームに基づいて、ビデオデータの前記ブロックとビデオデータの前記予測子ブロックとの間の変位を表すベクトルを決定するステップをさらに備える、請求項1に記載の方法。

【請求項 13】

ビデオデータを符号化または復号するためのデバイスであって、前記デバイスが、

前記ビデオデータの1つまたは複数のピクチャを記憶するように構成された参照ピクチャバッファと、

1つまたは複数のプロセッサとを備え、前記1つまたは複数のプロセッサが、

前記ビデオデータのピクチャが前記ピクチャ自体を時間的に予測するために使用され

る参照ピクチャリスト(RPL)内に存在し得るかどうかを示す構文要素をコーディングし、
参照ピクチャバッファ内に、前記ビデオデータの現在のピクチャのバージョンを記憶し、

前記構文要素が、前記ビデオデータのピクチャが前記ピクチャ自体を時間的に予測するために使用されるRPL内に存在し得ることを示すことに基づいて、前記現在のピクチャのブロックを時間的に予測する際に使用されるRPL内に前記現在のピクチャの指標を挿入し、

前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記バージョン内に含まれるビデオデータの予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第1のブロックをコーディングするように構成された、デバイス。

【請求項14】

前記1つまたは複数のプロセッサが、

前記第1のブロックをコーディングした後に、前記参照ピクチャバッファ内の前記現在のピクチャの前記バージョンを、前記コーディングされた第1のブロックを含む前記現在のピクチャの更新されたバージョンで更新し、

前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記更新されたバージョン内に含まれる予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第2のブロックをコーディングするようにさらに構成された、請求項13に記載のデバイス。

【請求項15】

前記1つまたは複数のプロセッサが、前記現在のピクチャによって参照されるビデオパラメータセット(VPS)、前記現在のピクチャによって参照されるシーケンスパラメータセット(SPS)、または前記現在のピクチャによって参照されるピクチャパラメータセット(PPS)内に前記構文要素をコーディングするようにさらに構成された、請求項13に記載のデバイス。

【請求項16】

前記1つまたは複数のプロセッサが、前記第1のブロックがイントラブロックコピー(イントラBC)を使用してコーディングされるかどうかを示す構文要素をコーディングするように構成されていない、請求項15に記載のデバイス。

【請求項17】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスのための並置参照インデックスを示す構文要素が前記現在のピクチャではないピクチャを示す、請求項13に記載のデバイス。

【請求項18】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記予測子ブロックが前記現在のスライス内に含まれる、請求項13に記載のデバイス。

【請求項19】

前記RPL内の各エントリがインデックス値を有し、前記現在のピクチャを、前記現在のピクチャを時間的に予測するために使用される前記RPL内に挿入するために、前記1つまたは複数のプロセッサが、少なくとも、

長期RPS内のピクチャのインデックス値よりも小さいインデックス値を用いて前記RPL内に前記現在のピクチャを挿入するか、

長期RPS内のピクチャのインデックス値よりも大きいインデックス値を用いて前記RPL内に前記現在のピクチャを挿入するか、

または固定されたインデックス値を用いて前記RPL内に前記現在のピクチャを挿入することによって、

1つまたは複数の参照ピクチャセット(RPS)に基づいて前記現在のピクチャを時間的に予測するために使用される前記RPLを構成するように構成された、請求項13に記載のデバイス。

【請求項 20】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスがイントラスライスであり、前記現在のピクチャを、前記現在のピクチャを時間的に予測するために使用される前記RPL内に挿入するために、前記1つまたは複数のプロセッサが、前記現在のピクチャを時間的に予測するために使用される前記RPLが前記現在のピクチャのみを含むように、前記現在のピクチャを時間的に予測するために使用される前記RPLを構成するように構成された、請求項13に記載のデバイス。

【請求項 21】

前記ブロックが前記現在のピクチャの現在のスライス内に含まれ、前記現在のスライスがインタースライスであり、前記現在のピクチャを、前記現在のピクチャを時間的に予測するために使用される前記RPL内に挿入するために、前記1つまたは複数のプロセッサが、前記現在のピクチャを時間的に予測するために使用される前記RPLが、前記現在のピクチャと、ビデオデータの1つまたは複数の他のピクチャとを含むように、前記現在のピクチャを時間的に予測するために使用される前記RPLを構成するように構成された、請求項13に記載のデバイス。

【請求項 22】

前記1つまたは複数のプロセッサが、

前記現在のピクチャの前記ブロックをコーディングする前に、前記ビデオデータの前記現在のピクチャを長期参照ピクチャとしてマーキングし、

前記現在のピクチャの前記ブロックをコーディングした後に、前記ビデオデータの前記現在のピクチャを短期参照ピクチャとしてマーキングするようにさらに構成された、請求項13に記載のデバイス。

【請求項 23】

前記ビデオコーダがビデオエンコーダであり、前記1つまたは複数のプロセッサが、符号化ビデオビットストリーム内に、ビデオデータの前記ブロックとビデオデータの前記予測子ブロックとの間の変位を表すベクトルの表現を符号化するようにさらに構成された、請求項13に記載のデバイス。

【請求項 24】

前記ビデオコーダがビデオデコーダであり、前記1つまたは複数のプロセッサが、符号化ビデオビットストリームに基づいて、ビデオデータの前記ブロックとビデオデータの前記予測子ブロックとの間の変位を表すベクトルを決定するようにさらに構成された、請求項13に記載のデバイス。

【請求項 25】

ビデオデータを符号化または復号するためのデバイスであって、

前記ビデオデータのピクチャが前記ピクチャ自体を時間的に予測するために使用される参照ピクチャリスト(RPL)内に存在し得るかどうかを示す構文要素をコーディングするための手段と、

参照ピクチャバッファ内に、前記ビデオデータの現在のピクチャのバージョンを記憶するための手段と、

前記構文要素が、前記ビデオデータのピクチャが前記ピクチャ自体を時間的に予測するために使用されるRPL内に存在し得ることを示すことに基づいて、前記現在のピクチャのブロックを時間的に予測する際に使用されるRPL内に前記現在のピクチャの指標を挿入するための手段と、

前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記バージョン内に含まれるビデオデータの予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第1のブロックをコーディングするための手段とを備えるデバイス。

【請求項 26】

実行されたとき、ビデオコーダの1つまたは複数のプロセッサに、

ビデオデータのピクチャが前記ピクチャ自体を時間的に予測するために使用される参照

ピクチャリスト(RPL)内に存在し得るかどうかを示す構文要素をコーディングさせ、

参照ピクチャバッファ内に、前記ビデオデータの現在のピクチャのバージョンを記憶させ、

前記構文要素が、前記ビデオデータのピクチャが前記ピクチャ自体を時間的に予測するために使用されるRPL内に存在し得ることを示すことに基づいて、前記現在のピクチャのブロックを時間的に予測する際に使用されるRPL内に前記現在のピクチャの指標を挿入させ、

前記RPLに基づいて、前記参照ピクチャバッファ内に記憶された前記現在のピクチャの前記バージョン内に含まれるビデオデータの予測子ブロックに基づいて、前記現在のピクチャ内のビデオデータの第1のブロックをコーディングさせる命令を記憶するコンピュータ可読記憶媒体。

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2015/021866

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04N19/503
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	SIU-LEONG YU AND CHRISTOS CHRYSAFIS: "New Intra Prediction using Intra-Macroblock Motion Compensation", JOINT VIDEO TEAM (JVT) OF ISO/IEC MPEG & ITU-T VCEG(ISO/IEC JTC1/SC29/WG11 AND ITU-T SG16 Q.6), JVT-C151, 3RD MEETING: FAIRFAX, VIRGINIA, USA, 6 May 2002 (2002-05-06), pages 1-3, XP008158737, Retrieved from the Internet: URL: http://www.google.nl/#hl=en&tbo=d&outp ut=search&scIient=psy-ab&q=Joint+Video+Team+(JVT)+of+ISO%2FIEC+MPEG+%26+ITU-T+VCEG(I SO%2FIEC+JTC1%2FSC29%2FWG11+and+ITU-T+SG16 +Q.6)%2C+JVT-C151%2C+3rd+Meeting:+Fairfax% 2C+Virginia%2C+USA&oq=Joint+Video+Team+(JV T)+of+ISO%2FIEC+MPEG+%26+ITU-T+VCEG(ISO%2F IEC+JTC1%2 -/-	1,2,14, 15,27,29

☒ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

2 July 2015

Date of mailing of the international search report

10/07/2015

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Lindgren, Johan

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2015/021866

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	the whole document ----- SULLIVAN G J ET AL: "Meeting Report of 15th JCT-VC Meeting", 15. JCT-VC MEETING; 23-10-2013 - 1-11-2013; GENEVA; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-01000, 9 January 2014 (2014-01-09), XP030115455, the whole document -----	1,3-11, 14, 16-24, 27-30
X	FLYNN D ET AL: "Range Extensions Draft 4", 14. JCT-VC MEETING; 25-7-2013 - 2-8-2013; VIENNA; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-N1005-v3, 8 August 2013 (2013-08-08), XP030114950, Chapters 7.4.3.2, 7.4.9.5, 8.4.4.2.7 -----	1,3-14, 16-30

フロントページの続き

(81)指定国 AP(BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), EA(AM, AZ, BY, KG, KZ, RU, TJ, TM), EP(AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US

(72)発明者 チャオ・パン

アメリカ合衆国・カリフォルニア・92121-1714・サン・ディエゴ・モアハウス・ドライブ・5775

(72)発明者 イン・チェン

アメリカ合衆国・カリフォルニア・92121-1714・サン・ディエゴ・モアハウス・ドライブ・5775

(72)発明者 イェークイ・ワン

アメリカ合衆国・カリフォルニア・92121-1714・サン・ディエゴ・モアハウス・ドライブ・5775

Fターム(参考) 5C159 MA04 MA05 MA21 MC11 ME01 NN01 RB09 UA02 UA05 UA33