

【特許請求の範囲】**【請求項 1】**

ビデオデータを復号する方法であって、

包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断することと、

前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替することと、

前記候補リスト中の複数のブロックのうちの 1 つから視差動きベクトル予測子を選択することと、

前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号することとを備える方法。

【請求項 2】

前記包含ブロックが動き推定領域 (MER) を備え、前記現在のブロックがコーディングユニット (CU) および予測ユニット (PU) のうちの 1 つを備える、請求項 1 に記載の方法。

【請求項 3】

代替することが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記 MER の左下隣接ブロックで代替することを備える、請求項 2 に記載の方法。

【請求項 4】

代替することが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記 MER の左隣接ブロックで代替することを備え、結果的に前記 MER の前記左隣接ブロックが前記現在のブロックの前記左下隣接ブロックと同じ垂直位置を有するようになる、請求項 2 に記載の方法。

【請求項 5】

代替することが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記 MER の右上隣接ブロックで代替することを備える、請求項 2 に記載の方法。

【請求項 6】

代替することが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記 MER の上隣接ブロックで代替することを備え、結果的に前記 MER の前記上隣接ブロックが前記現在のブロックの前記右上隣接ブロックと同じ水平位置を有するようになる、請求項 2 に記載の方法。

【請求項 7】

代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記 MER の左上隣接ブロックで代替することを備える、請求項 2 に記載の方法。

【請求項 8】

代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記 MER の左隣接ブロックで代替することを備え、結果的に前記 MER の前記左隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ垂直位置を有するようになる、請求項 2 に記載の方法。

【請求項 9】

代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記 MER の上隣接ブロックで代替ことを備え、結果的に前記 MER の前記上隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ水平位置を有するようになる、請求項 2 に記載の方法。

【請求項 10】

前記包含ブロックがコーディングユニット (CU) を備え、前記現在のブロックが予測

10

20

30

40

50

ユニット（PU）を備える、請求項１に記載の方法。

【請求項１１】

前記視差動きベクトルを使用してビデオデータの前記現在のブロックを復号することをさらに備える、請求項１に記載の方法。

【請求項１２】

前記視差動きベクトルを復号することが、

動きベクトル差分値を復号することと、

前記視差動きベクトル予測子に前記動きベクトル差分値を追加して前記視差動きベクトルを再生することと

を備える、請求項１に記載の方法。

10

【請求項１３】

ビデオデータを復号するためのデバイスであって、包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断することと、前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替することと、前記候補リスト中の複数のブロックのうちの１つから視差動きベクトル予測子を選択することと、前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号することとを行うように構成されたビデオデコーダを備えるデバイス。

【請求項１４】

前記包含ブロックが動き推定領域（MER）を備え、前記現在のブロックがコーディングユニット（CU）および予測ユニット（PU）のうちの１つを備える、請求項１３に記載のデバイス。

20

【請求項１５】

前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記MERの左下隣接ブロックで代替するように構成される、請求項１４に記載のデバイス。

【請求項１６】

前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記MERの左隣接ブロックで代替するように構成され、結果的に前記MERの前記左隣接ブロックが前記現在のブロックの前記左下隣接ブロックと同じ垂直位置を有するようになる、請求項１４に記載のデバイス。

30

【請求項１７】

前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記MERの右上隣接ブロックで代替するように構成される、請求項１４に記載のデバイス。

【請求項１８】

前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記MERの上隣接ブロックで代替するように構成され、結果的に前記MERの前記上隣接ブロックが前記現在のブロックの前記右上隣接ブロックと同じ水平位置を有するようになる、請求項１４に記載のデバイス。

40

【請求項１９】

前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの左上隣接ブロックで代替するように構成される、請求項１４に記載のデバイス。

【請求項２０】

前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの左隣接プロ

50

ックで代替するように構成され、結果的に前記 M E R の前記左隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ垂直位置を有するようになる、請求項 1 4 に記載のデバイス。

【請求項 2 1】

前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記 M E R の上隣接ブロックで代替するように構成され、結果的に前記 M E R の前記上隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ水平位置を有するようになる、請求項 1 4 に記載のデバイス。

【請求項 2 2】

前記包含ブロックがコーディングユニット (C U) を備え、前記現在のブロックが予測ユニット (P U) を備える、請求項 1 3 に記載のデバイス。

【請求項 2 3】

前記ビデオデコーダが、前記視差動きベクトルを使用してビデオデータの前記現在のブロックを復号するようにさらに構成される、請求項 1 3 に記載のデバイス。

【請求項 2 4】

前記ビデオデコーダが、動きベクトル差分値を復号することと、前記視差動きベクトル予測子に前記動きベクトル差分値を追加して前記視差動きベクトルを再生することとを行うように構成される、請求項 1 3 に記載のデバイス。

【請求項 2 5】

前記デバイスが、
集積回路、
マイクロプロセッサ、および
ワイヤレス通信デバイス
のうちの少なくとも 1 つを備える、請求項 1 3 に記載のデバイス。

【請求項 2 6】

ビデオデータを復号するためのデバイスであって、
包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断するための手段と、
前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替するための手段と、
前記候補リスト中の複数のブロックのうちの 1 つから視差動きベクトル予測子を選択するための手段と、
前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号するための手段とを備えるデバイス。

【請求項 2 7】

前記包含ブロックが動き推定領域 (M E R) を備え、前記現在のブロックがコーディングユニット (C U) および予測ユニット (P U) のうちの 1 つを備える、請求項 2 6 に記載のデバイス。

【請求項 2 8】

実行されたとき、1 つまたは複数のプロセッサに、
包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断することと、
前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替することと、
前記候補リスト中の複数のブロックのうちの 1 つから視差動きベクトル予測子を選択することと、
前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号することと

10

20

30

40

50

を行わせる命令を記憶したコンピュータ可読記憶媒体。

【請求項 29】

前記包含ブロックが動き推定領域 (MER) を備え、前記現在のブロックがコーディングユニット (CU) および予測ユニット (PU) のうちの 1 つを備える、請求項 28 に記載のコンピュータ可読記憶媒体。

【請求項 30】

ビデオデータを符号化する方法であって、

視差動きベクトルを使用して、包含ブロック内のビデオデータの現在のブロックを符号化することと、

前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替することと、

前記候補リスト中の複数のブロックのうちの 1 つから視差動きベクトル予測子を選択することと、

前記視差動きベクトル予測子に基づいて前記視差動きベクトルを符号化することとを備える方法。

【請求項 31】

前記包含ブロックが動き推定領域 (MER) を備え、前記現在のブロックがコーディングユニット (CU) および予測ユニット (PU) のうちの 1 つを備える、請求項 30 に記載の方法。

【請求項 32】

代替することが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記 MER の左下隣接ブロックで代替することを備える、請求項 31 に記載の方法。

【請求項 33】

代替することが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記 MER の左隣接ブロックで代替することを備え、結果的に前記 MER の前記左隣接ブロックが前記現在のブロックの前記左下隣接ブロックと同じ垂直位置を有するようになる、請求項 31 に記載の方法。

【請求項 34】

代替することが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記 MER の右上隣接ブロックで代替することを備える、請求項 31 に記載の方法。

【請求項 35】

代替することが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記 MER の上隣接ブロックで代替することを備え、結果的に前記 MER の前記上隣接ブロックが前記現在のブロックの前記右上隣接ブロックと同じ水平位置を有するようになる、請求項 31 に記載の方法。

【請求項 36】

代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記 MER の左上隣接ブロックで代替することを備える、請求項 31 に記載の方法。

【請求項 37】

代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記 MER の左隣接ブロックで代替することを備え、結果的に前記 MER の前記左隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ垂直位置を有するようになる、請求項 31 に記載の方法。

【請求項 38】

代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記 MER の上隣接ブロックで

10

20

30

40

50

代替することを備え、結果的に前記MERの前記上隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ水平位置を有するようになる、請求項31に記載の方法。

【請求項39】

前記包含ブロックがコーディングユニット(CU)を備え、前記現在のブロックが予測ユニット(PU)を備える、請求項30に記載の方法。

【請求項40】

前記視差動きベクトルを符号化することが、
前記視差動きベクトルと前記視差動きベクトル予測子との間の差分を計算することと、
前記差分を符号化することと
を備える、請求項30に記載の方法。

10

【請求項41】

ビデオデータを符号化するためのデバイスであって、視差動きベクトルを使用して、包含ブロック内のビデオデータの現在のブロックを符号化することと、前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替することと、前記候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、前記視差動きベクトル予測子に基づいて前記視差動きベクトルを符号化することとを行うように構成されたビデオエンコーダを備えるデバイス。

【請求項42】

前記包含ブロックが動き推定領域(MER)を備え、前記現在のブロックがコーディングユニット(CU)および予測ユニット(PU)のうちの1つを備える、請求項41に記載のデバイス。

20

【請求項43】

前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記MERの左下隣接ブロックで代替するように構成される、請求項42に記載のデバイス。

【請求項44】

前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記MERの左隣接ブロックで代替するように構成され、結果的に前記MERの前記左隣接ブロックが前記現在のブロックの前記左下隣接ブロックと同じ垂直位置を有するようになる、請求項42に記載のデバイス。

30

【請求項45】

前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記MERの右上隣接ブロックで代替するように構成される、請求項42に記載のデバイス。

【請求項46】

前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記MERの上隣接ブロックで代替するように構成され、結果的に前記MERの前記上隣接ブロックが前記現在のブロックの前記右上隣接ブロックと同じ水平位置を有するようになる、請求項42に記載のデバイス。

40

【請求項47】

前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの左上隣接ブロックで代替するように構成される、請求項42に記載のデバイス。

【請求項48】

前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの左隣接ブロックで代替するように構成され、結果的に前記MERの前記左隣接ブロックが前記現在

50

のブロックの前記左上隣接ブロックと同じ垂直位置を有するようになる、請求項 4 2 に記載のデバイス。

【請求項 4 9】

前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記 M E R の上隣接ブロックで代替するように構成され、結果的に前記 M E R の前記上隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ水平位置を有するようになる、請求項 4 2 に記載のデバイス。

【請求項 5 0】

集積回路、
マイクロプロセッサ、および
ワイヤレス通信デバイス

のうちの少なくとも 1 つを備える、請求項 4 1 に記載のデバイス。

【発明の詳細な説明】

【関連出願】

【0 0 0 1】

[0001] 本出願は、その内容全体が参照により本明細書に組み込まれる、2012年4月20日に提出された米国仮出願第61/636,554号の利益を主張する。

【技術分野】

【0 0 0 2】

[0002] 本開示は、ビデオコーディングに関する。

【背景技術】

【0 0 0 3】

[0003] デジタルビデオ機能は、デジタルテレビジョン、デジタルダイレクトブロードキャストシステム、ワイヤレスブロードキャストシステム、携帯情報端末(PDA)、ラップトップまたはデスクトップコンピュータ、タブレットコンピュータ、電子ブックリーダー、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」、ビデオ遠隔会議デバイス、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得る。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, Advanced Video Coding (AVC)、現在開発中の高効率ビデオコーディング(HEVC: High Efficiency Video Coding)規格によって定義された規格、およびそのような規格の拡張に記載されているビデオコーディング技法など、ビデオコーディング技法を実装する。HEVCが、ITU-T Video Coding Experts Group (VCEG)とISO/IEC Motion Picture Experts Group (MPEG)とのJoint Collaboration Team on Video Coding (JCT-VC)によって開発されている。HEVCのドラフトは、http://phenix.int-evry.fr/jct/doc_end_user/documents/8_San%20Jose/wg11/JCTVC-H1003-v22.zipから入手可能である。H.264/AVCは、「Advanced video coding for generic audiovisual services」、ITU-T勧告H.264、2010年3月に記載されている、Scalable Video Coding (SVC)およびMultiview Video Coding (MVC)の拡張を含む。ビデオデバイスは、そのようなビデオコーディング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/または記憶し得る。

【0 0 0 4】

[0004] ビデオコーディング技法は、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の空間的(イントラピクチャ(intra-picture))予測および/または時間的(

インターピクチャ(inter-picture)) 予測を含む。ブロックベースのビデオコーディングの場合、ビデオスライス(例えば、ビデオフレームまたはビデオフレームの一部)が、ツリーブロック、コーディングユニット(CU: coding units) および/またはコーディングノードと呼ばれることもあるビデオブロックに区分され得る。ピクチャのイントラコーディングされた(I)スライス中のビデオブロックは、同じピクチャ中の近隣ブロック中の参照サンプルに対する空間的予測を使用して符号化される。ピクチャのインターコーディングされた(PまたはB)スライス中のビデオブロックは、同じピクチャ中の近隣ブロック中の参照サンプルに対する空間的予測、または他の参照ピクチャ(reference pictures)中の参照サンプルに対する時間的予測を使用し得る。ピクチャはフレームと呼ばれることがあり、参照ピクチャは参照フレーム(reference frames)と呼ばれることがある。

10

【0005】

[0005] 空間的予測または時間的予測は、コーディングされるべきブロックの予測ブロックを生じる。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分を表す。インターコード化ブロックは、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、コード化ブロックと予測ブロックとの間の差分を示す残差データとに従って符号化される。イントラコード化ブロックは、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、残差変換係数(residual transform coefficients)が得られ得、その残差変換係数は、次いで量子化され得る。量子化変換係数(quantized transform coefficients)は、最初は2次元アレイで構成され、変換係数の1次元ベクトルを生成するために走査され得、なお一層の圧縮を達成するために、エントロピーコーディング(entropy coding)が適用され得る。

20

【発明の概要】**【0006】**

[0006] 概して、本開示は、来たるべき高効率ビデオコーディング(HEVC: High Efficiency Video Coding)規格の技法を行うように構成されたコーデックなどのアドバンストコーデックに基づくマルチビュービデオコーディングに関する技法について説明する。そのようなアドバンストコーデックは、HEVCに従って2つ以上のビューをコーディングするように構成されることがあり、こうしたビューの画像が実質的に同時にレンダリングされて、3次元(3D)ビデオ効果をもたらすことができる。より具体的には、本開示は、視差動きベクトル生成(disparity motion vector generation)に関する技法について説明する。

30

【0007】

[0007] 一例で、ビデオデータを復号する方法は、包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトル(disparity motion vector)を使用して符号化されていると判断することと、現在のブロック(current block)の隣接ブロック(neighboring block)も包含ブロック(containing block)内にあるとの判断に基づいて、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替することと、候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子(disparity motion vector predictor)を選択することと、視差動きベクトル予測子に基づいて視差動きベクトルを復号することとを含む。

40

【0008】

[0008] 別の例で、ビデオデータを復号するためのデバイスは、包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断することと、現在のブロックの隣接ブロックも包含ブロック内にあるとの判断に基づいて、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替することと、候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、視差動きベクトル予測子に基づいて視差動きベクトルを復号することとを行うように構成されたビデオデコーダを含む。

【0009】

50

【0009】 別の例で、ビデオデータを復号するためのデバイスは、包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断するための手段と、現在のブロックの隣接ブロックも包含ブロック内にあるとの判断に基づいて、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替するための手段と、候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択するための手段と、視差動きベクトル予測子に基づいて視差動きベクトルを復号するための手段とを含む。

【0010】

【0010】 別の例で、コンピュータ可読記憶媒体は、実行されたとき、1つまたは複数のプロセッサに、包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断することと、現在のブロックの隣接ブロックも包含ブロック内にあるとの判断に基づいて、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替することと、候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、視差動きベクトル予測子に基づいて視差動きベクトルを復号することとを行わせる命令を記憶している。

【0011】

【0011】 別の例で、ビデオデータを符号化する方法は、視差動きベクトルを使用して、包含ブロック内のビデオデータの現在のブロックを符号化することと、現在のブロックの隣接ブロックも包含ブロック内にあるとの判断に基づいて、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替することと、候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、視差動きベクトル予測子に基づいて視差動きベクトルを符号化することとを含む。

【0012】

【0012】 別の例で、ビデオデータを符号化するためのデバイスは、視差動きベクトルを使用して、包含ブロック内のビデオデータの現在のブロックを符号化することと、現在のブロックの隣接ブロックも包含ブロック内にあるとの判断に基づいて、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替することと、候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、視差動きベクトル予測子に基づいて視差動きベクトルを符号化することとを行うように構成されたビデオエンコーダを含む。

【0013】

【0013】 1つまたは複数の例の詳細は、添付の図面および以下の説明に記載されている。他の特徴、目的、および利点は、その説明および図面から、並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】

【0014】

【図1】 ビュー間予測(inter-view prediction)のための視差ベクトルを生成するための技法を利用し得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。

【図2】 ビュー間予測のための視差ベクトルを生成するための技法を実装し得るビデオエンコーダの一例を示すブロック図。

【図3】 ビュー間予測のための視差ベクトルを生成するための技法を実装し得るビデオデコーダの一例を示すブロック図。

【図4】 例示的なマルチビュービデオコーディング(MVC)の予測パターンを示す概念図。

【図5】 現在の予測ユニット(PU: prediction unit)並びに空間的に隣接するブロック A_0 、 A_1 、 B_0 、 B_1 、および B_2 の例を示すブロック図。

【図6】 様々なピクチャグループ(GOP: groups of pictures)のピクチャを示す、ピクチャが様々なタイプを有する概念図。

【図7】 一連のブロックにおける動きデータ圧縮を示す概念図。

【図8】 PUを含む動き推定範囲(MER: motion estimation range)の外での代替の

10

20

30

40

50

例を示す概念図。

【図 9】視差ベクトルが確認される P U の一例を示す概念図。

【図 10】P U を含む M E R の外での代替の代替例を示す概念図。

【図 11】P U の視差ベクトルに関する 1 つまたは複数の候補ブロックが P U を含む C U 内にある例を示す概念図。

【図 12】視差生成ブロックユニットのための視差ベクトル生成の一例を示す概念図。

【図 13】1 つの領域として結合されたときの隣接ブロックの走査順序の例を示す概念図。

【図 14】時間的視差ベクトル確認(temporal disparity vector checking)のための例示的な方法を示す概念図。

【図 15】本開示の技法による、視差動きベクトルを符号化するための例示的な方法を示すフローチャート。

【図 16】本開示の技法による、視差動きベクトルを復号するための例示的な方法を示すフローチャート。

【発明を実施するための形態】

【0015】

詳細な説明

【0030】 概して、ビデオコーディングで、ビデオコードは、空間的予測（またはイントラ予測）または時間的予測（またはインター予測）のいずれかを使用して、シーケンス中のピクチャを予測コーディングする。3次元（3D）効果を達成するために、2つ（またはそれ以上）のピクチャが実質的に同時に提示されてよく、それにより、異なるビューからの異なるピクチャが観察者の両目の各々によって認識される。特定のカメラアングルまたは水平カメラ位置に対応するシーケンス中のピクチャは、3Dビデオデータのビューを指す。3Dビデオコーディングで、ピクチャは、時間的予測の追加または代替としてビュー間予測され得る。

【0016】

【0031】 ISO/IEC Moving Picture Experts Group (MPEG) と ITU-T Video Coding Experts Group (VCEG) との Joint Collaborative Team on Video Coding (JCT-VC) における高効率ビデオコーディング (HEVC) に関するビデオコーディングの規格化の継続的取り組みがある。HEVC のワーキングドラフトは、JCTVC-H1003、「High efficiency video coding (HEVC) text specification draft 6」、第 8 回会議：サンノゼ、カリフォルニア、米国、2012 年 2 月に記載されており、http://phenix.int-evry.fr/jct/doc_end_user/documents/8_San%20Jose/wg11/JCTVC-H1003-v22.zip で入手可能である。

【0017】

【0032】 本開示の技法は一般に、3DV と呼ばれる HEVC ベースの 3D ビデオコーディングに関する。特に、本開示は、3DV におけるビュー合成モードを可能にするための技法を提供する。以下でさらに詳細に説明するように、HEVC では、ビデオエンコードおよびビデオデコードなどのビデオコードが、コーディングユニット (CU) と呼ばれるブロックベースのユニットを使用してピクチャをコーディングする。CU は、様々なコーディングモードを使用してコーディングされ得る。

【0018】

【0033】 視差補償予測 (DCP: disparity-compensated prediction) と呼ばれる 1 つのコーディングモードでは、(CU または CU の予測ユニット (PU) などの) ビデオデータのブロックが、視差ベクトルを使用して予測され得る。視差ベクトルは、同じ時間インスタンスからの異なるビューにおけるぴったり一致するブロックを、現在の CU を含むピクチャとして識別できる。「視差 (Disparity)」は一般に、オブジェクトの画像をキャ

10

20

30

40

50

ブチャまたは生成するときにカメラパースペクティブ(camera perspectives)の水平オフセットによって生じた1つのシーンにおける本質的に同じオブジェクトの2つのブロックの水平オフセットを指す。視差ベクトルは一般に、異なるビューにおけるピクチャに対するCUまたはCUのPUをコーディングするために使用されるベクトルに対応する。

【0019】

[0034] 視差ベクトル構成を行うための従来のHEVC技法は、いくつかの問題に遭遇している。例えば、動き推定領域(MER)が使用されるとき、空間的視差ベクトル(spatial disparity vector)を得るために使用される空間的隣接ブロックが、同じMER中にあるために利用不可能である可能性がある。この場合、最後の視差ベクトルは、利用不可能であるか、またはさほど正確ではないかのいずれかであり得る。また、いくつかのビュー間予測技法、例えば残差予測がCUレベルで行われ、いくつかのビュー間予測技法、例えば動き予測がPUレベルで行われる。そのため、視差ベクトル生成は、ブロックユニットの異なるレベルに適用される異なるビュー間予測技法に起因して、複数回行われる必要があり得る。視差動きベクトル予測が行われているCUまたはPUをMERが含むときに、MERが包含ブロックと呼ばれることがあり、視差動きベクトル予測が行われているPUをCUが含むときに、CUが包含ブロックと呼ばれることがある。

【0020】

[0035] その上、視差ベクトル構成がPUごとに必要とされる可能性があっても、このPUの時間的視差ベクトル(temporal disparity vectors)は、同じCU中の他のPUの時間的視差ベクトルと同様であり得る。さらに、時間的視差ベクトルを確認するとき、多数のPUが確認され得る。ビュー間参照ピクチャ(inter-view reference picture)のブロックが時間的と見なされ得るときでも、動きベクトルは必ずしも16×16ブロックユニットとして圧縮されとは限らない。

【0021】

[0036] 本開示の技法は、これらの問題のうちの1つまたは複数を克服することを目的とする。これらの技法は、現在の視差ベクトル生成技法を簡略化しかつ統一できる。いくつかの例で、MERが使用され、MERがPUよりも大きいとき、PUのビュー間予測は、現在のMERに隣接するブロックの空間的視差ベクトルに基づき得る。いくつかの例で、ビュー間予測がPUレベルで行われるときでも、視差ベクトル生成は、PUを包含するCUのために行われ得る。いくつかの例では、PUのためのより正確な視差ベクトルを生成するために、CUレベルで生成された視差ベクトルが改良され得る。改良は、現在のPUを包含するCUの空間的隣接ブロックをさらに確認することによってのみ起こり得る。代替または追加として、改良は、現在のPUを包含するMERの空間的隣接ブロックをさらに確認することのみによって起こり得る。代替または追加として、改良は、現在のPUの空間的隣接ブロックをさらに確認することのみによって起こり得る。

【0022】

[0037] 図1は、ビュー間予測のための視差ベクトルを生成するための技法を利用し得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(destination device)14によって後で復号されるべき符号化ビデオデータを与えるソースデバイス(source device)12を含む。特に、ソースデバイス12は、コンピュータ可読媒体16を介してビデオデータを宛先デバイス14に与える。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、ディスプレイデバイス、デジタルメディアプレーヤ、ビデオゲームコンソール、ビデオストリーミングデバイスなどを含む、広範囲のデバイスのいずれかを備え得る。場合によっては、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得る。

【0023】

[0038] 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得る。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 に符号化された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得る。一例で、コンピュータ可読媒体 16 は、ソースデバイス 12 が、符号化ビデオデータを宛先デバイス 14 にリアルタイムで直接送信することを可能にするための通信媒体を備え得る。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 に送信され得る。通信媒体は、無線周波数 (RF) スペクトルまたは 1 つもしくは複数の物理伝送線路など、任意のワイヤレスまたはワイヤード通信媒体を備え得る。通信媒体は、ローカルエリアネットワーク、ワイドエリアネットワーク、またはインターネットなどのグローバルネットワークなど、パケットベースネットワークの一部を形成し得る。通信媒体は、ソースデバイス 12 から宛先デバイス 14 への通信を可能にするために有用であり得るルータ、スイッチ、基地局、または任意の他の機器を含み得る。

10

20

30

40

50

【0024】

[0039] いくつかの例で、符号化データは、出力インターフェース 22 からストレージデバイスに出力され得る。同様に、符号化データは、入力インターフェースによってストレージデバイスからアクセスされ得る。ストレージデバイスは、ハードドライブ、ブルーレイ (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好適なデジタル記憶媒体など、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み得る。さらなる一例で、ストレージデバイスは、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し得るファイルサーバまたは別の中間ストレージデバイスに対応し得る。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ストレージデバイスから、記憶されたビデオデータにアクセスし得る。ファイルサーバは、符号化ビデオデータを記憶し、その符号化ビデオデータを宛先デバイス 14 に送信することが可能な任意のタイプのサーバであり得る。例示的なファイルサーバは、(例えば、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) 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、任意の標準のデータ接続 (standard data connection) を介して符号化ビデオデータにアクセスし得る。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好適であるワイヤレスチャネル (例えば、Wi-Fi (登録商標) 接続)、ワイヤード接続 (例えば、DSL、ケーブルモデムなど)、または両方の組合せを含み得る。ストレージ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り得る。

【0025】

[0040] 本開示の技法は、必ずしもワイヤレス適用例または設定に限定されとは限らない。本技法は、オーバージエアテレビジョン放送、ケーブルテレビジョン送信、衛星テレビジョン送信、動的適応ストリーミングオーバー HTTP (DASH: dynamic adaptive streaming over HTTP) などのインターネットストリーミングビデオ送信、データ記憶媒体上に符号化されたデジタルビデオ、データ記憶媒体に記憶されたデジタルビデオの復号、または他の適用例など、様々なマルチメディア適用例のいずれかをサポートするビデオコーディングに適用され得る。いくつかの例で、システム 10 は、ビデオストリーミング、ビデオ再生、ビデオブロードキャスト、および / またはビデオテレフォニーなどの適用例をサポートするために、一方向または双方向のビデオ送信をサポートするように構成され得る。

【0026】

[0041] 図 1 の例で、ソースデバイス 12 は、ビデオソース 18、ビデオエンコーダ 20、および出力インターフェース 22 を含む。宛先デバイス 14 は、入力インターフェース 28、ビデオデコーダ 30、およびディスプレイデバイス 32 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、ビュー間予測のための視差ベクトルを生成するための本技法を適用するように構成され得る。他の例で、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構成を含み得る。例えば、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得る。同様に、宛先デバイス 14 は、内蔵ディスプレイデバイスを含むのではなく、外部ディスプレイデバイスとインターフェースし得る。

【0027】

[0042] 図 1 の図示のシステム 10 は一例にすぎない。ビュー間予測のための視差ベクトルを生成するための技法は、任意のデジタルビデオ符号化および/または復号デバイスによって行われ得る。概して、本開示の技法はビデオ符号化デバイスによって行われるが、本技法は、一般に「コーデック」と呼ばれるビデオエンコーダ/デコーダによっても行われ得る。さらに、本開示の技法は、ビデオプリプロセッサによっても行われ得る。ソースデバイス 12 および宛先デバイス 14 は、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ド化ビデオデータを生成するような、コーディングデバイスの例にすぎない。いくつかの例で、デバイス 12、14 は、デバイス 12、14 の各々がビデオ符号化構成要素とビデオ復号構成要素とを含むように、実質的に対称的に動作し得る。従って、システム 10 は、例えば、ビデオストリーミング、ビデオ再生、ビデオブロードキャストまたはビデオ電話のための、ビデオデバイス 12 とビデオデバイス 14 との間の一方向または双方向のビデオ送信をサポートし得る。

10

【0028】

[0043] ソースデバイス 12 のビデオソース 18 は、ビデオカメラなどのビデオキャプチャデバイス、予めキャプチャされたビデオを含んでいるビデオアーカイブ、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ためのビデオフィードインターフェースを含み得る。さらなる代替として、ビデオソース 18 は、ソースビデオとしてのコンピュータグラフィックススペースのデータ、またはライブビデオとアーカイブビデオとコンピュータ生成ビデオとの組合せを生成し得る。場合によっては、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、いわゆるカメラフォンまたはビデオフォンを形成し得る。但し、上述のように、本開示で説明する技法は、概して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/またはワイヤード適用例に適用され得る。各場合において、キャプチャされたビデオ、以前にキャプチャされたビデオ、またはコンピュータ生成ビデオは、ビデオエンコーダ 20 によって符号化され得る。符号化されたビデオ情報は、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 上に出力され得る。

20

30

【0029】

[0044] コンピュータ可読媒体 16 は、ワイヤレスブロードキャストまたはワイヤードネットワーク送信などの一時媒体、またはハードディスク、フラッシュドライブ、コンパクトディスク、デジタルビデオディスク、ブルーレイディスク、または他のコンピュータ可読媒体などの記憶媒体（すなわち、非一時的記憶媒体）を含み得る。いくつかの例で、ネットワークサーバ（図示せず）は、例えば、ネットワーク送信を介して、ソースデバイス 12 から符号化されたビデオデータを受信し、宛先デバイス 14 に符号化されたビデオデータを与え得る。同様に、ディスクスタンピング設備など、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、その符号化ビデオデータを含んでいるディスクを生成し得る。従って、様々な例で、コンピュータ可読媒体 16 は、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得る。

40

【0030】

[0045] 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 から情報を受信する。コンピュータ可読媒体 16 の情報は、ビデオエンコーダ 20 によって定義され、またビデオデコーダ 30 によって使用される、ブロックおよび他のコード化ユニット、例えば、GOP の特性および/または処理を記述するシンタックス要素(syntax elements)を含む、シンタックス情報を含み得る。ディスプレイデバイス 32 は、復号されたビデオデータをユーザに対して表示し、陰極線管(CRT)、液晶ディスプレイ(

50

LCD)、プラズマディスプレイ、有機発光ダイオード(OLED)ディスプレイ、または別のタイプのディスプレイデバイスなど、様々なディスプレイデバイスのいずれかを備え得る。

【0031】

[0046] ビデオエンコーダ20およびビデオデコーダ30は、現在開発中の高効率ビデオコーディング(HEVC)規格などのビデオコーディング規格に従って動作し得、HEVCテストモデル(HM)に準拠し得る。代替的に、ビデオエンコーダ20およびビデオデコーダ30は、代替的にMPEG-4, Part 10, Advanced Video Coding(AVC)と呼ばれるITU-T H.264規格など、他のプロプライエタリ規格もしくは工業規格、またはそのような規格の拡張に従って動作し得る。但し、本開示の技法は、いかなる特定のコーディング規格にも限定されない。ビデオコーディング規格の他の例としては、MPEG-2およびITU-T H.263がある。図1には示されていないが、いくつかの態様で、ビデオエンコーダ20およびビデオデコーダ30は、それぞれオーディオエンコーダおよびオーディオデコーダと統合され得、適切なMUX-DEMUXユニット、または他のハードウェアおよびソフトウェアを含んで、共通のデータストリームまたは別個のデータストリーム中のオーディオとビデオの両方の符号化を処理し得る。適用可能な場合、MUX-DEMUXユニットは、ITU H.223マルチプレクサプロトコル、またはユーザデータグラムプロトコル(UDP)などの他のプロトコルに準拠し得る。

10

【0032】

[0047] ITU-T H.264/MPEG-4(AVC)規格は、Joint Video Team(JVT)として知られる共同パートナーシップの成果として、ISO/IEC Moving Picture Experts Group(MPEG)とともにITU-T Video Coding Experts Group(VCEG)によって策定された。いくつかの態様で、本開示で説明する技法は、H.264規格に概して準拠するデバイスに適用され得る。H.264規格は、ITU-T Study Groupによる2005年3月付けのITU-T勧告H.264「Advanced Video Coding for generic audiovisual services」に記載されており、本明細書でH.264規格もしくはH.264仕様、またはH.264/AVC規格もしくは仕様と呼ぶことがある。Joint Video Team(JVT)はH.264/MPEG-4 AVCへの拡張に取り組んでいる。

20

30

【0033】

[0048] ビデオエンコーダ20およびビデオデコーダ30はそれぞれ、1つまたは複数のマイクロプロセッサ、デジタル信号プロセッサ(DSP)、特定用途向け集積回路(ASIC)、フィールドプログラマブルゲートアレイ(FPGA)、ディスクリート論理、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任意の組合せなど、様々な好適なエンコーダ回路のいずれかとして実装され得る。本技法が部分的にソフトウェアで実装されるとき、デバイスは、好適な非一時的コンピュータ可読媒体にソフトウェアの命令を記憶し、1つまたは複数のプロセッサを使用してその命令をハードウェアで実行して、本開示の技法を行い得る。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中に含まれ得、そのいずれも、該当のデバイスにおいて複合エンコーダ/デコーダ(コーデック)の一部として統合され得る。

40

【0034】

[0049] JCT-VCは、HEVC規格の開発に取り組んでいる。HEVC規格化の取り組みは、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発展的モデルに基づく。HMは、例えば、ITU-T H.264/AVCに従う既存のデバイスに対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仮定する。例えば、H.264は9つのイントラ予測符号化モードを提供するが、HMは33個ものイントラ予測符号化モードを提供し得る。

50

【 0 0 3 5 】

[0050] 概して、H Mの作業モデルは、ビデオフレームまたはピクチャが、「コーディングツリーユニット」または「C T U」とも呼ばれる、ルーマサンプル(luma sample)とクロマサンプル(chroma sample)の両方を含む一連のツリーブロックまたは最大コーディングユニット(L C U)に分割され得ることを記載している。ビットストリーム内のシンタックスデータが、ピクセルの数に関して最大コーディングユニットであるL C Uのサイズを定義し得る。スライスは、コーディング順序でいくつかの連続するツリーブロックを含む。ビデオフレームまたはピクチャは、1つまたは複数のスライスに区分され得る。各ツリーブロックは、4分木に従ってコーディングユニット(C U)に分割され得る。概して、4分木データ構造はC U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。C Uが4つのサブC Uに分割された場合、C Uに対応するノードは4つのリーフノード(leaf nodes)を含み、リーフノードの各々はサブC Uのうちの1つに対応する。

10

【 0 0 3 6 】

[0051] 4分木データ構造の各ノードは、対応するC Uのシンタックスデータを与え得る。例えば、4分木のノードは、そのノードに対応するC UがサブC Uに分割されるかどうかを示す分割フラグを含み得る。C Uのシンタックス要素は、再帰的に定義され得、かつC UがサブC Uに分割されるかどうかに依存し得る。C Uがさらに分割されない場合、そのC UはリーフC Uと呼ばれる。本開示では、元のリーフC Uの明示的分割が存在しない場合でも、リーフC Uの4つのサブC UをリーフC Uとも呼ぶ。例えば、16×16サイズのC Uがさらに分割されない場合、この16×16C Uが決して分割されなくても、4つの8×8サブC UをリーフC Uとも呼ぶ。

20

【 0 0 3 7 】

[0052] C Uは、C Uがサイズの差異を有さないことを除いて、H . 2 6 4規格のマクロブロックと同様の目的を有する。例えば、ツリーブロックは、4つの子ノード(サブC Uとも呼ばれる)に分割され得、各子ノードは、今度は親ノードとなり、別の4つの子ノードに分割され得る。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C Uとも呼ばれるコーディングノードを備える。コード化ビットストリームに関連するシンタックスデータは、最大C U深さと呼ばれる、ツリーブロックが分割され得る最大回数を定義し得、また、コーディングノードの最小サイズを定義し得る。それに

30

【 0 0 3 8 】

[0053] C Uは、コーディングノードと、コーディングノードに関連する予測ユニット(P U : prediction unit)および変換ユニット(T U : transform unit)とを含む。C Uのサイズは、コーディングノードのサイズに対応し、かつ形状が方形でなければならない。C Uのサイズは、8×8ピクセルから最大64×64以上のピクセルを有するツリーブロックのサイズまでに及び得る。各C Uは、1つまたは複数のP Uと、1つまたは複数のT Uとを含み得る。C Uに関連するシンタックスデータは、例えば、C Uを1つまたは複数のP Uに区分することを記述し得る。区分モードは、C Uが、スキップモード符号化もしくはダイレクトモード符号化されるか、イントラ予測モード符号化されるか、またはインター予測モード符号化されるかによって異なり得る。P Uは、形状が非方形になるように区分され得る。C Uに関連するシンタックスデータは、例えば、4分木に従って、C Uを1つまたは複数のT Uに区分することも記述し得る。T Uは、形状が方形または非方形(例えば、矩形)であり得る。

40

【 0 0 3 9 】

[0054] H E V C規格は、C Uごとに異なり得るT Uに従う変換を可能にする。T Uは

50

、一般に、区分されたLCUについて定義された所与のCU内のPUのサイズに基づいてサイズ決定されるが、常にそうであるとは限らない。TUは、一般にPUと同じサイズであるかまたはPUよりも小さい。いくつかの例で、CUに対応する残差サンプルは、「残差クワッドツリー」(RQT: residual quad tree)として知られるクワッドツリー構造を使用して、より小さいユニットに再分割され得る。RQTのリーフノードは変換ユニット(TU)と呼ばれることがある。TUに関連するピクセル差分値は、量子化され得る変換係数を生成するために変換され得る。

【0040】

[0055] リーフCUは、1つまたは複数の予測ユニット(PU)を含み得る。概して、PUは、対応するCUの全部または一部分に対応する空間的エリアを表し、そのPUの参照サンプルを取り出すためのデータを含み得る。その上、PUは、予測に関係するデータを含む。例えば、PUがイントラモード符号化される時、PUについてのデータは、PUに対応するTUについてのイントラ予測モードを記述するデータを含み得る残差4分木(RQT)中に含まれ得る。別の例として、PUがインターモード符号化される時、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み得る。PUの動きベクトルを定義するデータは、例えば、動きベクトルの水平成分、動きベクトルの垂直成分、動きベクトルの解像度(例えば、1/4ピクセル精度または1/8ピクセル精度)、動きベクトルが指す参照ピクチャ、および/または動きベクトルの参照ピクチャリスト(例えば、リスト0、リスト1、またはリストC)を記述し得る。

【0041】

[0056] 1つまたは複数のPUを有するリーフCUはまた、1つまたは複数の変換ユニット(TU)を含み得る。変換ユニットは、上記で説明したように、(TU4分木構造とも呼ばれる)RQTを使用して指定され得る。例えば、分割フラグは、リーフCUが4つの変換ユニットに分割されるかどうかを示し得る。次いで、各変換ユニットは、さらに、さらなるサブTUに分割され得る。TUがさらに分割されないとき、そのTUはリーフTUと呼ばれることがある。概して、イントラコーディングの場合、リーフCUに属するすべてのリーフTUは同じイントラ予測モードを共有する。すなわち、概して、リーフCUのすべてのTUの予測値を計算するために同じイントラ予測モードが適用される。イントラコーディングの場合、ビデオエンコーダは、イントラ予測モードを使用して各リーフTUの残差値をTUに対応するCUの一部と元のブロックとの間の差分として計算し得る。TUは、必ずしもPUのサイズに制限されるとは限らない。従って、TUはPUよりも大きくまたは小さくなり得る。イントラコーディングの場合、PUは、同じCUの対応するリーフTUとコロケートされ得る。いくつかの例で、リーフTUの最大サイズは、対応するリーフCUのサイズに対応し得る。

【0042】

[0057] さらに、リーフCUのTUはまた、残差4分木(RQT)と呼ばれる、それぞれの4分木データ構造に関連付けられ得る。すなわち、リーフCUは、リーフCUがどのようにTUに区分されるかを示す4分木を含み得る。TU4分木のルートノードは概してリーフCUに対応し、CU4分木のルートノードは概してツリーブロック(またはLCU)に対応する。分割されないRQTのTUはリーフTUと呼ばれる。概して、本開示では、特に明記しない限り、リーフCUおよびリーフTUに言及するためにそれぞれCUおよびTUという用語を使用する。

【0043】

[0058] ビデオシーケンスは、一般に、一連のビデオフレームまたはピクチャを含む。ピクチャグループ(GOP)は、概して、ビデオピクチャのうちの一連の1つまたは複数を用意する。GOPは、GOP中に含まれるいくつかのピクチャを記述するシンタックスデータを、GOPのヘッダ中、ピクチャのうち1つまたは複数のヘッダ中、または他の場所を含み得る。ピクチャの各スライス、該当のスライスの符号化モードを記述するスライスシンタックスデータを含み得る。ビデオエンコーダ20は、一般に、ビデオデータを符号化するために個々のビデオスライス内のビデオブロックに対して動作する。ビデオブ

ロックは、CU内のコーディングノードに対応し得る。ビデオブロックは、固定サイズまたは可変サイズを有し得、かつ指定のコーディング規格に応じてサイズが異なり得る。

【0044】

[0059] 一例として、HMは、様々なPUサイズでの予測をサポートする。特定のCUのサイズが $2N \times 2N$ であると仮定すると、HMは、 $2N \times 2N$ または $N \times N$ のPUサイズでのイントラ予測をサポートし、 $2N \times 2N$ 、 $2N \times N$ 、 $N \times 2N$ 、または $N \times N$ の対称的なPUサイズでのインター予測をサポートする。HMはまた、 $2N \times nU$ 、 $2N \times nD$ 、 $nL \times 2N$ 、および $nR \times 2N$ のPUサイズでのインター予測のための非対称区分をサポートする。非対称区分で、CUの一方は区分されないが、他の方向は25%と75%とに区分される。25%の区分に対応するCUの部分は、「n」とその後ろに付く「Up」、「Down」、「Left」、または「Right」という表示によって示される。従って、例えば、「 $2N \times nU$ 」は、上部の $2N \times 0.5N$ PUと下部の $2N \times 1.5N$ PUとで水平方向に区分された $2N \times 2N$ CUを指す。

【0045】

[0060] 本開示で、「 $N \times N$ (NxN)」および「 $N \times N$ (N by N)」は、垂直寸法および水平寸法に関するビデオブロックのピクセル寸法、例えば、 16×16 (16x16) ピクセルまたは 16×16 (16 by 16) ピクセルを指すために互換的に使用され得る。概して、 16×16 ブロックは、垂直方向に16ピクセルを有し($y = 16$)、水平方向に16ピクセルを有する($x = 16$)。同様に、 $N \times N$ ブロックは、概して、垂直方向にNピクセルを有し、水平方向にNピクセルを有し、但し、Nは非負整数値を表す。ブロック中のピクセルは行と列で構成され得る。さらに、ブロックは、必ずしも、水平方向に垂直方向と同じ数のピクセルを有する必要はない。例えば、ブロックは $N \times M$ ピクセルを備え得、但し、Mは必ずしもNに等しいとは限らない。

【0046】

[0061] CUのPUを使用したイントラ予測コーディングまたはインター予測コーディングの後、ビデオエンコーダ20は、CUのTUのための残差データを計算し得る。PUは、(ピクセル領域(pixel domain)とも呼ばれる)空間領域(spatial domain)において予測ピクセルデータを生成する方法またはモードを記述するシンタックスデータを備え得、TUは、変換、例えば、残差ビデオデータへの離散コサイン変換(DCT: discrete cosine transform)、整数変換(integer transform)、ウェーブレット変換(wavelet transform)、または概念的に同様の変換の適用後に、変換領域(transform domain)において係数を備え得る。残差データは、符号化されていないピクチャのピクセルと、PUに対応する予測値との間のピクセル差分に対応し得る。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、TUを変換して、CUの変換係数(transform coefficients)を生成し得る。

【0047】

[0062] 変換係数を生成するための任意の変換の後に、ビデオエンコーダ20は、変換係数の量子化を行い得る。量子化は、概して、さらなる圧縮を提供する、係数を表すために使用されるデータの量をできるだけ低減するために変換係数を量子化するプロセスを指す。量子化プロセスは、係数の一部または全部に関連するビット深度(bit depth)を低減し得る。例えば、量子化中にnビット値がmビット値に切り捨てられ得、但し、nはmよりも大きい。

【0048】

[0063] 量子化の後に、ビデオエンコーダは、変換係数を走査して、量子化変換係数を含む2次元行列から1次元ベクトルを生成し得る。走査は、より高いエネルギー(従ってより低い周波数)の係数をアレイの前方に配置し、より低いエネルギー(従ってより高い周波数)の係数をアレイの後方に配置するように設計され得る。いくつかの例で、ビデオエンコーダ20は、エントロピー符号化され得る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予め定義された走査順序を利用し得る。他の例で、ビデオエンコーダ20は適応型走査を行い得る。量子化変換係数を走査して1次元ベクトルを

形成した後に、ビデオエンコーダ 20 は、例えば、コンテキスト適応型可変長コーディング (C A V L C : context-adaptive variable length coding)、コンテキスト適応型バイナリ算術コーディング (C A B A C : context-adaptive binary arithmetic coding)、シンタックスベースコンテキスト適応型バイナリ算術コーディング (S B A C : syntax-based context-adaptive binary arithmetic coding)、確率間隔区分エントロピー (P I P E : 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得る。ビデオエンコーダ 20 はまた、ビデオデータを復号する際にビデオデコーダ 30 が使用するための符号化ビデオデータに関連するシンタックス要素をエントロピー符号化し得る。

【0049】

10

[0064] C A B A C を行うために、ビデオエンコーダ 20 は、送信されるべきシンボルに、コンテキストモデル内のコンテキストを割り当て得る。コンテキストは、例えば、シンボルの隣接値が非 0 であるか否かに関係し得る。C A V L C を行うために、ビデオエンコーダ 20 は、送信されるべきシンボルのための可変長コードを選択し得る。V L C におけるコードワードは、比較的短いコードが優勢シンボル (more probable symbols) に対応し、より長いコードが劣勢シンボル (less probable symbols) に対応するように構成され得る。このようにして、V L C の使用は、例えば、送信されるべき各シンボルのために等長コードワード (equal-length codewords) を使用するよりも、ビット節約を達成し得る。確率判断は、シンボルに割り当てられたコンテキストに基づき得る。

【0050】

20

[0065] ビデオエンコーダ 20 およびビデオデコーダ 30 は、本開示の技法のいずれかまたはすべてを行うように構成され得る。例えば、ビデオエンコーダ 20 およびビデオデコーダ 30 は、動き推定領域 (M E R) を使用して並列にビデオデータをコーディングするように構成され得る。本開示の技法によれば、ビデオエンコーダ 20 およびビデオデコーダ 30 が、現在コーディングされている P U よりも大きい M E R を使用するとき、ビデオエンコーダ 20 およびビデオデコーダ 30 は、現在の M E R 内のブロックではなく、その M E R に隣接するブロックの空間的視差ベクトルに基づいて P U のビュー間予測を行うように構成され得る。そのような技法の例が、図 8 ~ 図 10 に関して以下でより詳細に説明される。

【0051】

30

[0066] 追加または代替として、ビデオエンコーダ 20 およびビデオデコーダ 30 が、P U レベルでビュー間予測を行うように構成されるとき、それでもビデオエンコーダ 20 およびビデオデコーダ 30 は、P U を包含する C U のための視差ベクトル生成を行い得る。そのような技法の例が、図 11 に関して以下でより詳細に説明される。同様に、ビデオエンコーダ 20 およびビデオデコーダ 30 は、P U のためのより正確な視差ベクトルを生成するために、C U レベルで生成された視差ベクトルを改良するように構成され得る。例えば、ビデオエンコーダ 20 およびビデオデコーダ 30 は、改良のために、現在の P U を包含する C U の空間的隣接ブロックを確認できる。別の例として、ビデオエンコーダ 20 およびビデオデコーダ 30 は、改良のために、現在の P U を包含する M E R の空間的隣接ブロックを確認できる。また別の例として、ビデオエンコーダ 20 およびビデオデコーダ 30 は、改良のために、現在の P U の空間的隣接ブロックを確認できる。

40

【0052】

[0067] 従って、ビデオエンコーダ 20 は、視差動きベクトルを使用して、包含ブロック内のビデオデータの現在のブロックを符号化するように構成され得る。例えば、ビデオエンコーダ 20 は、ビュー間予測を使用して現在のブロックを符号化することが最良の、または少なくとも許容できるレートひずみ最適化 (R D O : rate-distortion optimization) 特性をもたらすと判断し得る。その上、現在のブロックは、動き推定領域 (M E R) などの包含ブロック内で生じ得る。代替として、現在のブロックは、C U 内で生じる P U を備え得る。

【0053】

50

[0068] いずれの場合にも、現在のブロックの隣接ブロックも包含ブロック内にあるとの判断に基づいて、ビデオエンコーダ20は、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替できる。すなわち、動きベクトル符号化（例えば、ビュー間予測の場合の視差動きベクトル符号化）を行う目的で、ビデオエンコーダ20は、動きベクトル予測子候補のリストを含む候補リストを構成できる。概して、ビデオエンコーダ20は、リストに含める候補動きベクトル予測子として、現在のブロックの空間的隣接ブロックおよび/または時間的隣接ブロックから動きベクトルを選択できる。本開示の技法によれば、ビデオエンコーダ20は、包含ブロック内のビデオデータの並列符号化（および同様に、並列復号）を可能にするために、包含ブロック内のブロックの動きベクトルを、包含ブロック外のブロックの動きベクトルで代替できる。

10

【0054】

[0069] ビデオエンコーダ20は次いで、候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択できる。例えば、ビデオエンコーダ20は、（例えば、ベクトル差分計算によって割り出されるような）現在のブロックの視差動きベクトルと最もぴったり一致するリスト中の候補動きベクトル予測子のうちの1つを選択できる。ビデオエンコーダ20は次いで、視差動きベクトル予測子に基づいて視差動きベクトルを符号化できる。例えば、ビデオエンコーダ20は、マージモード(merge mode)または高度動きベクトル予測（AMVP: advanced motion vector prediction）技法を使用して視差動きベクトルを符号化し得る。

20

【0055】

[0070] マージモードで、ビデオエンコーダ20は、動き情報の取り出し元になる隣接ブロックを識別するマージインデックス値を、その隣接ブロックの動き情報を現在のブロックに関する動き情報としてビデオデコーダ30に使用させるために、シグナリングできる。本開示の技法によれば、シグナリングされた隣接ブロックが包含ブロックの外にある場合、そのようなシグナリングは、ビデオデコーダ30に、現在のブロックに関する動き情報として使用するために、代替ブロックに関する動き情報を取り出すことを促し得る。マージインデックスに対応するブロックから継承された動き情報は、参照ピクチャリストと参照ピクチャに対応する参照ピクチャインデックスとをさらに含み得る。HEVCに対する高レベルシンタックス（HLS）のみの拡張を利用するために、ビデオエンコーダ20は、長期参照ピクチャとしてビュー間参照ピクチャを、および短期参照ピクチャとしてビュー内参照ピクチャ(intra-view reference pictures)をシグナリングするように構成されてよく、この場合に参照ピクチャインデックスおよび参照ピクチャリストのシグナリングは、単一のビューコーディングのためのHEVCの技法と実質的に同じまたは同一のままであり得る。

30

【0056】

[0071] AMVPで、ビデオエンコーダ20は、マージモードの場合のマージインデックスと実質的に同じ方法で隣接ブロックに対応し得る、動き情報の取り出し元になる隣接ブロックを識別するAMVPインデックスをシグナリングできる。但し、ビデオエンコーダ20はさらに、視差動きベクトルとAMVPインデックスに対応するブロックの動きベクトル予測子との間の差分を表す動きベクトル差分値を符号化できる。本開示の技法によれば、AMVPインデックスが、同じく包含ブロック内にある現在のブロックの隣接ブロックに対応するとき、動きベクトル予測子を定義する動き情報は、実際には、包含ブロック外のブロックから取り出され得る。ビデオエンコーダ20はまた、参照ピクチャリストと参照インデックスとを表すデータをシグナリングし得る。その上、ビデオデコーダ30は、視差動きベクトルを使用して現在のブロックを符号化できる。

40

【0057】

[0072] 従って、ビデオデコーダ30は、ビデオエンコーダ20の技法とは実質的に逆の技法を行うように構成され得る。例えば、ビデオデコーダ30は、包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断することと、現在のブロックの隣接ブロックも包含ブロック内にあるとの判断に基づいて

50

、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替することと、候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、視差動きベクトル予測子に基づいて視差動きベクトルを復号することとを行うように構成され得る。

【0058】

[0073] 例えば、ビデオデコーダ30は、現在のブロックが視差動きベクトルを使用して符号化されていることを示すシグナリングデータを受信し得る。ビデオデコーダ30はさらに、包含ブロック内の現在のブロックの位置、並びに現在のブロックの隣接ブロックの位置を、候補リストに隣接ブロックを追加するか、または包含ブロックの外にあって包含ブロックに隣接するブロックを追加するかを決定するために、判断できる。

10

【0059】

[0074] ビデオデコーダ30は、視差動きベクトルを復号するために隣接ブロックのうちのいずれを使用するかを識別する識別子（例えば、マージインデックスまたはAMVPインデックス）を受信し得る。本開示の技法によれば、シグナリングされた隣接ブロックが、現在のブロックを包含する包含ブロック内にある場合、ビデオデコーダ30は、実際には、包含ブロックの外にある包含ブロックの隣接ブロックから動き情報を取り出し得る。ビデオデコーダ30は次いで、識別子に対応するブロックから取り出された動きベクトル予測子を使用して視差動きベクトルを復号できる。同様に、ビデオデコーダ30は、視差動きベクトルを使用して現在のブロックを復号できる。

【0060】

20

[0075] ビデオエンコーダ20およびビデオデコーダ30はそれぞれ、適用可能なとき、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)、ディスプレイ論理回路、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任意の組合せなど、様々な好適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得る。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中に含まれ得、そのいずれも複合ビデオエンコーダ/デコーダ(コーデック)の一部として統合され得る。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得る。

30

【0061】

[0076] 図2は、ビュー間予測のための視差ベクトルを生成するための技法を実装し得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は、ビデオスライス内のビデオブロックの時間的インタコーディングと時間的インターコーディングとビュー間コーディングとを行い得る。時間的インタコーディングは、所与のビデオフレームまたはピクチャ内のビデオの空間的冗長性を低減または除去するために空間的予測に依拠する。時間的インターコーディングは、ビデオシーケンスの隣接フレームまたはピクチャ内のビデオの時間的冗長性を低減または除去するために時間的予測に依拠する。ビュー間コーディングは、パースペクティブによって水平にシフトされ得る別のビューからのピクチャの冗長性を低減または除去するためにビュー間予測に依拠する。時間的インタラモード(Iモード)は、いくつかの空間ベースのコーディングモードのいずれかを指し得る。単方向予測(Pモード)または双方向予測(Bモード)などの時間的インターモードは、いくつかの時間ベースのコーディングモードのいずれかを指し得る。ビュー間モードは、視差動きベクトルまたは他のそのようなビュー間コーディング技法の使用を指し得る。一般に、ビュー間予測は、時間的インターコーディングと実質的に同様の方法で行われ得る。

40

【0062】

[0077] 図2に示すように、ビデオエンコーダ20は、符号化されるべきビデオフレーム内の現在のビデオブロックを受信する。図2の例で、ビデオエンコーダ20は、モード選択ユニット40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

50

と、量子化ユニット 5 4 と、エントロピーエンコーディングユニット 5 6 とを含む。モード選択ユニット 4 0 は、今度は、動き補償ユニット 4 4 と、動き推定ユニット 4 2 と、イントラ予測ユニット 4 6 と、パーティションユニット 4 8 とを含む。ビデオブロック再構成のために、ビデオエンコーダ 2 0 はまた、逆量子化ユニット 5 8 と、逆変換ユニット 6 0 と、加算器 6 2 とを含む。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(blockiness artifacts)を除去するためにブロック境界をフィルタ処理するデブロッキングフィルタ(deblocking filter) (図 2 に図示せず) も含まれ得る。所望される場合、デブロッキングフィルタは、一般に、加算器 6 2 の出力をフィルタ処理することになる。また、デブロッキングフィルタに加えて追加のフィルタ(ループ内またはループ後)が使用され得る。そのようなフィルタは、簡潔のために示されていないが、所望される場合、(ループ内フィルタとして) 加算器 5 0 の出力をフィルタ処理し得る。

10

【0063】

[0078] 符号化プロセス中に、ビデオエンコーダ 2 0 は、コーディングされるべきビデオフレームまたはスライスを受信する。フレームまたはスライスは複数のビデオブロックに分割され得る。動き推定ユニット 4 2 および動き補償ユニット 4 4 は、時間的予測を行うために、1 つまたは複数の参照フレーム中の 1 つまたは複数のブロックに対して、受信されたビデオブロックのインター予測コーディング(時間的かビュー間か)を行う。イントラ予測ユニット 4 6 は、代替的に、空間的予測を行うために、コーディングされるべきブロックと同じフレームまたはスライス中の 1 つまたは複数の隣接ブロックに対して受信されたビデオブロックのイントラ予測コーディングを行い得る。ビデオエンコーダ 2 0 は、例えば、ビデオデータのブロックごとに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行い得る。

20

【0064】

[0079] その上、パーティションユニット 4 8 は、前のコーディングパスにおける前の区分方式(previous partitioning schemes)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区分し得る。例えば、パーティションユニット 4 8 は、初めにフレームまたはスライスを LCU に区分し、レートひずみ分析(例えば、レートひずみ最適化)に基づいて LCU の各々をサブCU に区分し得る。モード選択ユニット 4 0 は、LCU をサブCU に区分することを示す 4 分木データ構造をさらに生成し得る。4 分木のリーフノードCU は、1 つまたは複数のPU および 1 つまたは複数のTU を含み得る。

30

【0065】

[0080] モード選択ユニット 4 0 は、例えば、誤差結果に基づいて、コーディングモード、(時間的イントラコーディング、時間的インターコーディング、またはビュー間コーディング)のうちの 1 つを選択でき、残差ブロックデータを生成するために、得られたイントラコーディングされたブロックまたはインターコーディングされた(時間的またはビュー間)ブロックを加算器 5 0 に与え、かつ参照フレームとして使用するための符号化されたブロックを復元するために、得られたイントラコーディングされたブロックまたはインターコーディングされたブロックを加算器 6 2 に与える。モード選択ユニット 4 0 はまた、動きベクトル(例えば、時間的動きベクトルまたは視差動きベクトル)、イントラモードインジケータ、パーティション情報、および他のそのようなシンタックス情報などのシンタックス要素をエントロピー符号化ユニット 5 6 に与える。

40

【0066】

[0081] 動き推定ユニット 4 2 および動き補償ユニット 4 4 は、高度に統合され得るが、概念的な目的のために別々に示してある。動き推定ユニット 4 2 によって行われる動き推定は、ビデオブロックの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。視差動きベクトルは、一般に、水平シフト、すなわち、異なるビューのやや異なるカメラパースペクティブから生じる、異なるビューからの対応するピクチャの 2 つのブロックの間の視差を表す。動きベクトルは、例えば、現在のフレーム(または他のコード化ユニット)内でコーディングされている現在のブロックに対する参照フレーム(または他のコード化ユニット)内の予測ブロックに対する現在のビデオフレームまたはピクチャ内のビデオブ

50

ロックのPUの変位を示し得る。予測ブロックは、絶対値差分和 (SAD: sum of absolute difference)、2乗差分和 (SSD: sum of square difference)、または他の差分メトリックによって判断され得るピクセル差分に関して、コーディングされるべきブロックにぴったり一致することがわかるブロックである。いくつかの例で、ビデオエンコーダ20は、参照ピクチャメモリ64に記憶された参照ピクチャのサブ整数ピクセル位置(sub-integer pixel positions)の値を計算し得る。例えば、ビデオエンコーダ20は、参照ピクチャの1/4ピクセル位置、1/8ピクセル位置、または他の分数ピクセル位置(fractional pixel positions)の値を補間し得る。従って、動き推定ユニット42は、フルピクセル位置と分数ピクセル位置とに対する動き探索を行い、分数ピクセル精度で動きベクトルを出力し得る。

10

【0067】

[0082] 動き推定ユニット42は、PUの位置を参照ピクチャの予測ブロックの位置と比較することによって、インターコード化(時間的またはビュー間)スライスにおけるビデオブロックのPUのための動きベクトルを計算する。参照ピクチャは、第1の参照ピクチャリスト(リスト0)または第2の参照ピクチャリスト(リスト1)から選択され得、それらの参照ピクチャリストの各々は、参照ピクチャメモリ64に記憶された1つまたは複数の参照ピクチャを識別する。いくつかの例で、ビュー間参照ピクチャは長期参照ピクチャとしてシグナリングされてよく、時間的参照ピクチャは短期参照ピクチャとしてシグナリングされてよい。代替として、ビュー間参照ピクチャは短期参照ピクチャとしてシグナリングされてよく、時間的参照ピクチャは長期参照ピクチャとしてシグナリングされてよい。動き推定ユニット42は、計算された動きベクトルをエントロピー符号化ユニット56と動き補償ユニット44とに送る。

20

【0068】

[0083] 動き補償ユニット44によって行われる動き補償は、動き推定ユニット42によって判断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に関与し得る。この場合も、いくつかの例で、動き推定ユニット42および動き補償ユニット44は機能的に統合され得る。現在のビデオブロックのPUについての動きベクトルを受信すると、動き補償ユニット44は、動きベクトルが参照ピクチャリストのうちの1つにおいて指す予測ブロックの位置を特定し得る。加算器50は、以下で説明するように、コーディングされている現在のビデオブロックのピクセル値から予測ブロックのピクセル値を減算し、ピクセル差分値を形成することによって、残差ビデオブロックを形成する。概して、動き推定ユニット42はルーマ成分に対して動き推定を行い、かつ動き補償ユニット44は、クロマ成分とルーマ成分の両方のためにルーマ成分に基づいて計算された動きベクトルを使用する。モード選択ユニット40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が使用するためのビデオブロックとビデオスライスとに関連するシンタックス要素を生成し得る。

30

【0069】

[0084] さらに、ビデオエンコーダ20は、例えば、マージモードまたはAMVPのうちの1つを使用して、動きベクトルを符号化できる。本開示の技法によれば、現在のブロックが包含ブロック(例えば、MER)内に包含される場合、包含ブロックが並列にコーディングされることが可能な領域を定義すると仮定すると、現在のブロックの隣接ブロックが包含ブロック内にあるとき、ビデオエンコーダ20は、現在のブロックに包含ブロックの隣接ブロックの動きベクトルを適用できる。すなわち、現在のブロックが包含ブロック内にあるとき、また現在のブロックに隣接するブロックも包含ブロック内にあるとき、ビデオエンコーダ20は、動きベクトルコーディングの目的で、現在のブロックの隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替できる。

40

【0070】

[0085] 本開示による動きベクトルコーディング技法の説明の目的で、「隣接ブロック」という用語は、現在のブロックを包含する包含ブロックの外に隣接ブロックがあるときには、現在のブロックに隣接するブロックを指すと理解されるべきであるが、別様に隣接

50

ブロックが包含ブロック内にあるときには、包含ブロックに隣接し、包含ブロックの外にあるブロックを指すと理解されるべきである。マージモードで、ビデオエンコーダ 20 は、現在のブロック（例えば、現在の P U）に隣接ブロックの動きベクトルを適用できる。A M V P モードで、ビデオエンコーダ 20 は、現在のブロックの動きベクトルと隣接ブロックの動きベクトルとの間の動きベクトル差分を計算でき、エントロピー符号化ユニット 56 は、動きベクトル差分を表すデータを符号化できる。その上、マージモードで、エントロピー符号化ユニット 56 は、隣接ブロックを識別するマージインデックスを定義するデータを符号化でき、A M V P で、エントロピー符号化ユニット 56 は、隣接ブロックを識別する A M V P インデックスを定義するデータを符号化できる。マージインデックスまたは A M V P インデックスは、隣接ブロック識別子の例を表す。

10

【0071】

[0086] より詳細には、ビデオエンコーダ 20 は、現在のブロック（例えば、現在の P U）の 1 つまたは複数の空間ネイバーに関する識別子を含む候補リストを構成できる。空間ネイバーの例は以下の図 5 に示されている。但し、現在のブロックが包含ブロック内にある（また包含ブロックが、並列コーディングを許容する領域を定義する）と仮定すると、ビデオエンコーダ 20 は、同じく包含ブロック内にある現在のブロックの隣接ブロックのいずれかに対する代替を決定できる。代替方式の例は図 8 ~ 図 11 に示されている。

【0072】

[0087] 概して、ビデオエンコーダ 20 は、包含ブロック内にある現在のブロックの隣接ブロックと同様に位置付けられる包含ブロック外のブロックで代替できる。例えば、いくつかの例で、包含ブロック内の隣接ブロックが現在のブロックの特定の隅にある場合、代替ブロックは、包含ブロックの同じ隅にあり得る。別の例として、隣接ブロックが現在のブロックの左側にある場合、代替ブロックは、同じ垂直位置であるが、包含ブロックの外であるが包含ブロックの隣に代替ブロックを配置する水平位置を有し得る。同様に、隣接ブロックが現在のブロックの上側にある場合、代替ブロックは、同じ水平位置であるが、包含ブロックの外であるが包含ブロックの隣に代替ブロックを配置する垂直位置を有し得る。ビデオエンコーダ 20 は、包含ブロック外の隣接ブロックを表すデータを、隣接ブロックが包含ブロック内にあって現在のブロックに隣接しているかのように符号化できる。但し、図 3 に関して以下で説明するように、ビデオデコーダ 30 は、隣接ブロックが本来であれば包含ブロック内にある場合に、コード化データが包含ブロック外の隣接ブロックに対応すると判断するように構成され得る。

20

30

【0073】

[0088] イントラ予測ユニット 46 は、上記で説明したように、動き推定ユニット 42 および動き補償ユニット 44 によって行われるインター予測の代替として、現在のブロックをイントラ予測し得る。特に、イントラ予測ユニット 46 は、現在のブロックを符号化するために使用すべきイントラ予測モードを判断し得る。いくつかの例で、イントラ予測ユニット 46 は、例えば、別個の符号化パス中に、様々なイントラ予測モードを使用して現在のブロックを符号化し得、イントラ予測ユニット 46（または、いくつかの例で、モード選択ユニット 40）は、テストされたモードから使用するのに適切なイントラ予測モードを選択し得る。

40

【0074】

[0089] 例えば、イントラ予測ユニット 46 は、様々なテストされたイントラ予測モードのためのレートひずみ分析を使用してレートひずみ値を計算し、テストされたモードの中で最良のレートひずみ特性を有するイントラ予測モードを選択し得る。レートひずみ分析は、概して、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符号化されていないブロックとの間のひずみ（または誤差）の量、並びに符号化ブロックを生成するために使用されるビットレート（すなわち、ビット数）を判断する。イントラ予測ユニット 46 は、どのイントラ予測モードがブロックについて最良のレートひずみ値を呈するかを判断するために、様々な符号化ブロックのひずみおよびレートから比率を計算し得る。

50

【 0 0 7 5 】

[0090] ブロックのためのイントラ予測モードを選択した後に、イントラ予測ユニット 4 6 は、ブロックのための選択されたイントラ予測モードを示す情報をエントロピー符号化ユニット 5 6 に提供し得る。エントロピー符号化ユニット 5 6 は、選択されたイントラ予測モードを示す情報を符号化し得る。ビデオエンコーダ 2 0 は、送信ビットストリーム中に、複数のイントラ予測モードインデックステーブルおよび複数の変更されたイントラ予測モードインデックステーブル（コードワードマッピングテーブルとも呼ばれる）と、様々なブロックの符号化コンテキストの定義と、コンテキストの各々について使用すべき、最確イントラ予測モード、イントラ予測モードインデックステーブル、および変更されたイントラ予測モードインデックステーブルの指示とを含み得る構成データを含み得る。

10

【 0 0 7 6 】

[0091] ビデオエンコーダ 2 0 は、コーディングされている元のビデオブロックから、モード選択ユニット 4 0 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 5 0 は、この減算演算を行う 1 つまたは複数の構成要素を表す。変換処理ユニット 5 2 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する。変換処理ユニット 5 2 は、DCT と概念的に同様である他の変換を行い得る。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も使用され得る。いずれの場合も、変換処理ユニット 5 2 は、変換を残差ブロックに適用し、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得る。変換処理ユニット 5 2 は、得られた変換係数を量子化ユニット 5 4 に送り得る。量子化ユニット 5 4 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。量子化プロセスは、係数の一部または全部に関連するビット深度を低減し得る。量子化の程度は、量子化パラメータを調整することによって変更され得る。いくつかの例で、量子化ユニット 5 4 は、次いで、量子化変換係数を含む行列の走査を行い得る。代替的に、エントロピー符号化ユニット 5 6 が走査を行い得る。

20

【 0 0 7 7 】

[0092] 量子化の後、エントロピー符号化ユニット 5 6 は、量子化変換係数をエントロピーコーディングする。例えば、エントロピー符号化ユニット 5 6 は、コンテキスト適応型可変長コーディング（CAVLC）、コンテキスト適応型バイナリ算術コーディング（CABAC：context adaptive variable length coding）、シンタックスベースコンテキスト適応型バイナリ算術コーディング（SBAC：syntax-based context-adaptive binary arithmetic coding）、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コーディング技法を行い得る。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、コンテキストは、隣接するブロックに基づき得る。エントロピーコーディングユニット 5 6 によるエントロピーコーディングの後、符号化されたビットストリームは、別のデバイス（例えば、ビデオデコーダ 3 0）に送信されるか、または後で送信するかまたは取り出すためにアーカイブされ得る。

30

【 0 0 7 8 】

[0093] 逆量子化ユニット 5 8 および逆変換ユニット 6 0 は、それぞれ逆量子化および逆変換を適用して、例えば参照ブロックとして後で使用するために、ピクセル領域中で残差ブロックを再構成する。動き補償ユニット 4 4 は、残差ブロックを参照ピクチャメモリ 6 4 のフレームのうちの 1 つの予測ブロックに加算することによって参照ブロックを計算し得る。動き補償ユニット 4 4 はまた、再構成された残差ブロックに 1 つまたは複数の補間フィルタを適用して、動き推定において使用するサブ整数ピクセル値を計算し得る。加算器 6 2 は、再構成された残差ブロックを、動き補償ユニット 4 4 によって生成された動き補償予測ブロックに加算して、参照ピクチャメモリ 6 4 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。再構成されたビデオブロックは、後続のビデオフレーム中のブロックをインターコーディングするために動き推定ユニット 4 2 および動き補償ユニッ

40

50

ト 4 4 によって参照ブロックとして使用され得る。

【 0 0 7 9 】

[0094] 図 2 のビデオエンコーダ 2 0 は、視差動きベクトルを使用して、包含ブロック内のビデオデータの現在のブロックを符号化することと、現在のブロックの隣接ブロックも包含ブロック内にあるとの判断に基づいて、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替することと、候補リスト中の複数のブロックのうちの 1 つから視差動きベクトル予測子を選択することと、視差動きベクトル予測子に基づいて視差動きベクトルを符号化することを行うように構成されたビデオエンコーダの例を表している。

【 0 0 8 0 】

[0095] 図 3 は、ビュー間予測のための視差ベクトルを生成するための技法を実装し得るビデオデコーダ 3 0 の一例を示すブロック図である。図 3 の例で、ビデオデコーダ 3 0 は、エン트로ピー復号ユニット 7 0 と、動き補償ユニット 7 2 と、イントラ予測ユニット 7 4 と、逆量子化ユニット 7 6 と、逆変換ユニット 7 8 と、参照ピクチャメモリ 8 2 と、加算器 8 0 とを含む。ビデオデコーダ 3 0 は、いくつかの例では、ビデオエンコーダ 2 0 (図 2) に関して説明した符号化パスとは概して逆の復号パスを行い得る。動き補償ユニット 7 2 は、エン트로ピー復号ユニット 7 0 から受信された時間的または視差動きベクトルに基づいて予測データを生成し得、イントラ予測ユニット 7 4 は、エン트로ピー復号ユニット 7 0 から受信されたイントラ予測モードインジケータに基づいて予測データを生成し得る。

【 0 0 8 1 】

[0096] 復号プロセス中に、ビデオデコーダ 3 0 は、ビデオエンコーダ 2 0 から、符号化ビデオスライスのビデオブロックと、関連するシンタックス要素とを表す符号化ビデオビットストリームを受信する。ビデオデコーダ 3 0 のエン트로ピー復号ユニット 7 0 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、および他のシンタックス要素を生成するためにビットストリームをエン트로ピー復号する。エン트로ピー復号ユニット 7 0 は、動きベクトル (時間的および / または視差) とインター予測 (時間的および / または視差) に関係する他の予測シンタックス要素とを動き補償ユニット 7 2 に転送する。ビデオデコーダ 3 0 は、ビデオスライスレベルおよび / またはビデオブロックレベルでシンタックス要素を受信し得る。

【 0 0 8 2 】

[0097] ビデオスライスがイントラコード化 (I) スライスとしてコーディングされるとき、イントラ予測ユニット 7 4 は、シグナリングされたイントラ予測モードと、現在フレームまたはピクチャの、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得る。ビデオフレームがインターコード化 (すなわち、時間的 B、時間的 P またはビュー間) スライスとしてコーディングされるとき、動き補償ユニット 7 2 は、エン트로ピー復号ユニット 7 0 から受信された動きベクトルと他のシンタックス要素とに基づいて、現在のビデオスライスのビデオブロック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のうちの 1 つ内の参照ピクチャのうちの 1 つから生成され得る。ビデオデコーダ 3 0 は、参照ピクチャメモリ 8 2 に記憶された参照ピクチャに基づいて、デフォルトの構成技法を使用して、参照フレームリスト、すなわち、リスト 0 およびリスト 1 を構成し得る。

【 0 0 8 3 】

[0098] 動き補償ユニット 7 2 は、動きベクトルと他のシンタックス要素とをパースすることによって現在のビデオスライスのビデオブロックのための予測情報を判断し、その予測情報を使用して、復号されている現在のビデオブロックのための予測ブロックを生成する。例えば、動き補償ユニット 7 2 は、ビデオスライスのビデオブロックをコーディングするために使用される予測モード (例えば、イントラ予測、時間的インター予測、またはビュー間予測) と、インター予測スライスタイプ (例えば、B スライス、P スライス、または G P B スライス) と、スライスの参照ピクチャリストのうちの 1 つまたは複数のた

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ド化ビデオブロックのためのインター予測ステータスと、現在のビデオスライス中のビデオブロックを復号するための他の情報とを判断するために、受信されたシンタックス要素のいくつかを使用する。

【0084】

[0099] 本開示の技法によれば、動き補償ユニット72は、現在のブロックに関する動き情報を再生する方法を示すシンタックス要素を受信し得る。動き補償ユニット72は、現在のブロックに関する動き情報を復号するために使用される動き情報の取り出し元になる空間的隣接ブロックまたは時間的隣接ブロックの識別子を示す1つまたは複数のシンタックス要素を受信し得る。そのようなシンタックス要素は、例えば、マージインデックスまたはAMVPインデックスを含み得る。現在のブロックは、包含ブロック内にあることがあり（例えば、現在のブロックが、MER内のCUもしくはPUまたはCU内のPUを備え得る場合）、シンタックス要素によって示される空間的隣接ブロックは、同じく包含ブロック内にある現在のブロックの隣接ブロックに対応し得る。

【0085】

[0100] そのような場合、本開示の技法によれば、動き補償ユニット72は、（識別子が、シンタックス要素のうちの1つ、例えばマージインデックスまたはAMVPインデックスに対応し得る場合）識別子によって示される現在のブロックの隣接ブロックを、包含ブロック外にある包含ブロックの隣接ブロックで代替できる。例えば、識別子が現在のブロックの隅にある隣接ブロックに対応する場合、動き補償ユニット72は、例えば、図8に関して後述するように、包含ブロックの対応する隅にあるブロックで代替できる。代替として、識別子が現在のブロックの隅にある隣接ブロックに対応する場合、動き補償ユニット72は、例えば、図10に関して後述するように、同様の垂直位置または水平位置にあるが、垂直または水平にシフトされて現在のブロックの外にあるブロックで代替できる。概して、ビデオデコーダ30は、動き情報が適切に復号され得るように、ビデオエンコーダ20によって使用される方法と同じ方法で、現在のブロックのそのような隣接ブロックを、包含ブロックの隣接ブロックで代替するように構成され得る。

【0086】

[0101] より詳細には、動き補償ユニット72は、動き情報の取り出し元になり得る1つまたは複数の候補隣接ブロック（空間的または時間的）を含む候補リストを作ることができる。本開示の技法によれば、現在のブロックの隣接ブロックが、現在のブロックを包含する包含ブロック内にあるとき、ビデオデコーダ30は候補リスト中で、現在のブロックの隣接ブロックを、包含ブロックの隣接ブロックで代替できる。予測動き情報の取り出し元になるブロックを識別する識別子（例えば、マージインデックスまたはAMVPインデックス）は、候補リスト中のエントリに対応し得る。

【0087】

[0102] 包含ブロックは、並列復号領域(parallel decoding region)を定義し得る。包含ブロック外のデータを使用して（例えば、マージモードまたはAMVPモードを使用して）現在のブロックに関する動き情報を予測することによって、現在のブロックに関する動き情報を復号する前に包含ブロック内の情報が復号される必要がなくなる。このようにして、動き補償ユニット72は、包含ブロック内の2つ以上のブロックに関する動き情報を復号できる。従って、包含ブロックは、並列復号領域を定義すると言われ得る。言い換えれば、本開示の技法は、ビデオデコーダ30が、包含ブロック内のデータを並列に復号することを可能にでき、それにより、包含ブロック内の2つ以上のブロックの実質的に同時の復号を可能にし得る。

【0088】

[0103] 動き補償ユニット72は、前に復号されたピクチャから、例えば、参照ピクチャメモリ82からデータを取り出すために復号された動きベクトルを使用し得る。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行い得る。動き補償ユニット72は、ビデオブロックの符号化中にビデオエンコーダ20によって使用された補間フィルタ

を使用して、参照ブロックのサブ整数ピクセルの補間値を計算し得る。この場合、動き補償ユニット72は、受信されたシンタックス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判断し、その補間フィルタを使用して予測ブロックを生成し得る。

【0089】

[0104] 逆量子化ユニット76は、ビットストリーム中で与えられ、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 (inverse quantize)、すなわち、逆量子化 (de-quantize) する。逆量子化プロセスは、量子化の程度を判断し、同様に、適用されるべき逆量子化の程度を判断するための、ビデオスライス中のビデオブロックごとにビデオデコーダ30によって計算される量子化パラメータ QP_y の使用を含み得る。逆変換ユニット78は、ピクセル領域において残差ブロックを生成するために、逆変換、例えば逆DCT、逆整数変換、または概念的に同様の逆変換プロセスを変換係数に適用する。

10

【0090】

[0145] 動き補償ユニット72が、動きベクトルと他のシンタックス要素とに基づいて現在のビデオブロックのための予測ブロックを生成した後に、ビデオデコーダ30は、逆変換ユニット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に加算することによって、復号されたビデオブロックを形成する。加算器80は、この加算演算を行う1つまたは複数の構成要素を表す。所望される場合、ブロックキネスアーティファクトを除去するために、復号ブロックをフィルタ処理するためにデブロッキングフィルタも適用され得る。ピクセル遷移を平滑化するために、または場合によってはビデオ品質を改善するために、他のループフィルタも (コーディンググループ中またはコーディンググループ後のいずれかで) 使用され得る。所与のフレームまたはピクチャ中の復号されたビデオブロックは、次いで、その後の動き補償のために使用される参照ピクチャを記憶する参照ピクチャメモリ82に記憶される。参照ピクチャメモリ82はまた、図1のディスプレイデバイス32などのディスプレイデバイス上での後の表示のための、復号されたビデオを記憶する。

20

【0091】

[0106] このようにして、図3のビデオデコーダ30は、包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断することと、現在のブロックの隣接ブロックも包含ブロック内にあるとの判断に基づいて、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替することと、候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、視差動きベクトル予測子に基づいて視差動きベクトルを復号することとを行うように構成されたビデオデコーダの例を表している。

30

【0092】

[0107] 図4は、例示的なMVC予測パターンを示す概念図である。マルチビュービデオコーディング (MVC) は、ITU-T H.264/AVCの拡張である。同様の技法が、HEVCに適用され得る。図4の例では、(ビューID「S0」~「S7」を有する) 8つのビューが示され、12個の時間口ケーション (「T0」~「T11」) がビューごとに示されている。すなわち、図4中の各行はビューに対応し、各列は時間口ケーションを示す。

40

【0093】

[0108] MVCがH.264/AVCデコーダによって復号可能である、いわゆるベースビューを有し、また、ステレオビューペアがMVCによってサポートされ得るが、MVCの1つの利点は、MVCが、3Dビデオ入力として3つ以上のビューを使用し、複数のビューによって表されるこの3Dビデオを復号する例をサポートできることである。MVCデコーダを有するクライアントのレンダラは、複数のビューを用いて3Dビデオコンテンツを予想し得る。

【0094】

[0109] 典型的なMVC復号順序は、タイムファーストコーディングと呼ばれる。アク

50

セスユニットは、1つの出力時間インスタンスのためのすべてのビューのコーディングされたピクチャを含み得る。例えば、時間T0のピクチャの各々は、共通のアクセスユニットに含まれ得るが、時間T1のピクチャの各々は、第2の共通のアクセスユニットに含まれ得る、など。復号順序は、出力順序または表示順序と必ずしも同一とは限らない。

【0095】

[0110] 図4中のフレームは、対応するフレームがイントラコード化される（すなわち、Iフレームである）のか、または一方向に（すなわち、Pフレームとして）インターコード化されるのか、もしくは複数の方向に（すなわち、Bフレームとして）インターコード化されるのかを指定する、文字を含む影付きブロックを使用して、図4中の各行と各列との交差部において示されている。概して、予測は矢印によって示され、ここで矢印の終点のフレームは、予測参照のために矢印の始点のオブジェクトを使用する。例えば、時間ロケーションT0におけるビューS2のPフレームは、時間ロケーションT0におけるビューS0のIフレームから予測される。

10

【0096】

[0111] シングルビュービデオ符号化の場合と同様に、マルチビュービデオコーディングビデオシーケンスのフレームは、異なる時間ロケーションにおけるフレームに関して予測符号化され得る。例えば、時間ロケーションT1におけるビューS0のbフレームは、時間ロケーションT0におけるビューS0のIフレームからそのbフレームに向けられた矢印を有し、その矢印は、bフレームがIフレームから予測されることを示す。しかしながら、さらに、マルチビュービデオ符号化のコンテキストにおいて、フレームは、ビュー間予測され得る。すなわち、ビューコンポーネントは、参照のために他のビュー中のビューコンポーネントを使用できる。MVCでは、例えば、別のビュー中のビューコンポーネントがインター予測参照であるかのように、ビュー間予測が実現される。潜在的なビュー間参照は、シーケンスパラメータセット（SPS）MVC拡張においてシグナリングされ、インター予測またはビュー間予測参照のフレキシブルな順序付けを可能にする参照ピクチャリスト構成プロセスによって変更され得る。

20

【0097】

[0112] H.264/AVCのMVC拡張で、一例として、ビュー間予測は視差動き補償によってサポートされ、視差動き補償は、H.264/AVC動き補償のシンタックスを使用するが、異なるビュー中のピクチャが参照ピクチャとして使用されることが可能になる。MVCによって2つのビューのコーディングがサポートされ得、それは、一般に、立体視ビューと呼ばれる。MVCの利点のうちの1つは、MVCエンコーダが3Dビデオ入力として3つ以上のビューをとらえることができ、また、MVCデコーダがそのようなマルチビュー表現を復号できることである。従って、MVCデコーダをもつレンダリングデバイスは、3つ以上のビューをもつ3Dビデオコンテンツを予想し得る。

30

【0098】

[0113] MVCでは、同じアクセスユニット中の（すなわち、同じ時間インスタンスを有する）ピクチャ間でビュー間予測（IVP）が可能になる。アクセスユニットは、概して、共通の時間インスタンスのためのすべてのビューコンポーネント（例えば、すべてのNALユニット）を含むデータのユニットである。従って、MVCでは、同じアクセスユニット中のピクチャ間でビュー間予測が許可される。非ベースビューのうちの1つの中のピクチャをコーディングするとき、ピクチャが異なるビュー中にあるが同じ時間インスタンス（例えば、同じPOC値）をもつ場合（従って、同じアクセスユニット中にある）、そのピクチャは参照ピクチャリストに追加され得る。ビュー間予測の参照ピクチャは、任意のインター予測の参照ピクチャと同様に、参照ピクチャリストの任意の位置に置かれ得る。

40

【0099】

[0114] マルチビュービデオコーディングのコンテキストでは、2種類の動きベクトルが存在する。一方は、時間的参照ピクチャを指す動きベクトルであり、対応するインター予測モードは、動き補償予測（MCP：motion compensated prediction）と呼ばれる。

50

他方は、異なるビューにおけるピクチャを指す視差動きベクトルであり、対応するビュー間予測モードは、視差補償予測（DCP：disparity-compensated prediction）と呼ばれる。

【0100】

【0115】 従来のHEVCでは、動きパラメータの予測に関する2つのモードが存在するが、一方はマージモードであり、他方は高度動きベクトル予測（AMVP）である。マージモードでは、動きパラメータ（参照ピクチャおよび動きベクトル）の候補リストが構成され、候補は、空間的隣接ブロックまたは時間的隣接ブロックからであり得る。空間的隣接ブロックおよび時間的隣接ブロックは、候補リスト、すなわち動き予測情報が選択され得る1組の候補を形成できる。従って、動き予測情報として選択される動きパラメータは、候補リストへのインデックスをコーディングすることによってシグナリングされ得る。デコーダ側で、インデックスが復号されると、インデックスが指す対応するブロックのすべての動きパラメータは、マージモードにおいて継承され得る。

10

【0101】

【0116】 AMVP、従って従来のHEVCで、各動き仮定のための動きベクトル予測子の候補リストは、コーディングされた参照インデックスに基づいて導出される。このリストは、同じ参照インデックスと、時間的参照ピクチャにおいてコロケートブロックの隣接ブロックの動きパラメータに基づいて導出される時間的動きベクトル予測子とに関連する隣接ブロックの動きベクトルを含む。選択された動きベクトルは、候補リストへのインデックスを送信することによってシグナリングされる。加えて、参照インデックス値および動きベクトル差分もシグナリングされる。

20

【0102】

【0117】 さらに、2つの動きデータ圧縮方法がHEVCにおいて定義されている。これらの方法については、以下で図6に関してより詳細に説明する。

【0103】

【0118】 図4は、ビュー間予測の様々な例を提供する。図4の例で、ビューS1のフレームは、ビューS1の異なる時間ロケーションにおけるフレームから予測されるものとして、並びに同じ時間ロケーションにおけるビューS0およびS2のフレームのうちのフレームからビュー間予測されるものとして示されている。例えば、時間ロケーションT1におけるビューS1のbフレームは、時間ロケーションT0およびT2におけるビューS1のBフレームの各々、並びに時間ロケーションT1におけるビューS0およびS2のbフレームから予測される。

30

【0104】

【0119】 図4の例で、大文字の「B」および小文字の「b」は、異なる符号化方法ではなく、フレーム間の異なる階層関係を示すものとする。概して、大文字の「B」フレームは、小文字の「b」フレームよりも予測階層が比較的高い。図4はまた、異なるレベルの陰影を使用して予測階層の変形形態を示し、より大きい量の陰影の（すなわち、比較的より暗い）フレームは、より少ない陰影を有する（すなわち、比較的より明るい）それらのフレームよりも予測階層が高い。例えば、図4中のすべてのIフレームは、完全陰影を用いて示されるが、Pフレームは、いくぶんより明るい陰影を有し、Bフレーム（および小文字のbフレーム）は、互いに様々なレベルの陰影を有するが、PフレームおよびIフレームの陰影よりも常に明るい。

40

【0105】

【0120】 概して、階層が比較的高いそれらのフレームが、階層が比較的低いフレームの復号中に参照フレームとして使用され得るように、予測階層が比較的高いフレームは、階層が比較的低いフレームを復号する前に復号されるべきであるという点で、予測階層はビュー順序インデックスに関係する。ビュー順序インデックスは、アクセスユニット中のビューコンポーネントの復号順序を示すインデックスである。H.264/AVC（MVC追補）の付属書類Hにおいて規定されているように、ビュー順序インデックスはSPSMVC拡張において暗示されている。SPSでは、各インデックス*i*について、対応する

50

$view_id$ がシグナリングされる。いくつかの例で、ビューコンポーネントの復号は、ビュー順序インデックスの昇順に従う。すべてのビューが提示された場合、ビュー順序インデックスは、0 から $num_views_minus_1$ までの連続する順序になっている。

【0106】

[0121] このようにして、参照フレームとして使用されるフレームは、その参照フレームを参照して符号化されたフレームを復号する前に復号され得る。ビュー順序インデックスは、アクセスユニット中のビューコンポーネントの復号順序を示すインデックスである。各ビュー順序インデックス i について、対応する $view_id$ がシグナリングされる。ビューコンポーネントの復号は、ビュー順序インデックスの昇順に従う。すべてのビューが提示された場合、ビュー順序インデックスのセットは、0 からビューの全数よりも 1 少ない数までの連続的な順序付きセットを備え得る。

10

【0107】

[0122] 階層の等しいレベルにおけるいくつかのフレームの場合、復号順序は、互いに重要でないことがある。例えば、時間ロケーション T_0 におけるビュー S_0 の I フレームは、時間ロケーション T_0 におけるビュー S_2 の P フレームのための参照フレームとして使用され、その P フレームは今度は、時間ロケーション T_0 におけるビュー S_4 の P フレームのための参照フレームとして使用される。従って、時間ロケーション T_0 におけるビュー S_0 の I フレームは、時間ロケーション T_0 におけるビュー S_2 の P フレームの前に復号されるべきであり、その P フレームは、時間ロケーション T_0 におけるビュー S_4 の P フレームの前に復号されるべきである。しかしながら、ビュー S_1 および S_3 は、予測のために互いに依拠しないが、代わりに、予測階層がより高いビューからのみ予測されるので、ビュー S_1 と S_3 との間で復号順序は重要でない。その上、ビュー S_1 がビュー S_0 および S_2 の後に復号される限り、ビュー S_1 はビュー S_4 の前に復号され得る。

20

【0108】

[0123] このようにして、ビュー $S_0 \sim S_7$ を記述するために階層順序が使用され得る。表記法 $SA > SB$ は、ビュー SA がビュー SB の前に復号されるべきであることを意味する。この表記法を使用すると、図 4 の例で、 $S_0 > S_2 > S_4 > S_6 > S_7$ である。また、図 4 の例に関して、 $S_0 > S_1$ 、 $S_2 > S_1$ 、 $S_2 > S_3$ 、 $S_4 > S_3$ 、 $S_4 > S_5$ 、および $S_6 > S_5$ である。これらの要件に違反しないビューのための任意の復号順序が可能である。従って、いくつかの制限のみをもつ、多くの異なる復号順序が可能である。

30

【0109】

[0124] 本開示の技法によれば、ビデオエンコーダ 20 およびビデオデコーダ 30 は、ブロック（例えば、包含ブロック）内のデータを並列にコーディングするように構成され得る。これを達成するために、ビデオエンコーダ 20 およびビデオデコーダ 30 は、包含ブロック内のデータを、包含ブロック内の他のデータを予測するために使用しないように構成され得る。例えば、ビデオエンコーダ 20 およびビデオデコーダ 30 は、包含ブロック内の現在のブロックの動き情報を、現在のブロックの外にあるブロックからの予測動き情報を使用してコーディングするように構成され得る。従って、ビデオエンコーダ 20 およびビデオデコーダ 30 は、例えば時間的動きベクトルおよび / または視差動きベクトルをコーディングするための動き予測候補で代替できる。このようにして、ビデオエンコーダ 20 およびビデオデコーダ 30 は、本開示の技法を使用して、例えば図 4 の時間的動きベクトルおよび / または視差動きベクトルを符号化できる。

40

【0110】

[0125] 図 5 は、現在の PU 並びに空間的に隣接するブロック A_0 、 A_1 、 B_0 、 B_1 、および B_2 の例を示すブロック図である。図 5 は、現在の PU とその空間的に隣接する PU との関係を示している。この説明のために、以下のシンボルが定義される。

【0111】

[0126] 現在のピクチャの左上のサンプルに関連する現在の PU の左上のルーマサンプルを明示するために、ルーマロケーション (x_P, y_P) （すなわち、ルミナンス成分）

50

が使用され得る。

【0112】

【0127】 変数 $nPSW$ および $nPSH$ は、ルーマに関する、すなわちルミナンス成分における現在の PU の幅および高さを示し得る。

【0113】

【0128】 この例で、現在のピクチャの左上のサンプルに関連する現在の予測ユニット N の左上のルーマサンプルは (x_N, y_N) である。その場合、 (x_N, y_N) (N は、 A_0 、 A_1 、 B_0 、 B_1 または B_2 によって置き換えられる) は、それぞれ (x_{P-1}, y_{P+nPSH}) 、 $(x_{P-1}, y_{P+nPSH-1})$ 、 (x_{P+nPSW}, y_{P-1}) 、 $(x_{P+nPSW-1}, y_{P-1})$ または (x_{P-1}, y_{P-1}) と定義され得る。

10

【0114】

【0129】 上述のように、いくつかの例で、ビデオエンコーダ 20 またはビデオデコーダ 30 のようなビデオコードは、時間的動きベクトル予測を行なえる。すなわち、現在の PU のための動きベクトルは、時間的動きベクトル予測子 ($TMVP$) に関連してコーディングされ得る。まず、 $TMVP$ を得るために、ビデオコードは、コロケートピクチャ (例えば、現在のピクチャと同じアクセスユニット内のピクチャ) を識別し得る。現在のピクチャのスライスが B スライスである場合、`collocated_from_l0_flag` が、コロケートピクチャが $RefPicList0$ からのものかまたは $RefPicList1$ からのものかを示すために、スライスヘッダにおいてシグナリングされ得る。

【0115】

20

【0130】 さらに、 B スライスの場合、最終参照ピクチャリスト ($RefPicList0$ および $RefPicList1$) が構成された後、複合リスト ($RefPicListC$) が構成され得る。複合リストに関して参照ピクチャリスト変更シンタックスが存在する場合、複合リストはさらに変更され得る。

【0116】

【0131】 参照ピクチャリストが識別された後、スライスヘッダでシグナリングされる `collocated_ref_idx` が、リスト中のピクチャの中のピクチャを識別するために使用され得る。コロケート PU が次いで、コロケートピクチャを確認することによって識別され得る。この PU を包含する CU の右下の PU の動き、またはこの PU を包含する CU の中心 PU 内の右下の PU の動きのいずれかが、使用され得る。

30

【0117】

【0132】 $AMVP$ またはマージモードの動き候補を生成するために、上記のプロセスによって識別された動きベクトルが使用されるとき、動きベクトルは、(ピクチャのピクチャ順序カウンタ (POC : picture order count) 値によって反映される) 時間ロケーションに基づいてスケールされ得る。 $HEVC$ で、ピクチャパラメータセット (PPS : picture parameter set) は、フラグ `enable_temporal_mv_flag` を含む。0 に等しい `temporal_id` をもつ特定のピクチャが、0 に等しい `enable_temporal_mv_flag` を有する PPS を参照するとき、 DPB 中のすべての参照ピクチャは、「時間的動きベクトル予測のために使用されない」としてマークされ得、復号順序でその特定のピクチャの前のピクチャからの動きベクトルは、特定のピクチャの復号または復号順序でその特定のピクチャの後のピクチャの復号において時間的動きベクトル予測子として使用されないことになる。

40

【0118】

【0133】 図 6 は、様々なピクチャグループ (GOP) のピクチャを示す、ピクチャが様々なタイプを有する概念図である。具体的には、図 6 は、前の $GOP100$ と、現在の $GOP102$ と、次の $GOP104$ とを示している。前の $GOP100$ は、ピクチャ 106、108 を含む。現在の $GOP102$ は、ピクチャ 110 ~ 124 (ピクチャ 124 は、図 6 の例でクリーンランダムアクセスピクチャ (CRA : clean random access picture) の例を表している) を含む。次の $GOP104$ は、ピクチャ 126、128 を含む。

【0119】

50

[0134] ピクチャ 106 ~ 128 の各々は、対応するピクチャの POC 値を表す数値を含む。復号順序は、POC 値によって示される表示順序とは異なり得ることに留意されたい。図 6 中の矢印は、潜在的予測を表している。具体的には、矢印が指すピクチャは、矢印の起点となっているピクチャによって参照ピクチャとして使用され得る。例えば、ピクチャ 108 (POC 値 16 を有する) は、ピクチャ 116 (POC 値 20 を有する) を予測するための参照ピクチャとして使用され得る。

【0120】

[0135] HEVC では、ネットワークアブストラクションレイヤ (NAL: network abstraction layer) ユニットタイプによって識別され得る 4 つのピクチャタイプが存在する。これらは、瞬時復号リフレッシュ (IDR: instantaneous decoding refresh) ピクチャ、クリーンランダムアクセス (CRA: clean random access) ピクチャ、時間レイヤアクセス (TLA: temporal layer access) ピクチャ、および IDR ピクチャ、CRA ピクチャまたは TLA ピクチャでないコード化ピクチャである。この例で、POC 値 24 をもつピクチャは CRA ピクチャである。

10

【0121】

[0136] IDR ピクチャおよびコード化ピクチャは、H.264 / AVC 規格から実質的に継承されたピクチャタイプである。CRA および TLA のピクチャタイプは HEVC において新しく、H.264 / AVC 規格において利用できない。CRA ピクチャは、IDR ピクチャを挿入するよりも効率的である、ビデオシーケンスの中央の任意のランダムアクセスポイントから始まる復号を可能にするピクチャタイプである。HEVC では、これらの CRA ピクチャから始まるビットストリームもまた、ビットストリームに適合している。TLA ピクチャは、有効な時間レイヤスイッチングポイントを示すために使用され得るピクチャタイプである。

20

【0122】

[0137] ブロードキャストおよびストリーミング (例えば、動的適応ストリーミングオーバー HTTP (DASH: Dynamic Adaptive Streaming over HTTP)、リアルタイムトランスポートプロトコル (RTP: Real-time Transport Protocol) またはリアルタイムストリーミングプロトコル (RTSP: Real-Time Streaming Protocol) に従うなどしてネットワークを介する) などのビデオ適用では、ユーザが最小限の遅延で異なるチャネル間で切り替えて、ビデオの特定の部分にジャンプするための重要な機能が必要とされる。この機能は、ビデオビットストリームで一定の間隔をおいてランダムアクセスピクチャを有することによって可能にされ得る。H.264 / AVC と HEVC の両方の中で指定される IDR ピクチャは、ランダムアクセスに使用され得る。しかしながら、IDR ピクチャは、コード化ビデオシーケンスを開始させ、復号ピクチャバッファ (DPB) を常に削除するので、復号順序で IDR に続くピクチャは、IDR ピクチャの前に復号されたピクチャを参照として使用できない。その結果、ランダムアクセスに対する IDR ピクチャに依存するビットストリームは、著しく低いコーディング効率 (例えば、6%) を有することがある。コーディング効率を改善するために、復号順序で CRA ピクチャに後続するが出力順序で CRA ピクチャに先行するピクチャが、CRA ピクチャより前に復号されたピクチャを参照ピクチャとして使用することを、HEVC における CRA ピクチャが可能にする。

30

40

【0123】

[0138] CRA ピクチャ (ピクチャ 124) の前後の例示的な予測構造が図 6 に示されている。CRA ピクチャ 124 (POC 24 を有する) は現在の GOP 102 に属し、現在の GOP 102 は、復号順序で CRA ピクチャ 124 に後続するが出力順序で CRA ピクチャ 124 に先行する他のピクチャ 110 ~ 122 (POC 値 17 ~ 23 を有する) を包含する。すなわち、図 6 の例におけるピクチャ 110 ~ 122 (POC 17 ~ 23 を有する) は、ピクチャ 124 に、またはやはりピクチャ 124 に依存するピクチャに依存し得るので、ピクチャ 124 はピクチャ 110 ~ 122 の前に復号されなければならない。従って、ピクチャ 110 ~ 122 は、CRA ピクチャ 124 の先行ピクチャと呼ばれ、現

50

在のC R Aピクチャの前のI D RピクチャまたはC R Aピクチャから復号が開始された場合に適切に復号され得る。しかしながら、先行ピクチャは、C R Aピクチャ1 2 4からのランダムアクセスが生じたときには適切に復号され得ない。すなわち、図6の例に示すように、先行ピクチャは、ピクチャ1 0 8などのビットストリームにおけるより早い段階のピクチャに依存し得る。従って、先行ピクチャ1 1 0 ~ 1 2 2は通常、ランダムアクセス復号中に処分され得る。

【0 1 2 4】

[0139] どこで復号が開始されるかに応じて利用不可能なことがある参照ピクチャからのエラー伝搬を防ぐために、復号順序と出力順序の両方でC R Aピクチャ1 2 4に後続する、図6に示す次のG O P 1 0 4におけるすべてのピクチャは、復号順序または出力順序のいずれかでC R Aピクチャ1 2 4に先行するピクチャ（P O C値1 7 ~ 2 3を有する先行ピクチャ1 1 0 ~ 1 2 2を含む）を参照に一切使用しないようにコーディングされ得る。

10

【0 1 2 5】

[0140] 同様のランダムアクセス機能が、リカバリポイントS E Iメッセージを有するH . 2 6 4 / A V Cにおいてサポートされる。H . 2 6 4 / A V Cデコーダ実装がその機能をサポートすることまたはサポートしないこともある。H E V Cにおいて、C R Aピクチャによって開始するビットストリームは、適合ビットストリームと見なされる。ビットストリームがC R Aピクチャによって開始するとき、C R Aピクチャの先行ピクチャは利用不可能参照ピクチャを指し、従って正しく復号され得ない。しかしながら、H E V Cは、開始するC R Aピクチャの先行ピクチャは出力されず、従って「クリーンランダムアクセス」という名称であることを明記している。ビットストリーム適合性要件を確立するために、H E V Cは、非出力先行ピクチャを復号するために利用不可能参照ピクチャを生成するための復号プロセスを指定する。しかしながら、復号プロセスがビットストリームの開始から行われるときと比較して、適合デコーダ実装が同等の出力を生成し得る限り、適合デコーダ実装は、その復号プロセスに従う必要はない。

20

【0 1 2 6】

[0141] H E V Cにおいて、適合ビットストリームは、I D Rピクチャをまったく含まず、従ってコード化ビデオシーケンスのサブセットまたは不完全なコード化ビデオシーケンスを含み得ることは注目に値する。

30

【0 1 2 7】

[0142] 図7は、一連のブロックにおける動きデータ圧縮を示す概念図である。具体的には、ブロックは、1 6 × 1 6ブロック境界1 5 0 A ~ 1 5 0 Eによって分割される。図7の例において、影付きブロックに関する動き情報がラインバッファに記憶され得る。影なしブロックは、隣接ブロックの動き情報を共有できる。従って、ブロックペア1 5 2 A ~ 1 5 2 Iの各々は動き情報を共有できる。H E V Cにおける動きベクトル記憶圧縮方法は、マージ/スキップモードおよびA M V Pのための時間的動きベクトルの記憶およびローディングに費やされるメモリ帯域幅を減らすように設計される。1 6 × 圧縮方法は基本的に、水平方向と垂直方向の両方において4倍で動きベクトルフィールドをダウンサンプリングし、1 6 × 1 6領域ごとに同じ動きベクトルが生じる。

40

【0 1 2 8】

[0143] 動きデータは一般に、インター予測方向（例えば、より早い表示時間（L i s t 0）を有するピクチャに、より遅い表示時間（L i s t 1）を有するピクチャに、または双方向）と、参照ピクチャインデックス（例えば、参照ピクチャリストへのインデックス）と、水平成分と垂直成分とを含み得る動きベクトル（M V）とを含む。上位L C U行の動きデータにアクセスする必要があるP Uごとに、ラインバッファにおける圧縮動きデータのみがアクセスされ得る。図7は、1つのラインの2 : 1の動きデータ圧縮を示している。4つの4 × 4ブロックごとに、最初と最後の4 × 4ブロックの動きデータがラインバッファに記憶される。最初のブロックの動きデータは、最初の2つのブロックの動きデータを表す。最後のブロックの動きデータは、最後の2つのブロックの動きデータを表す。

50

。このようにして、ちょうど半分だけのメモリが必要とされる。

【0129】

[0144] HEVCで、LCUは、並列動き推定領域(MER)に分割されてよく、現在のPUとは異なるMERに属する隣接PUのみがマージ/スキップMVPリスト構成プロセスに含まれることを許容し得る。MERのサイズは、 $\log_2 \text{parallel_merge_level_minus } 2$ としてピクチャパラメータセット(PPS)でシグナリングされ得る。 $2N \times 2N$ が最小CUサイズであって、MERサイズが $N \times N$ よりも大きいとき、空間的隣接ブロックが、それが現在のPUと同じMERの中にある場合に利用不可能と見なされる方式で、MERは実施され得る。

【0130】

[0145] MPEGにおけるHEVCベースの3Dビデオコーディング(3D-HEVC)コーデックは、m22570(Heiko Schwarz、Christian Bartsch、Sebastian Bosse、Heribert Brust、Tobias Hinz、Haricharan Lakshman、Detlev Marpe、Philipp Merkle、Karsten Muller、Hunn Rhee、Gerhard Tech、Martin Winken、Thomas Wiegand、「Description of 3D Video Technology Proposal by Fraunhofer HHI(HEVC compatible; configuration A)」、ISO/IEC JTC1/SC29/WG11 MPEG2011/m22570、ジュネーブ、スイス、2011年11月)およびm22571で提案された解決策に基づいている。3D-HEVCの参照ソフトウェアの記述は、Heiko Schwarz、Krzysztof Wegner、「Test Model under Consideration for HEVC based 3D video coding」、ISO/IEC JTC1/SC29/WG11 MPEG2011/N12559、サンノゼ、米国、2012年2月として入手可能である。参照ソフトウェアは、HTMと呼ばれ、https://hevc.hhi.fraunhofer.de/svn_3DVCSsoftware/trunkで入手可能である。

【0131】

[0146] Chenらによる「Disparity Vector Construction Method for 3D-HEVC」、代理人整理番号第121833U1/1212:178USP1号および代理人整理番号第121833U2/1212:178USP2号、2012年3月14日に出願された米国仮出願第61/610,961号および2012年4月11日に出願された米国仮出願第61/623,041号では、空間的視差ベクトル(SDV)、時間的視差ベクトル(TDV)または暗黙的視差ベクトル(IDV:Implicit Disparity Vector)による視差ベクトル生成(構成)方法がビュー間動き予測について説明されている。その中で定義されているように、SDVは、ビュー間参照を指している現在のPUの空間的隣接PUの動きベクトルである。TDVは、同じアクセスユニットによる任意の参照ピクチャまたはビュー間ピクチャにおけるコロケートPUまたはコロケートLCUからの視差動きベクトルであり得る。代替として、TMVPに使用されたピクチャからのコロケートPUの動きベクトルまたはTMVPによって生成された動きベクトルが視差ベクトルである場合、それはTDVとして扱われてもよい。現在のPUの空間的隣接PUまたは時間的隣接PUがビュー間動き予測を使用した場合、隣接PUに使用された視差ベクトルはIDVであり得る。上記3種類による視差ベクトルは、視差ベクトル候補リストを構成するために使用されてよく、それらのうちの1つは、いくつかの基準に基づいて選択され、IMPに直接使用され得る。

【0132】

[0147] ビュー間動き予測は、AMVPモードとMERGEモードの両方に適用され得る。PUごとに、ビュー間動き予測用視差ベクトル(DVIVMP:Disparity Vector for Inter-View Motion Prediction)が、構成される視差ベクトルであり、ビュー間動き

10

20

30

40

50

予測に使用され得る。ビュー間動き予測を含めるために、AMVPモードは、ビュー間動きベクトル予測子が候補リストに追加され得るように拡張されている。現在のブロックの中央サンプルの深度推定に基づいて、視差ベクトルおよび参照ビューにおける参照ブロックが上述のように決定され得る。現在のブロックの参照インデックスがビュー間参照ピクチャを指す場合、ビュー間動きベクトル予測子は、対応する視差ベクトルに等しく設定される。現在の参照インデックスが時間的参照ピクチャを指し、参照ブロックが、現在の参照インデックスと同じアクセスユニットを指す動き仮定を使用する場合には、この動き仮定に関連付けられる動きベクトルは、ビュー間動きベクトル予測子として使用され得る。他の場合には、ビュー間動きベクトル予測子は、無効とマークされ、動きベクトル予測子候補のリストに含まれないことがある。

10

【0133】

[0148] AMVPモードと同様に、動きパラメータの候補リストは、ビュー間動き予測を使用して取得され得る動きパラメータセットによって拡張され得る。潜在的動き仮定ごとに、参照ピクチャリストの最初の2つの参照インデックスが所与の順序で調査され得る。AMVPモードの場合と同様に、参照インデックス0の動きベクトル候補が導出され得る。導出された動きベクトルが有効である場合、参照インデックス0および導出された動きベクトルは、当該仮定に使用され得る。そうでない場合、参照インデックス1が同様にテストされ得る。それがまた無効な動きベクトルをもたらす場合、動き仮定は利用不可能とマークされる。時間的予測を愛好するために、第1のインデックスがビュー間参照ピクチャを指す場合、参照インデックスがテストされる順序が逆転し得る。すべての潜在的動き仮定が利用不可能とマークされた場合、ビュー間候補は選択され得ない。

20

【0134】

[0149] ビュー間残差予測(IRP)が3D-HEVCで有効にされる。現在のピクチャの対応する深度マップ(それは推定深度マップであり得る)に基づいて、現在のブロックについて視差ベクトルが決定され得る。次いで視差ベクトルは、参照ビューにおける残差ブロック(残差参照ブロック)の位置を特定するために使用され得る。残差予測がブロックに対して有効であるとき、残差参照ブロックが、動き補償予測子およびこのブロックに対するシグナリングされた残差の他に加えられ得る。

【0135】

[0150] 代理人整理番号第122144P1/1212:203USP1号を有するChenらによる「Motion Parameter Prediction and Coding for Multiview Video」、2012年4月9日に出席された米国仮出願第61/621,929号では、空間的隣接ブロックおよび時間的隣接ブロックによる視差ベクトル生成方法が、IMPとIRPの両方について説明されている。それらの解決策のうちの1つは次のように定義されている。

30

【0136】

[0151] 候補空間的隣接ブロックは、AMVP中に確認されたものと同じであってよく、それらの順序は、図5に示すように、 A_0 、 A_1 、 B_0 、 B_1 、または B_2 である。順序付けられた空間的隣接ブロックのうちの1つの動きベクトルが視差動きベクトルである場合、それは視差ベクトルを生成するために使用され得る。時間的隣接ブロックの動きベクトルからの時間的視差ベクトル(TDV)が、参照ピクチャ(指定された候補ピクチャ)のセットに基づいて導出されてよく、候補ピクチャごとに、異なるブロックからなるセットが確認され得る。

40

【0137】

[0152] 参照インデックスの昇順に候補ピクチャが確認され得る。1つの参照インデックスがRefPicList0およびRefPicList1における2つのピクチャに対応するとき、RefPicListXからのピクチャは他方のピクチャに先行し、ここでXは、collocated_from_l0_flagに等しい。候補ピクチャごとに3つの候補領域が決定され得る。1つの領域が複数のPUまたは 16×16 領域をカバーするとき、そのような候補領域のPUまたは各 16×16 領域の左上 4×4 ブロックは

50

ラスト走査順序で走査され得る。他の場合、左上 4×4 ブロックが視差ベクトル導出のために選択され得る。CPUは、現在のPUのコロケート領域に対応することがあり、CLCUは、現在のPUのコロケート領域をカバーする最大コーディングユニット(LCU)に対応することがあり、かつBRは、右下 4×4 ブロックに対応することがある。視差ベクトル生成プロセスは、それぞれ、IMPまたはIRPに視差ベクトルが使用されるとき、PUレベルまたはCUレベルで行われ得る。

【0138】

[0153] 最も効率的なコーディングツールのうちの1つは、ビュー間動き予測(IMP)であり、この場合、従属ビューにおけるブロックの動きパラメータが、同じアクセスユニットの他のビューにおけるすでにコーディングされた動きパラメータに基づいて予測または推測される。

10

【0139】

[0154] 図8は、PUを含む動き推定範囲(MER)の外での代替の例を示す概念図である。図8の例では、PU162、PU164、およびPU166がMER160内にある。すなわち、MER160は、PU162と、PU164と、PU166とを包含する包含ブロックの例を表している。本開示の技法によれば、ビデオエンコーダ20および/またはビデオデコーダ30などのビデオコーダが、空間的視差ベクトル確認を行うように構成され得る。いくつかの例で、ビデオコーダは、PUをコーディングするときにMERを使用し得る。このように、ビデオコーダは、空間的視差ベクトルを得るためにMER内のブロックを使用できないことがある。この場合、代替ブロックが使用され得る。

20

【0140】

[0155] PU162、PU164、PU166は、図5のブロックA0、A1、B0、B1、およびB2に対応する隣接ブロックを有する。例えば、PU162は、隣接ブロック172A、176B、176C、176D、および176Eに隣接する。別の例として、PU164は、隣接ブロック174A~174Eに隣接する。さらに別の例として、PU166は、隣接ブロック178A~178Eに隣接する。さらに、MER160は、それ自体の隣接ブロック、ブロック172A~172Eに隣接する。全般に、終わりに文字「A」を有する隣接ブロックは、図5のブロックB2に対応し、終わりに文字「B」を有する隣接ブロックは、図5のブロックB1に対応し、終わりに文字「C」を有する隣接ブロックは、図5のブロックB0に対応し、終わりに文字「D」を有する隣接ブロックは、図5のブロックA1に対応し、終わりに文字「E」を有する隣接ブロックは、図5のブロックA0に対応する。

30

【0141】

[0156] 図8に示すように、PUのA0またはA1の隣接(左下)ブロックがMER160内にある(ブロック174Dおよび174Eなど)とき、それらのブロックは利用不可能と見なされ得る。従って、MER160のA0またはA1の隣接ブロック(ブロック172Dおよび172E)が代替として使用され得る。図8でPU164に関して示すように、PU164のA0およびA1ブロックはブロック174Dおよび174Eであり、この例でPU164をも含むMER160内にあるので利用不可能と見なされる。従って、ビデオエンコーダ20およびビデオデコーダ30は、ブロック174Dをブロック172Dで、およびブロック174Eをブロック172Eで代替できる。

40

【0142】

[0157] このようにして、ブロック174Dおよび174Eは、現在のブロック(この例ではPU164)の左下隣接ブロックの例を表している。その上、この例では、ビデオエンコーダ20およびビデオデコーダ30は、それぞれブロック174Dおよびブロック174Eを、ブロック172Dおよびブロック172Eで代替できる。この代替は、現在のブロックの左下隣接ブロック(例えば、PU164のブロック174D、174E)を、MER160の左隣接ブロック(例えば、ブロック172D、172E)で代替する例を表している。

50

【 0 1 4 3 】

[0158] P UのB 0またはB 1の隣接（右上）ブロックがM E R 1 6 0内にあるとき、それらのブロックは利用不可能と見なされ得る。例えば、P U 1 6 6に隣接する隣接ブロック1 7 8 B、1 7 8 C、およびP U 1 6 4に隣接する隣接ブロック1 7 4 B、1 7 4 Cは、利用不可能と見なされ得る。従って、M E R 1 6 0のB 0またはB 1の隣接ブロック（この例ではブロック1 7 2 B、1 7 2 C）が代替として使用され得る。図8でP U 1 6 6に関して示すように、ブロック1 7 8 B、1 7 8 Cは、この例ではP U 1 6 6をも含むM E R 1 6 0内にあるので利用不可能である。従って、ビデオエンコーダ2 9およびビデオデコーダ3 0は、隣接ブロック1 7 8 B、1 7 8 Cを、隣接ブロック1 7 2 B、1 7 2 C（M E R 1 6 0の隣接ブロック）で代替できる。同様に、ビデオエンコーダ2 0およびビデオデコーダ3 0は、P U 1 6 4の隣接ブロック1 7 4 B、1 7 4 Cを、隣接ブロック1 7 2 B、1 7 2 C（M E R 1 6 0の隣接ブロック）で代替できる。

10

【 0 1 4 4 】

[0159] この代替は、現在のブロック（例えば、P U 1 6 4、1 6 6のうちの1つ）をも包含する包含ブロックにおける隣接ブロックが、現在のブロックの右上隣接ブロック（例えば、ブロック1 7 4 B、1 7 4 C、1 7 8 B、1 7 8 Cのうちの1つ）に対応するときに、ビデオエンコーダ2 0およびビデオデコーダ3 0が、現在のブロック（例えば、P U 1 6 4、1 6 6のうちの1つ）の右上隣接ブロック（例えば、ブロック1 7 4 B、1 7 4 C、1 7 8 B、1 7 8 Cのうちの1つ）を、M E R 1 6 0の右上隣接ブロック（例えば、ブロック1 7 2 B、1 7 2 Cのうちの1つ）で代替できる例を表している。図8で隣接ブロック間に引かれている矢印は、代替を表している。

20

【 0 1 4 5 】

[0160] P UのB 2隣接（左上）ブロックが、P Uをも含むM E R 1 6 0内にある（例えば、P U 1 6 4の隣接ブロック1 7 4 A）とき、隣接ブロックは利用不可能であると見なされ得る。従って、ビデオエンコーダ2 0およびビデオデコーダ3 0は、P U 1 6 4の隣接ブロック1 7 4 Aの代替として、M E R 1 6 0のB 2隣接ブロック（例えば、隣接ブロック1 7 2 A）で代替できる。この代替は、隣接ブロックが現在のブロックの左上隣接ブロックに対応するときに、ビデオエンコーダ2 0および/またはビデオデコーダ3 0が、現在のブロックの左上隣接ブロックを、M E Rの左上隣接ブロックで代替できる例を表している。

30

【 0 1 4 6 】

[0161] P Uのすべての隣接ブロックが、P Uを含むM E Rの外にあるとき、代替は必要とされない。これは、図8の例におけるP U 1 6 2の隣接ブロック1 7 2 A、1 7 6 B ~ 1 7 6 Eに当てはまる。

【 0 1 4 7 】

[0162] 図9は、視差ベクトルが確認されるP U 2 1 0の一例を示す概念図である。図9は、矢印を使用して相対隣接位置を示し、各位置定義の下に数字を示している。P U 2 1 0は、隣接空間的ブロック2 1 2 ~ 2 2 0と隣接時間的ブロック2 2 2とを有する。上述したChenらによる「Motion Parameter Prediction and Coding for Multiview Video」で説明されているように、視差ベクトル生成を行うビデオコーダが、図9の空間ネイバー視差ベクトルに加えて他のタイプの視差ベクトル、例えば時間的視差ベクトルを確認し得ることに留意されたい。図9の例は、図らずもA M V PまたはM E R G Eで確認されるものと同じである、隣接ブロック2 1 2 ~ 2 2 0から空間的視差ベクトルが確認される場合について説明している。そのような場合、M E Rが使用されるとき、図8に関して上述したように代替がもたらされ得る。他のプロセスは変わらないままでよい。

40

【 0 1 4 8 】

[0163] 代替として、空間的隣接ブロックがM E R内にあるとき、M E R内のブロックを確認する代わりに、M E Rの境界にある水平または垂直に位置合わせされたブロックが確認され得る。（それぞれブロックA 0、A 1に対応する）ブロック2 2 0、2 1 8の場

50

合、MERの左境界にある水平に位置合わせされたブロックが代替として使用され得る。(それぞれブロックB0、B1に対応する)ブロック216、214の場合、MERの上境界にある垂直に位置合わせされたブロックが代替として使用され得る。(ブロックB2に対応する)ブロック212の場合、MERの左境界にある水平に位置合わせされたブロックとMERの上境界にある垂直に位置合わせされたブロックの両方が代替として使用され得る。

【0149】

[0164] 図10は、PUを含むMERの外での代替の代替例を示す概念図である。図10の例は、PU232、236を含むMER230を示している。PU232は、空間的隣接ブロック234A~234Eを有する。PU236は、空間的隣接ブロック238A~238Eを有する。図10に示すように、PUのA0またはA1の隣接ブロックが、PUを含むMER内にあるとき、それらは利用不可能と見なされ得る。従って、水平方向におけるMERの外の位置合わせされたブロック(すなわち、垂直に位置合わせされたブロック)が代替として使用され得る。図10でPU232に関して示すように、PU232の(ブロックA0およびA1に対応する)ブロック234Dおよび234Eは、PU232をも含むMER230内にあるので利用不可能である。従って、ビデオエンコーダ20およびビデオデコーダ30は、それぞれブロック234D、234Eをブロック244D、244Eで代替できる。この例では、ブロック244D、244Eは、ブロック234D、234Eと同じ垂直位置を有するが、MER230の外にあり、MER230に隣接する。

10

20

【0150】

[0165] このようにして、ブロック234Dおよび234Eをブロック244Dおよび244Eで代替することは、現在のブロック(例えば、PU232)の隣接ブロックが現在のブロックの左下隣接ブロック(ブロック234D、234E)に対応するときに、ビデオエンコーダ20およびビデオデコーダ30が、現在のブロックの左下隣接ブロック(例えば、ブロック234D、234Eのうちの1つ)を、MER230の左隣接ブロック(例えば、ブロック244D、244Eのうちの対応する1つ)で代替でき、結果的にMERの左隣接ブロックが、現在のブロックの左下隣接ブロックと同じ垂直位置を有するようになる例を表している。具体的には、この例で、ブロック244Dはブロック234Dと同じ垂直位置を有し、ブロック244Eはブロック234Eと同じ垂直位置を有する。

30

【0151】

[0166] PUのB0またはB1の隣接(右上)ブロックが、PUを含むMER内にあるとき、それらは利用不可能と見なされ得る。従って、垂直方向におけるMERの外の位置合わせされたブロック(すなわち、水平に位置合わせされたブロック)が代替として使用され得る。図10でPU236に関して示すように、PU236の隣接ブロック238B、238Cは、PU236を含むMER230内にあるので利用不可能である。ビデオエンコーダ20およびビデオデコーダ30は、ブロック238B、238Cをブロック246B、246Cで代替でき、この場合、ブロック246B、246Cは、それぞれブロック238B、238Cと同じ水平位置を有するが、ブロック246B、246CをMER230の外に配置する垂直位置を有する。これはまた、PU232、具体的には、ビデオエンコーダ20およびビデオデコーダ30がそれぞれブロック244B、244Cで代替し得るPU232の隣接ブロック234B、234Cに当てはまる。

40

【0152】

[0167] それぞれブロック238B、238C、234B、234Cをブロック246B、246C、244B、244Cで代替することは、現在のブロックの隣接ブロックが現在のブロックの右上隣接ブロックに対応するときに、ビデオエンコーダ20およびビデオデコーダ30が、現在のブロックの右上隣接ブロックをMERの上隣接ブロックで代替でき、結果的にMERの上隣接ブロックが現在のブロックの右上隣接ブロックと同じ水平位置を有するようになる例を表している。

【0153】

50

【0168】 P UのB 2の隣接（左上）ブロックがM E R内にあるとき、そのブロックは利用不可能と見なされ得る。ビデオエンコーダ2 0およびビデオデコーダ3 0は、水平方向または垂直方向にあるが、M E R 2 3 0の外にある位置合わせされたブロックのいずれかで代替できる。これは、ブロック2 4 0、2 4 2が代替として使用され得る、図1 0のP U 2 3 2の隣接ブロック2 3 4 Aに当てはまる。ブロック2 3 4 Aをブロック2 4 0で代替することは、現在のブロックの隣接ブロックが現在のブロックの左上隣接ブロックに対応するときに、ビデオエンコーダ2 0およびビデオデコーダ3 0が、現在のブロックの左上隣接ブロックをM E Rの左隣接ブロックで代替でき、結果的にM E Rの左隣接ブロックが現在のブロックの左上隣接ブロックと同じ垂直位置を有するようになる例を表している。ブロック2 3 4 Aをブロック2 4 2で代替することは、現在のブロックの隣接ブロックが現在のブロックの左上隣接ブロックに対応するときに、現在のブロックの左上隣接ブロックをM E Rの上隣接ブロックで代替でき、結果的にM E Rの上隣接ブロックが現在のブロックの左上隣接ブロックと同じ水平位置を有するようになる例を表している。

10

【0 1 5 4】

【0169】 このようにして、図8の例の代替として、ビデオエンコーダ2 0およびビデオデコーダ3 0によって行われる視差動き生成方法は、様々な隣接ブロックを確認し得る。この場合、P Uの隣接ブロックがM E R内にあるとき、ビデオエンコーダ2 0およびビデオデコーダ3 0は、M E R内にあるP Uの隣接ブロックを、P Uを包含するM E Rの同様のロケーションまたは方向における隣接ブロックで代替できる。

20

【0 1 5 5】

【0170】 図1 1は、P Uの視差ベクトルに関する1つまたは複数の候補ブロックがP Uを含むC U内にある例を示す概念図である。この例は、P Uレベル視差ベクトル生成とC Uレベル視差生成との統一に関する。この例では、所与のP U（例えば、現在のP U 2 6 0）の場合、空間的隣接ブロック2 6 6、2 7 4、2 7 8を確認する代わりに、現在のP U 2 6 0を包含するC U 2 8 0の空間的隣接ブロックのみが確認され得る。

【0 1 5 6】

【0171】 図1 1に示す例に関して、確認されることになっていた現在のP U 2 6 0の隣接ブロック2 6 6、2 7 4、2 7 8は、C U 2 8 0の隣接ブロックでない可能性があるか、またはC UのB 0、B 1、B 2、A 0、A 1のロケーション（ロケーションB 0、B 1、B 2、A 0、A 1は、図9の説明に関連して定義されている）のいずれかに属さない可能性があるため、今は確認されない。上述のChenらによる「Motion Parameter Prediction and Coding for Multiview Video」で説明されているように、視差ベクトル生成は他のタイプの視差ベクトル、例えば時間的視差ベクトルを確認し得る。このようにして、この例は、P UおよびC Uの空間的隣接ブロックを統一でき、それにより、そのような空間的隣接ブロックは、空間的視差ベクトル確認についての同じプロセスを共有し、同じ結果をもたらすことができる。

30

【0 1 5 7】

【0172】 従って、C U 2 8 0は、現在のP U 2 6 0を包含する包含ブロックの例を表している。ビデオエンコーダ2 0およびビデオデコーダ3 0は、現在のP U 2 6 0の動きベクトルをコーディングするために、ブロック2 6 6をブロック2 6 4で、ブロック2 7 4をブロック2 7 2で、ブロック2 7 8をブロック2 7 6で代替できる。このようにして、P U 2 6 0、2 6 2は並列にコーディングされ得る。

40

【0 1 5 8】

【0173】 いくつかの例で、ビデオコーダは、M E Rの場合に図1 1に関して説明した技法と同様の技法を行なえ、例えば、M E R内のP Uの場合、P Uの空間的視差ベクトル確認のプロセスおよび結果は、P Uを包含するM E Rの場合と同じであり得る。

【0 1 5 9】

【0174】 いくつかの例では、M E RおよびP U / C Uの視差ベクトル生成技法が統一され得る。M E RおよびC Uの最大サイズは、空間的視差ベクトル確認に使用される視差生

50

成ブロックユニットのサイズになるように選択され得る。上記の例と同様に、MERまたはCUの中の各PUまたはCUは、空間的視差ベクトル確認についての同じプロセスおよび結果を共有し得る。

【0160】

[0175] 例えば、現在のCUのサイズが 16×16 であり、MERのサイズが 32×32 であると仮定する。この場合、空間的視差ベクトルを得るために使用される視差生成ブロックユニットは 32×32 のサイズを有し、現在のPUをカバーし、この 32×32 ブロックユニット内の各CUおよびPUは、空間的視差ベクトル確認についての同じプロセスおよび結果を共有する。

【0161】

[0176] 別の例として、現在のCUのサイズが 16×16 であり、MERのサイズが 4×4 であると仮定する。この場合、空間的視差ベクトルを得るために使用される視差生成ブロックユニットは 16×16 のサイズを有し、現在のPUをカバーし、この 16×16 コーディングユニット内の各PUは、空間的視差ベクトル確認についての同じプロセスおよび結果を共有する。

【0162】

[0177] 図12は、視差生成ブロックユニットのための視差ベクトル生成の一例を示す概念図である。この例で、視差生成ブロックユニットは、現在のCUおよび任意のPUを包含するMERのうちの大きい方として定義され得る。視差生成ブロックユニットが $2M \times 2M$ ピクセルのサイズを有し、左上ピクセル座標が (x_0, y_0) であると仮定する。視差生成ブロックユニット(DGBU)300の $(T_0 \quad 312$ および $T_1 \quad 314$ における)上隣接 (4×4) ブロックおよび $(L_0 \quad 316$ および $L_1 \quad 318$ における)左隣接ブロックは、所与のPUを予測するために使用され得る。

【0163】

[0178] 上隣接ブロックは2つの部分、 $T_0 \quad 312$ および $T_1 \quad 314$ に分けられ、 $T_0 \quad 312$ は、DGBU300の幅の第1の半分(M)内のブロックをカバーし、 $T_1 \quad 314$ は、DGBU300の幅の第2の半分(M)を含むブロックとDGBU300の右上隣接ブロックであるもう1つのブロックとをカバーする。

【0164】

[0179] 左隣接ブロックは2つの部分、 $L_0 \quad 316$ および $L_1 \quad 318$ に分けられ、 $L_0 \quad 316$ は、DGBU300の高さの第1の半分(M)内のブロックをカバーし、 $L_1 \quad 318$ は、DGBU300の高さの第2の半分(M)を含むブロックとDGBU300の左下隣接であるもう1つのブロックとをカバーする。

【0165】

[0180] B310として示された左上ブロックは、例えば領域のズーム0 302、ズーム1 304、ズーム2 306、ズーム3 308のうちの1つの中の、PUまたはCUのロケーションに応じて、 $T_0 \quad 312$ または $L_0 \quad 314$ のいずれかの一部であり得る。代替として、B310は常に $T_0 \quad 312$ に属し得る。代替として、Bは常に $L_0 \quad 316$ に属し得る。従って、隣接領域について $T_0 \quad 312$ 、 $T_1 \quad 314$ 、 $L_0 \quad 316$ 、および $L_1 \quad 318$ が示され得る。

【0166】

[0181] 確認される4つの領域の順序は、 (dx, dy) として示される、DGBU300の左上ピクセル (x_0, y_0) に対する中央のPUまたはCUの選択されたピクセルの相対ロケーションに依存し得る。選択されたピクセルは、PUまたはCUの中央からのものであり得る。まず、DGBU300は、4つのズーム(ズーム0 302、ズーム1 304、ズーム2 306、ズーム3 308)に分けられてよく、各ズームは次のように(図12に示す)、左上ピクセルロケーションおよび右下ピクセルロケーションの相対座標によって表される。ズーム0 302： $(0, 0)$ および (M, M) 、ズーム1 304： $(M+1, 0)$ および $(2M-1, M)$ 、ズーム2 306： $(0, M+1)$ および $(M, 2M-1)$ 、並びにズーム3 308： $(M+1, M+1)$ および $(2M-1$

10

20

30

40

50

, $2M - 1$)。代替として、選択されたピクセルは、PUまたはCUの左上隅であってよく、4つのズームは相応に再定義され得る。

【0167】

[0182] $T0$ 312、 $T1$ 314、 $L0$ 316、および $L1$ 318の順序は次のように定義され得る。 (dx, dy) がズーム0 312にあり、 $dx = dy$ である場合、順序は、 $L0$ 316、 $T0$ 312 312、 $L1$ 318 318、 $T1$ 314 314であり得る。代替として、 $L0$ 316 316および $T0$ 312 312の順序は交換可能なことがあり、かつ $L1$ 318 318および $T1$ 314 314の順序は交換可能なことがある。 (dx, dy) がズーム0 302にあり、 $dx < dy$ である場合、順序は、 $L0$ 316、 $T0$ 312、 $L1$ 318、 $T1$ 314であってよく、代替として、順序は、 $L0$ 316、 $L1$ 318、 $T0$ 312、 $T1$ 314であり得る。 (dx, dy) がズーム0 302にあり、 $dx > dy$ である場合、順序は、 $T0$ 312、 $L0$ 316、 $T1$ 314、 $L1$ 318であってよく、代替として、順序は、 $T0$ 312、 $T1$ 314、 $L0$ 316、 $L1$ 318であり得る。 (dx, dy) がズーム1 304にある場合、順序は、 $T1$ 314、 $T0$ 312、 $L0$ 316、 $L1$ 318であり得る。 (dx, dy) がズーム2 306にあり、 $dx = dy$ である場合、順序は、 $L1$ 318、 $T1$ 314、 $L0$ 316、 $T0$ 312であってよく、代替として、 $L0$ 316および $T0$ 312の順序は交換可能であることがあり、かつ $L1$ 318および $T1$ 314の順序は交換可能であることがある。 (dx, dy) がズーム3 308にあり、 $dx < dy$ である場合、順序は、 $L1$ 318、 $T1$ 314、 $L0$ 316、 $T0$ 312であってよく、代替として、順序は、 $L1$ 318、 $T1$ 314、 $L0$ 316、 $T0$ 312であり得る。 (dx, dy) がズーム3にあり、 $dx > dy$ である場合、順序は、 $T1$ 314、 $L1$ 318、 $T0$ 312、 $L0$ 316であってよく、代替として、順序は、 $T1$ 314、 $T0$ 312、 $L1$ 318、 $L0$ 316であり得る。

【0168】

[0183] 領域ごとに、ビデオコーダはすべてのブロック(4×4)を、それが上の領域に属するか、または左の領域に属するかに応じて、左から右に、または上から下に確認できる。視差ベクトルが利用可能であるとき、視差ベクトルが生成され、領域全体の走査が終了し、生成プロセスはこの視差ベクトルを返すことができる。視差ベクトルが利用不可能なとき、生成プロセスは空を返すことができる。

【0169】

[0184] 視差ベクトルの利用可能性は、以下が真である場合に利用可能と定義され得る。ブロック中の少なくとも1つの動きベクトルが視差動きベクトルであり、視差動きベクトルが、参照ビューおよびターゲットビューに関してターゲット視差ベクトルと一致する。代替として、視差ベクトルの利用可能性は、ブロック中の少なくとも1つの動きベクトルが視差動きベクトルである場合に利用可能と定義され得る。

【0170】

[0185] ビデオコーダは、任意のPUの空間的視差ベクトルを生成するように構成され得る。まず、各領域の視差ベクトルが生成され、一時的に記憶されてよく、そのためこの例では最大4つの視差ベクトルがあり得る。PUまたはCUごとに、 $L0$ 316、 $L1$ 318、 $T0$ 312、 $T1$ 314の順序が上述のように決定され得るので、各領域は所与の順序で確認され得る。領域の各々が確認され、その対応する視差ベクトル(すでに生成されているか、または空として知られている)が空でないとき、それはPUまたはCUの視差ベクトルに設定されてよく、プロセス全体が停止し得る。視差ベクトルが発見されない場合、プロセス全体は空(視差ベクトルを利用できない)を返すことができる。

【0171】

[0186] ビデオコーダは、左上ブロック「B310」を割り振ることもできる。ブロックBがいずれの領域に属するかに関して上述した1つの選択肢は、領域の順序に依存し得る。一例で、ビデオコーダは、いずれの領域が最初に確認されるかに応じて、B310を

L 0 3 1 6 または T 0 3 1 2 に割り振る。

【0172】

[0187] 代替例で、L 0 3 1 6 および L 1 3 1 8 は L に結合されてよく、かつ T 0 3 1 2 および T 1 3 1 4 は T に結合されてよい。一方、B 3 1 0 は、場合により T または L のいずれかから除去され得る。 (dx, dy) は依然として、選択されたピクセルの相対座標であり得る。確認対象領域の順序は、B、 (dx, dy) の場合に L、そうでない場合に T、および (dx, dy) の場合に T、そうでない場合に L と定義され得る。代替として、確認対象領域の順序は、 (dx, dy) の場合に L、そうでない場合に T、 (dx, dy) の場合に T、そうでない場合に L、そして B であり得る。

【0173】

[0188] 別の代替で、L または T 内の走査順序は、左から右または上から下だけではない。 (dx, dy) は依然として、選択されたピクセルの相対座標である。 $dx = M$ である場合、T 中のブロックの走査順序は左から右であり、そうでない場合は右から左であり得る。 $dy = M$ である場合、L 中のブロックの走査順序は上から下であってよく、そうでない場合は下から上である。

【0174】

[0189] 別の代替で、L 0 3 1 6、L 1 3 1 8、T 0 3 1 2 または T 1 3 1 4 内の走査順序は、左から右または上から下だけではない。L 0 3 1 6、L 1 3 1 8、T 0 3 1 2 または T 1 3 1 4 内の順序は、選択されたピクセルの相対座標 (dx, dy) に応じて逆転し得る。例えば、 $M/2 < dx = M$ である場合、T 0 3 1 2 内のブロックの走査順序は右から左であり得る。

【0175】

[0190] 別の代替では、L 0 3 1 6、L 1 3 1 8、T 0 3 1 2 または T 1 3 1 4 中のブロックのすべてが走査されるとは限らず、各領域 (L 0 3 1 6、L 1 3 1 8、T 0 3 1 2 および T 1 3 1 4) の中央のブロックのみが走査され得る。すべての座標が (x_0, y_0) との対比であり得る。L 0 3 1 6 の場合、中央ブロックは $M/2$ の相対 y 座標から始まる。L 1 3 1 8 の場合、中央ブロックは $M * 3/2$ の相対 y 座標から始まる。T 0 3 1 2 の場合、中央ブロックは $M/2$ の相対 x 座標から始まる。T 1 3 1 4 の場合、中央ブロックは $M * 3/2$ の相対 x 座標から始まる。この場合、B は追加的に加えられ得る。従って、全部で、視差ベクトル生成中に空間的ブロックについて、4 つまたは 5 つのブロックのみが確認され得る。

【0176】

[0191] 代替として、上述の結合された各領域 (L および T) の中央ブロックのみが走査され得る。L の場合、中央ブロックは M の相対 y 座標から始まってよく、T の場合、中央ブロックは M の相対 x 座標から始まってよい。さらに、B が追加され得る。従って、全部で、2 つまたは 3 つのブロックのみが確認され得る。

【0177】

[0192] 代替として、各領域 (L および T) の中央ブロックのみが走査され得る。ここでは、L または T は B を含むことができる。

【0178】

[0193] 別の代替で、T 0 3 1 2、T 1 3 1 4、L 0 3 1 6 および L 1 3 1 8 の順序は、視差生成ブロックユニットにおいてすべての PU または CU について固定され得る。確認順序は、L 1 3 1 8、L 0 3 1 6、T 1 3 1 4、T 0 3 1 2、B であり得る。代替として、上記順序の任意の置換が適用され得る。L 1 3 1 8 または L 0 3 1 6 中のブロックの場合、確認順序は、上から下または下から上のいずれかであり得る。T 1 3 1 4 または T 0 3 1 2 中のブロックの場合、確認順序は、左から右または右から左のいずれかであり得る。

【0179】

[0194] 代替として、L または T 内の走査順序は、左から右または上から下だけではないことがある。例えば、L は右から左に走査されてよく、T は下から右に走査されてよい

10

20

30

40

50

。代替として、走査順序の任意の他の置換が適用され得る。

【0180】

[0195] 図13は、1つの領域として結合されたときの近隣ブロックの走査順序の例を示す概念図である。図13の例は、図12のL、T、およびBの領域がさらに結合されて領域N332を形成していることを除いて、図12の上記例と同様である。図13に示すように、ビデオコードはN332を下から上に（図12のB310の位置に）、次いで左から右に走査できる。

【0181】

[0196] 代替的に、L、T、およびBの領域がさらに領域Nに結合されてよく、Nは逆転した順序から（例えば、右から左に、次いで上から下に）走査され得る。

10

【0182】

[0197] 代替的に、順序は（上のブロックでは）右から左に、そして（左のブロックでは）上から下になるように逆転してよく、それが元の順序であるか、または逆転した順序であるかは、 dx dy であるかどうかによって判断され得る。例えば、（ dx dy ）である場合、ビデオコードは元の順序を使用でき、そうでない場合、ビデオコードは逆転した順序を使用できる。

【0183】

[0198] 図14は、時間的視差ベクトル確認のための例示的な方法を示す概念図である。図14は、PUレベル視差ベクトル生成が空間的隣接ブロックについてCUレベルに統一される一例を示している。従って、この例は、PUレベル視差ベクトル生成とCUレベル視差生成との統一に関する。

20

【0184】

[0199] 現在のピクチャ352のLCU360におけるCU362の所与のPU364について、PU364が視差動きベクトル366を使用して予測される場合、例えば、ビデオエンコード20またはビデオデコード30などのビデオコードによって、コロケートCU（CCU）が常に確認され得る。現在のピクチャ352のCU中の各PUまたは各CUについて、CCU356として示されたCUのコロケート領域が識別され得る。図14に示すように、CLCU354によって示された、CCU356と（BR'358によって示された）CCU356の右下4×4ブロックとを包含する最大CUは、時間的隣接ブロックとして使用され得る。

30

【0185】

[0200] 上記3つの領域の順序は、Chenらによる「Motion Parameter Prediction and Coding for Multiview Video」に記述されている順序と同様であってよいことに留意されたい。例えば、順序は、CCU356、CLCU354、そしてBR'358、またはBR'358、CCU356、そしてCLCU354であり得る。

【0186】

[0201] 複数のブロックを含むCCUまたはCLCUを確認するとき、各CCUまたはCLCUの16×16ブロックがラスト走査順序で確認され得る。視差ベクトルが利用可能なとき、確認プロセスは終了し得る。代替として、BR'はCCUにおける中央の4×4ブロックのうちの1つに置き換えられ得る。代替として、BR'はCCUにおける任意の他の4×4ブロックに置き換えられ得る。代替として、BR'は利用不可能に設定され、確認されないことがある。代替として、BR'とCCUの両方が利用不可能に設定され、確認されないことがある。

40

【0187】

[0202] Chenらによる「Motion Parameter Prediction and Coding for Multiview Video」で説明されているように、視差ベクトル生成は他のタイプの視差ベクトル、例えば空間的視差ベクトルを確認し得ることに留意されたい。図14の例は、PUおよびCUの時間的隣接ブロックを統一でき、それにより、それらは、時間的視差ベクトル確認についての同じプロセスと同

50

じ結果とを共有できる。候補ピクチャ選択方法は、Chenらによる「Motion Parameter Prediction and Coding for Multiview Video」で説明されている方法と同じであり得る。

【0188】

[0203] 代替として、CUサイズが 16×16 よりも小さい場合、CCUは、このCUを包含する 16×16 ブロックに拡張され得る。残りのプロセスは同じであるが、拡張されたCCUに適用され得る。

【0189】

[0204] 代替として、プロセスは、Chenらによる「Motion Parameter Prediction and Coding for Multiview Video」で説明されているプロセスと同様であり得るが、PUサイズが 16×16 よりも小さい場合、CPUは、このCUを包含する 16×16 ブロックに拡張され得る。残りのプロセスは同じであるが、拡張されたCCUに適用され得る。

【0190】

[0205] 図15は、本開示の技法による、視差動きベクトルを符号化するための例示的な方法を示すフローチャートである。ビデオエンコーダ20に関して説明するが、他のデバイスまたは構成要素が図15の方法を行うように構成され得ることを理解されたい。その上、ただ1つの例が示されているが、図15の方法の複数の例が、共通包含ブロック（例えば、MERまたは共通CUの複数のPU）の複数のブロックについて実質的に同時に、例えば並列に行われ得ることを理解されたい。

【0191】

[0206] この例で、ビデオエンコーダ20は最初に、ビュー間予測を使用して包含ブロック中の現在のブロックを符号化できる（380）。例えば、様々な試験符号化パスを行った後、ビデオエンコーダ20は最終的に、現在のブロックがビュー間予測を使用して符号化されるべきであると判断できる。従って、ビデオエンコーダ20は、ビュー間参照ピクチャを指す視差動きベクトルを使用して現在のブロックを符号化できる。

【0192】

[0207] ビデオエンコーダ20は、次いで、視差動きベクトルを符号化し得る。そうするために、ビデオエンコーダ20は、視差動きベクトルのための候補動きベクトル予測子を含む候補リストを形成できる。特に、本開示の技法によれば、ビデオエンコーダ20は、現在のブロックの特定の隣接ブロックが、現在のブロックを含む包含ブロック中にあるかどうかを判断できる（382）。隣接ブロックが包含ブロック中ではない場合（382の「いいえ」分岐）、ビデオエンコーダ20は、現在のブロックの隣接ブロックを候補リストに追加できる（384）。一方、隣接ブロックが、現在のブロックをも含む包含ブロック中にある場合（382の「はい」分岐）、ビデオエンコーダ20は、包含ブロックの隣接ブロックを候補リストに追加できる（386）。言い換えれば、ビデオエンコーダ20は、現在のブロックの隣接ブロックを包含ブロックの隣接ブロックで代替できる。ビデオエンコーダ20は、例えば図8～図11に関して上述した様々な例または他の代替方式に従って、この代替を行い得る。

【0193】

[0208] 次いでビデオエンコーダ20は、前の隣接ブロックが、候補リストへの追加について検討されるべき最後の隣接ブロックであったかどうかを判断できる（388）。前の隣接ブロックが検討されるべき最後の隣接ブロックでなかった場合（388の「いいえ」分岐）、ビデオエンコーダ20は、次の隣接ブロック、また特に、上述したように、現在のブロックの次の隣接ブロックを、包含ブロックの隣接ブロックで代替するかどうかを検討できる。一方、前の隣接ブロックが最後の隣接ブロックであった場合（388の「はい」分岐）、ビデオエンコーダ20は、いくつかの例では、場合によっては候補リストをブルーニングした後、候補リストから視差動きベクトル予測子を選択できる（390）。ビデオエンコーダ20は、次いで、視差動きベクトル予測子を使用して、例えばマージモードまたはAMVPモードを使用して、視差動きベクトルを符号化できる（392）。

【 0 1 9 4 】

[0209] このようにして、図 1 5 の方法は、視差動きベクトルを使用して、包含ブロック内のビデオデータの現在のブロックを符号化することと、現在のブロックの隣接ブロックも包含ブロック内にあるとの判断に基づいて、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替することと、候補リスト中の複数のブロックのうちの 1 つから視差動きベクトル予測子を選択することと、視差動きベクトル予測子に基づいて視差動きベクトルを符号化することを含む方法の例を表している。

【 0 1 9 5 】

[0210] 図 1 6 は、本開示の技法による、視差動きベクトルを復号するための例示的な方法を示すフローチャートである。ビデオデコーダ 3 0 に関して説明するが、他のデバイスまたは構成要素が図 1 6 の方法を行うように構成され得ることを理解されたい。その上、ただ 1 つの例が示されているが、図 1 6 の方法の複数の例が、共通包含ブロック（例えば、MER または共通 CU の複数の PU）の複数のブロックについて実質的に同時に、例えば並列に行われ得ることを理解されたい。

【 0 1 9 6 】

[0211] 最初に、ビデオデコーダ 3 0 は、包含ブロック中の現在のブロックがビュー間予測を使用して符号化されていると判断し得る（4 0 0）。そこで、ビデオデコーダ 3 0 は、視差動きベクトルが復号されるべきであると判断し得る。従って、ビデオデコーダ 3 0 は、上記の図 1 5 に関して説明したのと実質的に同じ方法で候補リストを構成できる。例えば、ビデオデコーダ 3 0 は、視差動きベクトルのための候補動きベクトル予測子を含む候補リストを形成できる。特に、本開示の技法によれば、ビデオデコーダ 3 0 は、現在のブロックの特定の隣接ブロックが、現在のブロックを含む包含ブロック中にあるかどうかを判断できる（4 0 2）。隣接ブロックが包含ブロック中にない場合（4 0 2 の「いいえ」分岐）、ビデオデコーダ 3 0 は、現在のブロックの隣接ブロックを候補リストに追加できる（4 0 4）。一方、隣接ブロックが、現在のブロックをも含む包含ブロック中にある場合（4 0 2 の「はい」分岐）、ビデオデコーダ 3 0 は、包含ブロックの隣接ブロックを候補リストに追加できる（4 0 6）。言い換えれば、ビデオデコーダ 3 0 は、現在のブロックの隣接ブロックを包含ブロックの隣接ブロックで代替できる。ビデオデコーダ 3 0 は、例えば図 8 ~ 図 1 1 に関して上述した様々な例または他の代替方式に従って、この代替を行い得る。

【 0 1 9 7 】

[0212] 次いでビデオデコーダ 3 0 は、前の隣接ブロックが、候補リストへの追加について検討されるべき最後の隣接ブロックであったかどうかを判断できる（4 0 8）。前の隣接ブロックが検討されるべき最後の隣接ブロックでなかった場合（3 8 8 の「いいえ」分岐）、ビデオデコーダ 3 0 は、次の隣接ブロック、また特に、上述したように、現在のブロックの次の隣接ブロックを、包含ブロックの隣接ブロックで代替するかどうかを検討できる。一方、前の隣接ブロックが最後の隣接ブロックであった場合（4 0 8 の「はい」分岐）、ビデオデコーダ 3 0 は、いくつかの例では、場合によっては候補リストをブルーニングした後、候補リストから視差動きベクトル予測子を選択できる（4 1 0）。例えば、ビデオデコーダ 3 0 は、候補リストへのインデックスを表す、マージインデックスまたは AMVP インデックスを復号し、候補リストへのインデックスに対応する隣接ブロックの動き情報から視差動きベクトル予測子を選択できる。

【 0 1 9 8 】

[0213] ビデオデコーダ 3 0 は、次いで、視差動きベクトル予測子を使用して、例えばマージモードまたは AMVP モードを使用して、視差動きベクトルを復号できる（4 1 2）。例えば、マージモードで、ビデオデコーダ 3 0 は、現在のブロックの視差動きベクトルが、ステップ 4 1 0 で選択された視差動きベクトル予測子に相当すると判断し得る。代替として、AMVP モードで、ビデオデコーダ 3 0 はさらに、動きベクトル差分値を定義するデータを復号でき、ビデオデコーダ 3 0 は、視差動きベクトル予測子に動きベクトル

差分値を追加することによって、視差動きベクトルを再生できる。

【 0 1 9 9 】

[0214] いずれの場合にも、ビデオデコーダ 3 0 は、次いで、復号された視差動きベクトル予測子を使用して、現在のブロックを復号し得る (4 1 4)。すなわち、ビデオデコーダ 3 0 は、視差動きベクトルによって識別されたビュー間参照ブロックから予測ブロックを生成し、量子化変換係数を定義するシンタックス要素を復号し、これらの係数を逆量子化し逆変換して残差ブロックを再生し、(残差ブロックのピクセル値に予測ブロックのピクセル値を追加することによって)予測ブロックを残差ブロックと結合して最終的に現在のブロックをもたらすことができる。

【 0 2 0 0 】

[0215] このようにして、図 1 6 の方法は、包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断することと、現在のブロックの隣接ブロックも包含ブロック内にあるとの判断に基づいて、候補リスト中で隣接ブロックを、包含ブロックの外にあって包含ブロックに隣接するブロックで代替することと、候補リスト中の複数のブロックのうちの 1 つから視差動きベクトル予測子を選択することと、視差動きベクトル予測子に基づいて視差動きベクトルを復号することを含む方法の例を表している。

【 0 2 0 1 】

[0216] 例によっては、本明細書で説明された技法のうちいずれかの、いくつかの行為またはイベントは、異なるシーケンスで行われ得、追加、マージ、または完全に除外され得る (例えば、すべての説明した作用またはイベントが、本技法の実施のために必要であるとは限らない)ことを認識されたい。さらに、いくつかの例で、行為またはイベントは、連続的にではなく、例えば、マルチスレッド処理、割込み処理、または複数のプロセッサを通して、同時に行われ得る。

【 0 2 0 2 】

[0217] 1 つまたは複数の例で、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で実装され得る。ソフトウェアで実装される場合、機能は、1 つまたは複数の命令またはコードとしてコンピュータ可読媒体上に記憶されるか、またはコンピュータ可読媒体を介して送信され、ハードウェアベースの処理ユニットによって実行され得る。コンピュータ可読媒体は、例えば、通信プロトコルに従って、ある場所から別の場所へのコンピュータプログラムの転送を可能にする任意の媒体を含むデータ記憶媒体または通信媒体などの有形媒体に対応するコンピュータ可読記憶媒体を含み得る。このようにして、コンピュータ可読媒体は、概して、(1) 非一時的である有形コンピュータ可読記憶媒体、または (2) 信号または搬送波などの通信媒体に対応し得る。データ記憶媒体は、本開示で説明した技法の実装のための命令、コードおよび / またはデータ構造を取り出すために 1 つ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り得る。コンピュータプログラム製品はコンピュータ可読媒体を含み得る。

【 0 2 0 3 】

[0218] 限定ではなく例として、そのようなコンピュータ可読記憶媒体は、RAM、ROM、EEPROM (登録商標)、CD-ROM 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、もしくは他の磁気ストレージデバイス、フラッシュメモリ、または命令またはデータ構造の形態の所望のプログラムコードを記憶するために使用され得、コンピュータによってアクセスされ得る、任意の他の媒体を備えることができる。また、いかなる接続もコンピュータ可読媒体と適切に呼ばれる。例えば、命令が、同軸ケーブル、光ファイバーケーブル、ツイストペア、デジタル加入者回線 (DSL)、または赤外線、無線、およびマイクロ波などのワイヤレス技術を使用して、ウェブサイト、サーバ、または他のリモートソースから送信される場合、同軸ケーブル、光ファイバーケーブル、ツイストペア、DSL、または赤外線、無線、およびマイクロ波などのワイヤレス技術は、媒体の定義に含まれる。但し、コンピュータ可読記憶媒体およびデータ記憶媒体は、接続

10

20

30

40

50

、搬送波、信号、または他の一時媒体を含まないが、代わりに非一時的有形記憶媒体を対象とすることを理解されたい。本明細書で使用するディスク(disk)およびディスク(disc)は、コンパクトディスク(disc)(CD)、レーザーディスク(登録商標)(disc)、光ディスク(disc)、デジタル多用途ディスク(disc)(DVD)、フロッピー(登録商標)ディスク(disk)およびブルーレイディスク(disc)を含み、ディスク(disk)は、通常、データを磁氣的に再生し、ディスク(disc)は、データをレーザで光学的に再生する。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。

【0204】

[0219] 命令は、1つまたは複数のデジタル信号プロセッサ(DSP)、汎用マイクロプロセッサ、特定用途向け集積回路(ASIC)、フィールドプログラマブル論理アレイ(FPGA)、または他の等価な集積回路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得る。従って、本明細書で使用する「プロセッサ」という用語は、前述の構造、または本明細書で説明した技法の実装に好適な他の構造のいずれかを指し得る。さらに、いくつかの態様で、本明細書で説明した機能は、符号化および復号のために構成された専用ハードウェアおよび/もしくはソフトウェアモジュール内に与えられ得、または複合コーデックに組み込まれ得る。また、本技法は、1つまたは複数の回路または論理要素において十分に実装され得る。

10

【0205】

[0220] 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(例えば、チップセット)を含む、多種多様なデバイスまたは装置において実装され得る。本開示では、開示する技法を行うように構成されたデバイスの機能的態様を強調するために様々な構成要素、モジュール、またはユニットについて説明したが、それらの構成要素、モジュール、またはユニットを、必ずしも異なるハードウェアユニットによって実現する必要があるとは限らない。むしろ、上記で説明したように、様々なユニットが、好適なソフトウェアおよび/またはファームウェアとともに、上記で説明した1つまたは複数のプロセッサを含めて、コーデックハードウェアユニットにおいて組み合わせられるか、または相互動作ハードウェアユニットの集合によって与えられ得る。

20

【0206】

[0221] 様々な例について説明した。これらおよび他の例は以下の特許請求の範囲内に入る。

30

【図 1】

図 1

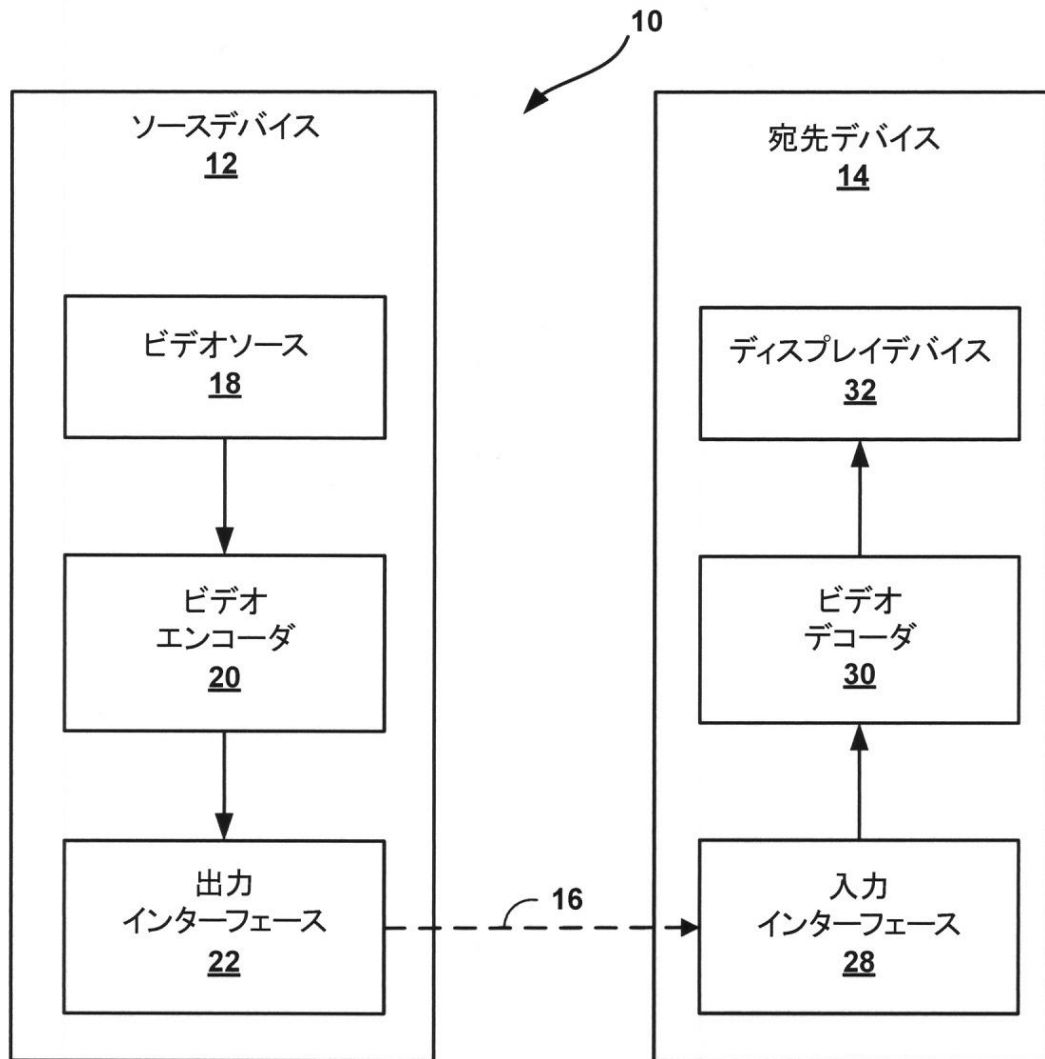


FIG. 1

【図 2】

図 2

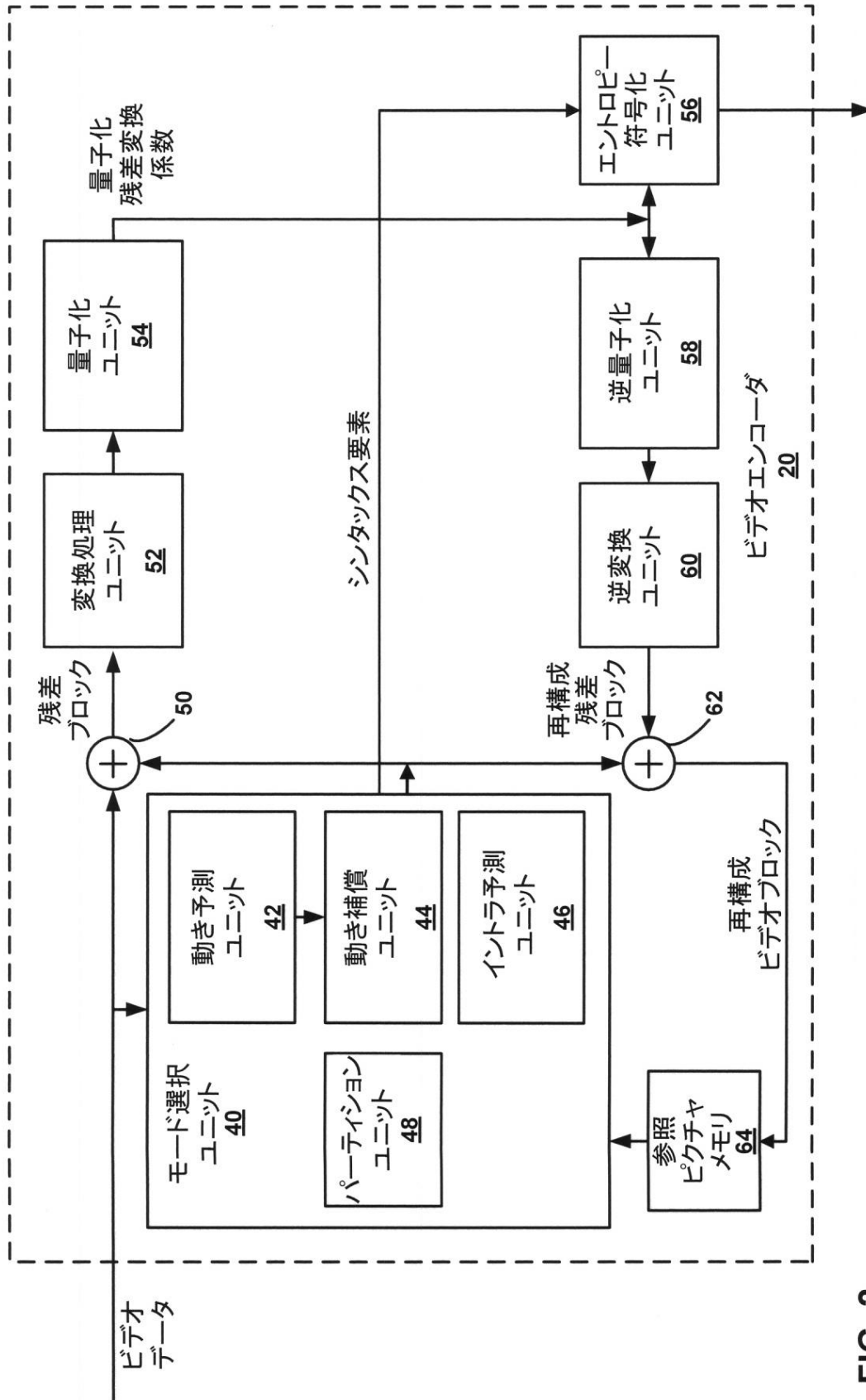


FIG. 2

【図 3】

図 3

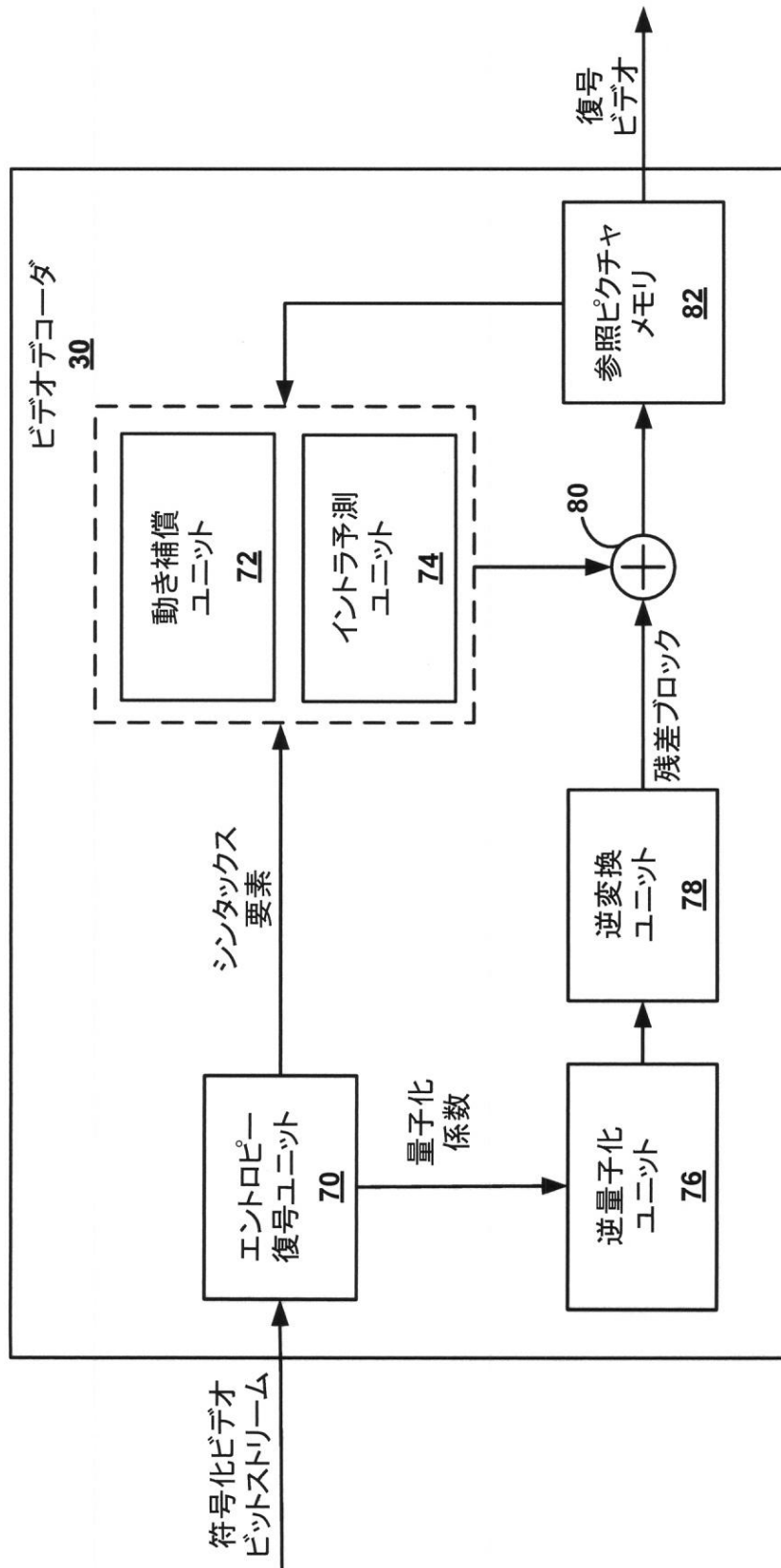


FIG. 3

【図 4】

図 4

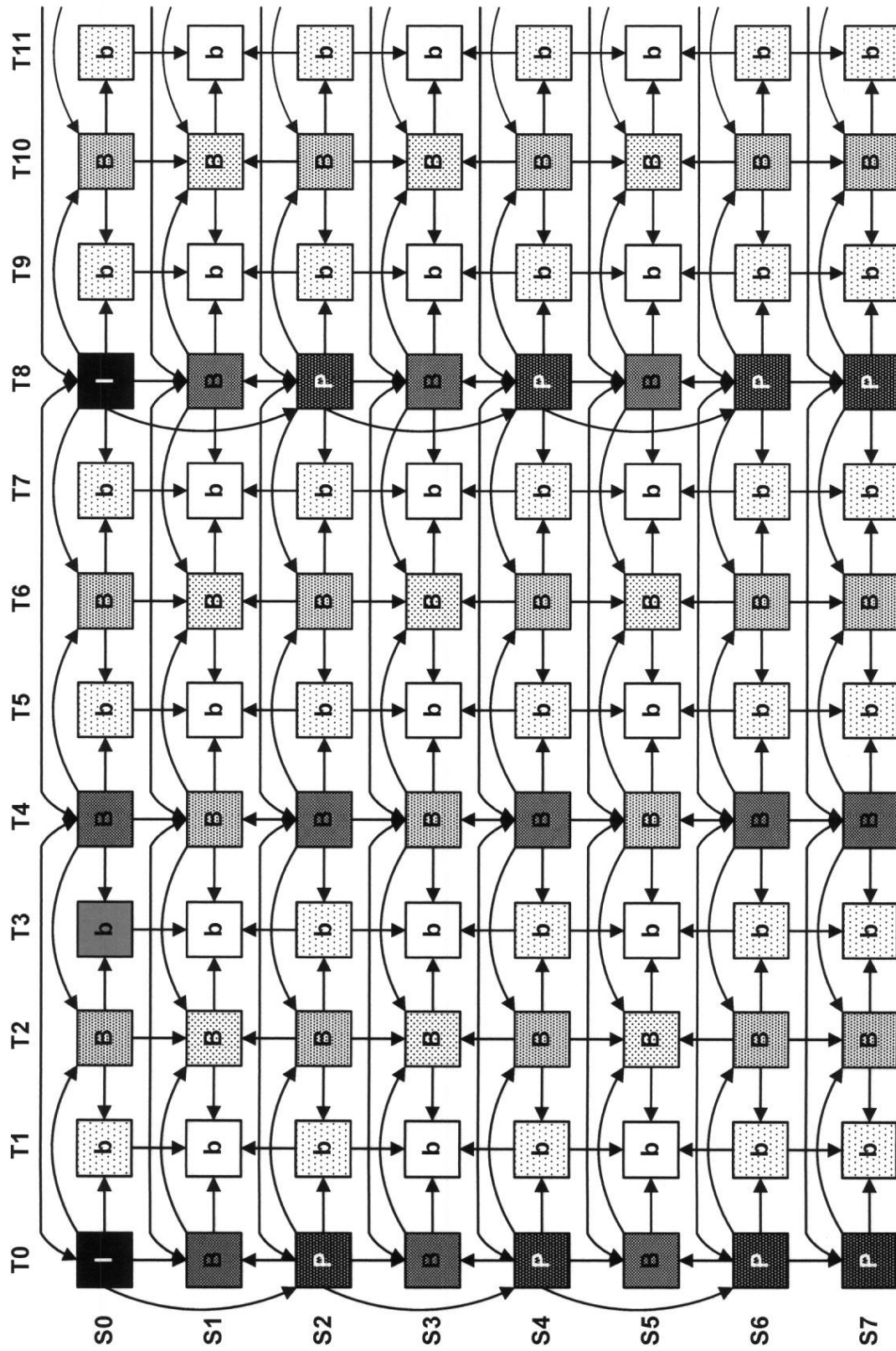


FIG. 4

【図 5】

図 5

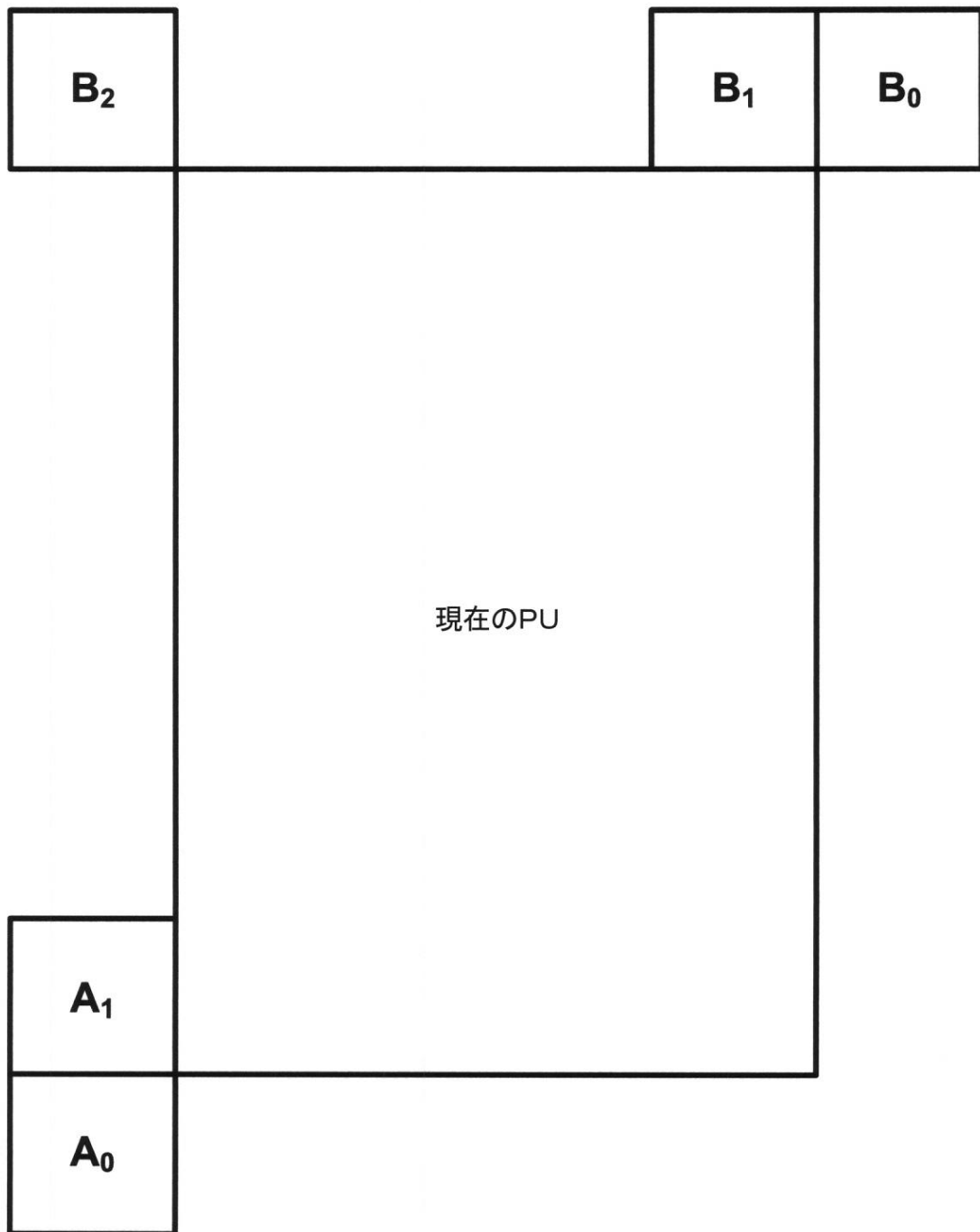


FIG. 5

【図 6】

図 6

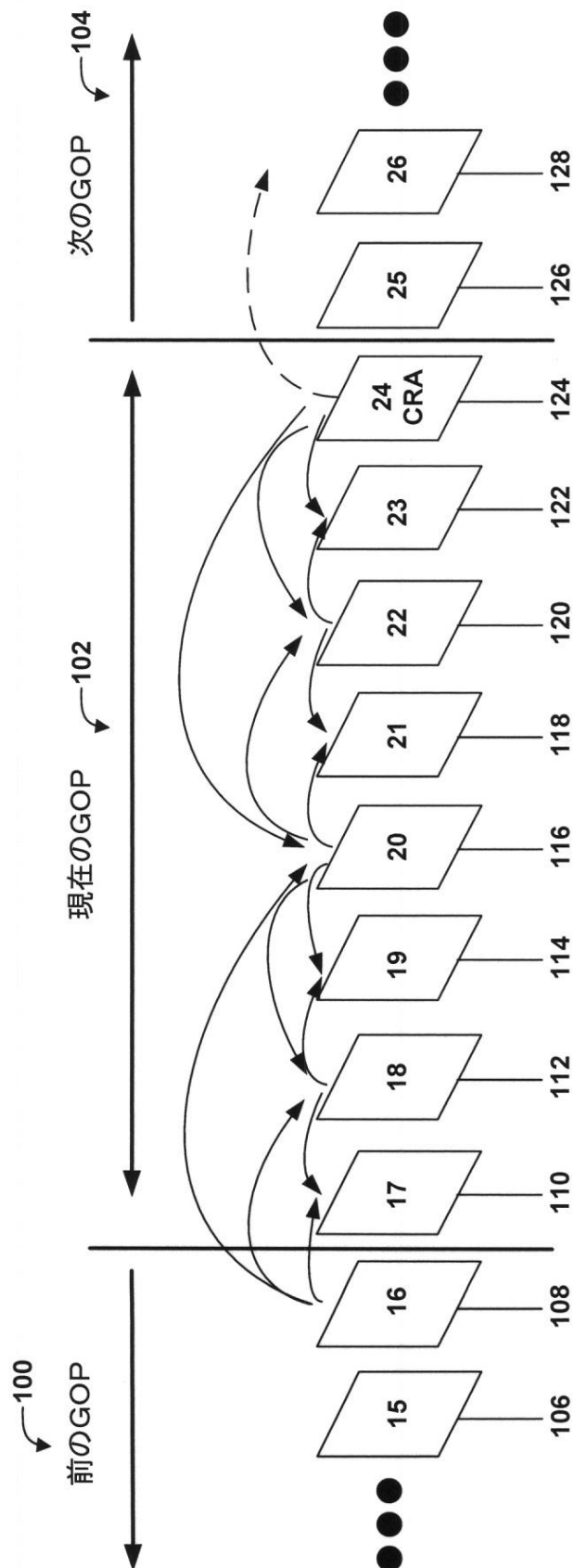


FIG. 6

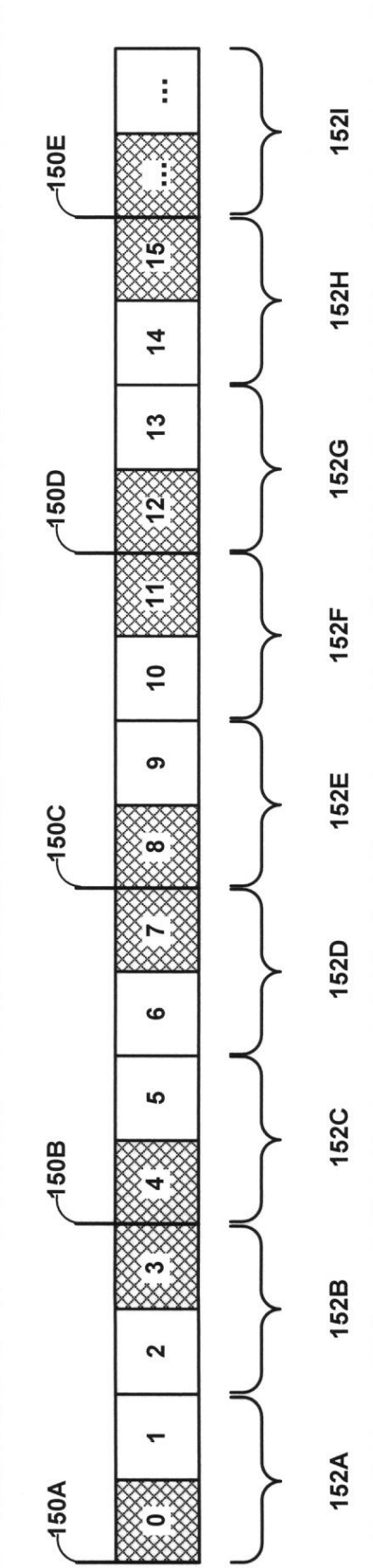


FIG. 7

【図 8】

図 8

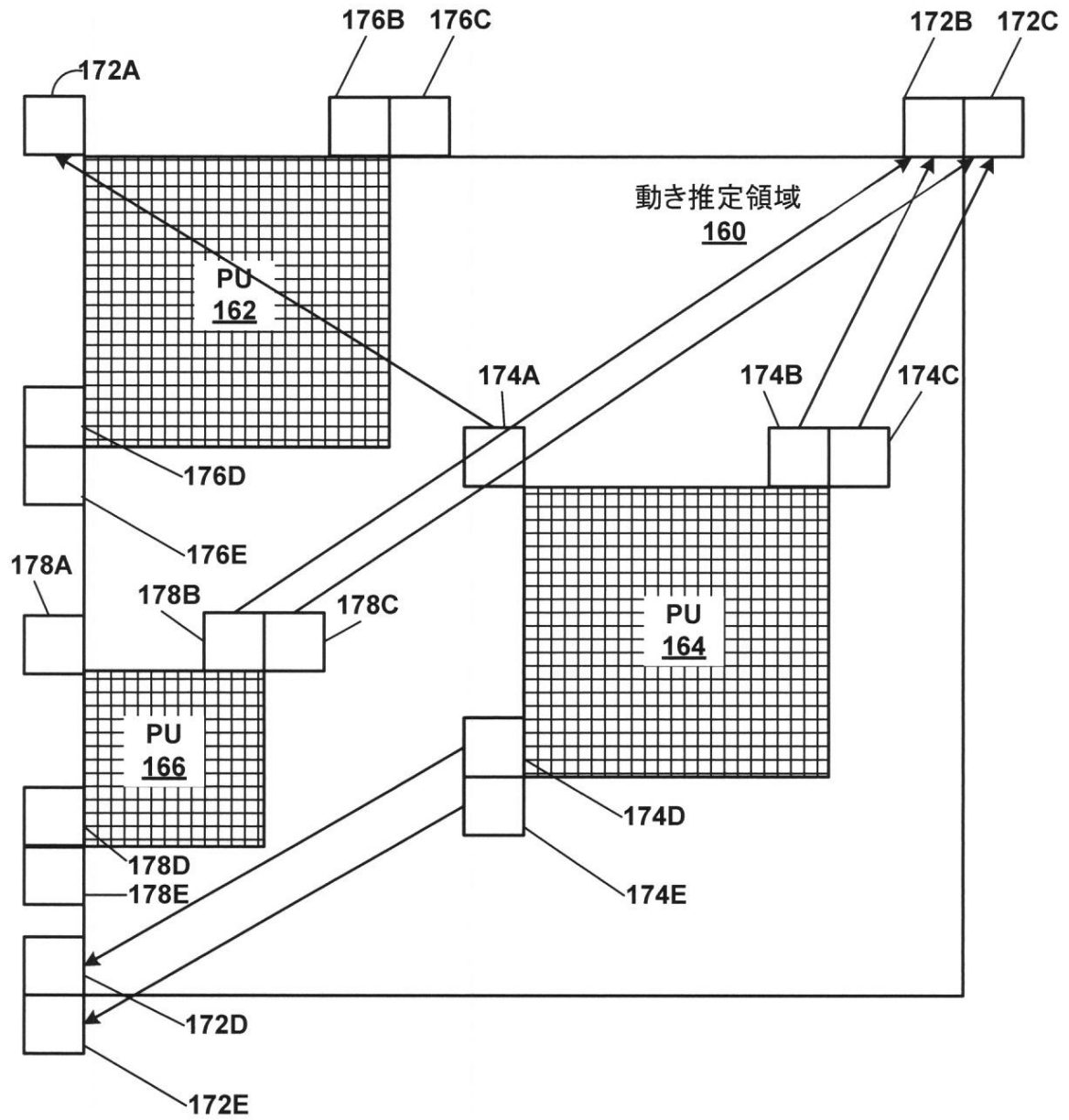


FIG. 8

【 図 9 】

図 9

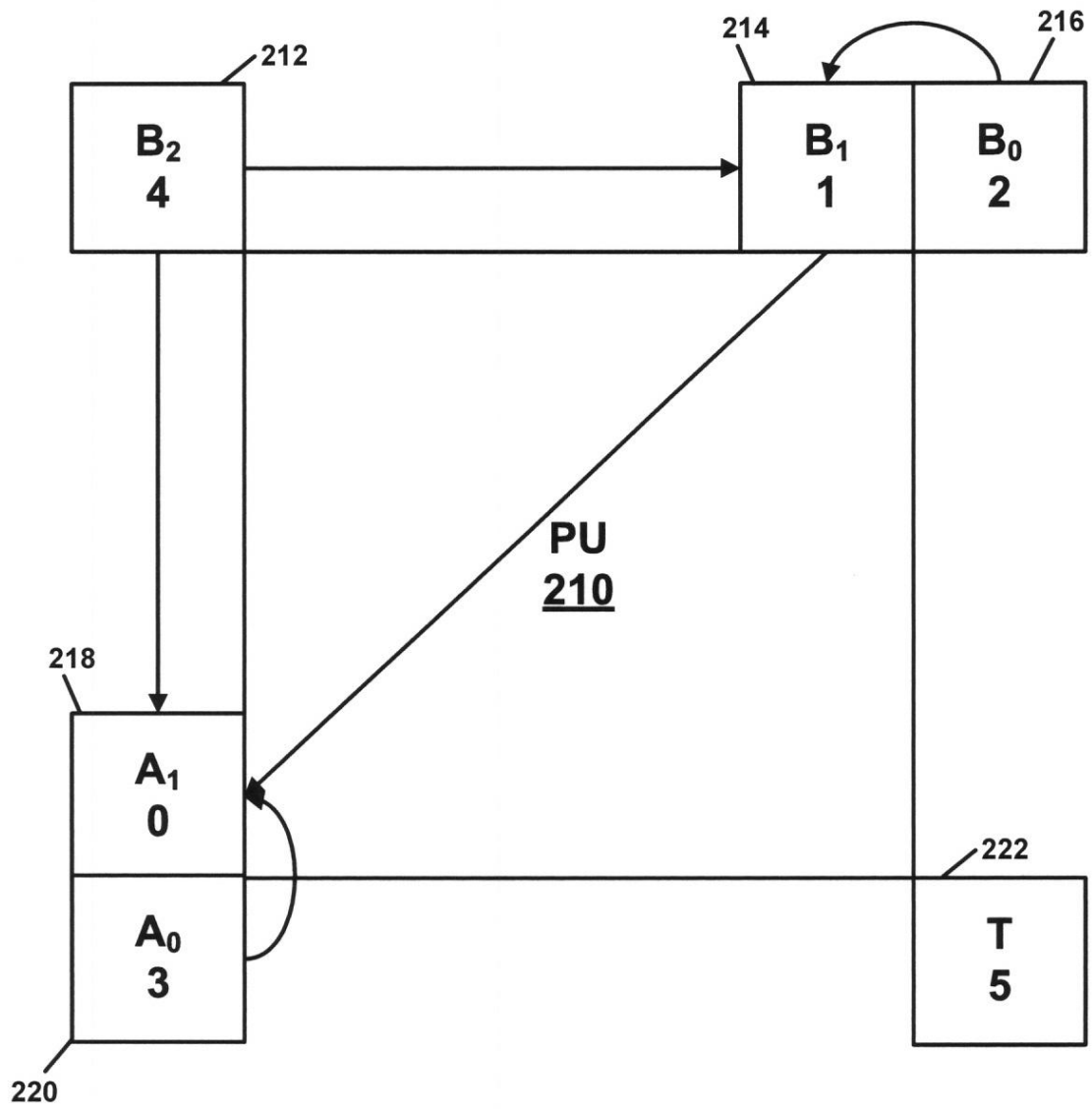


FIG. 9

【図 10】

図 10

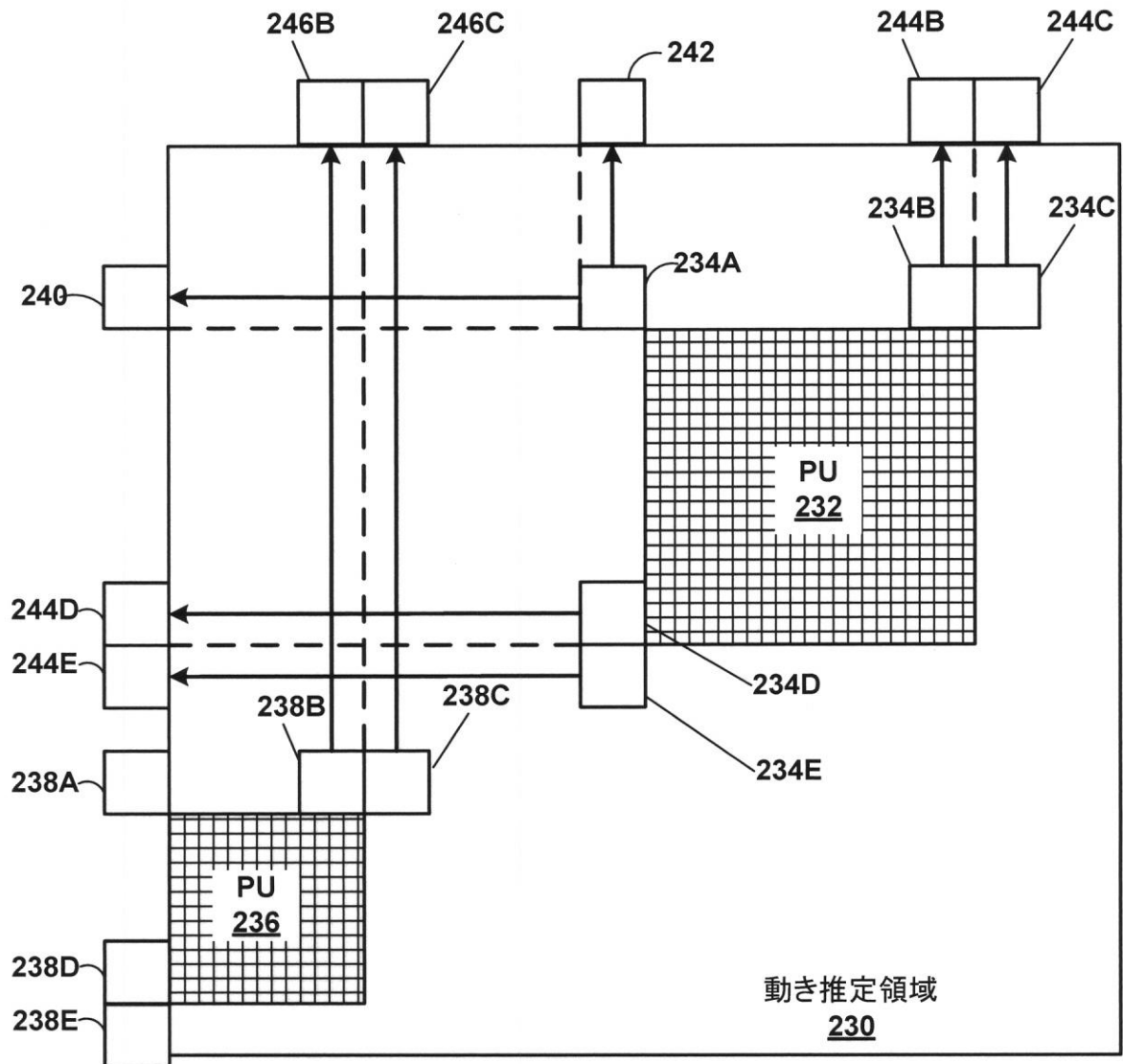


FIG. 10

【図 11】

図 11

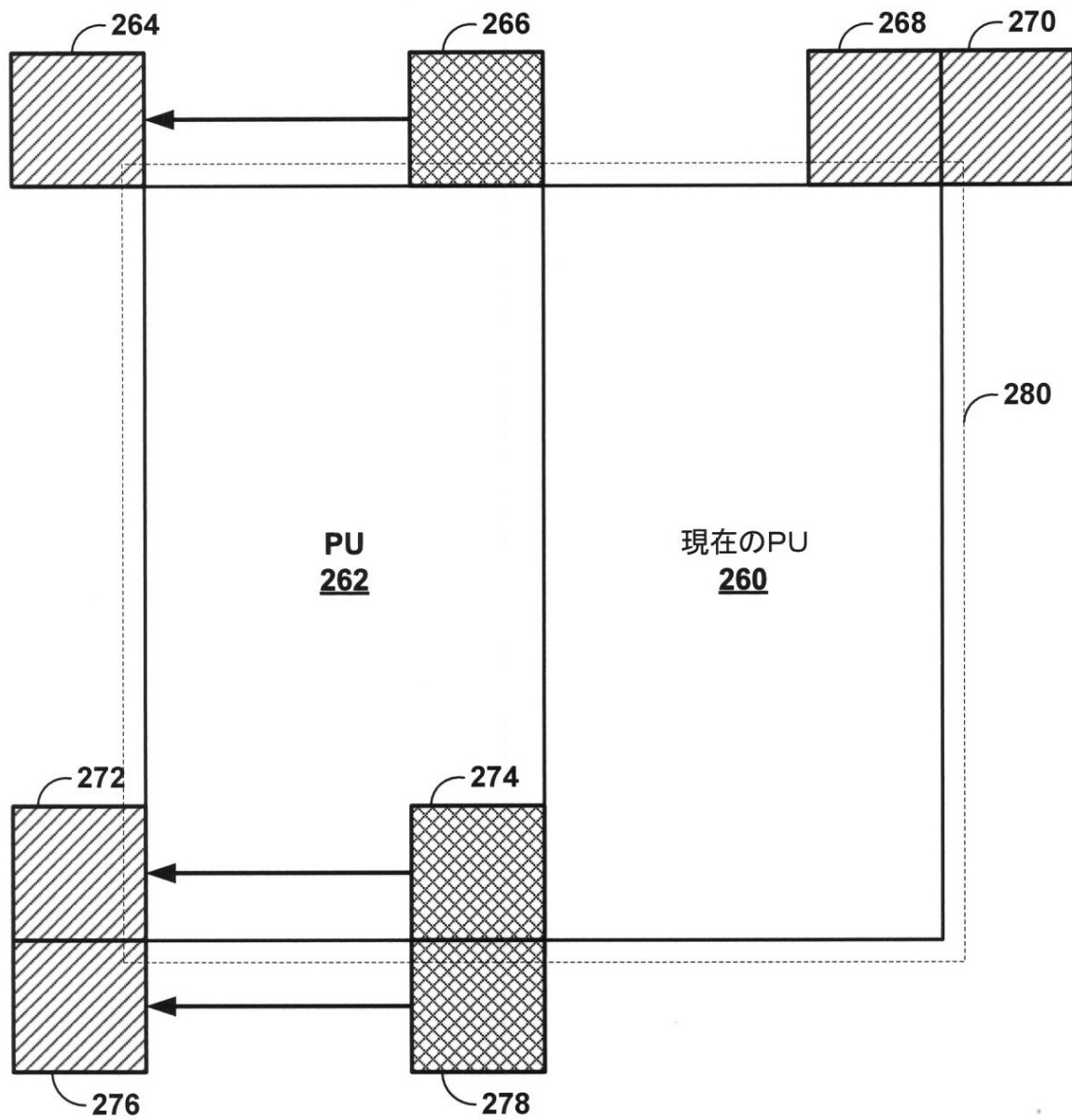


FIG. 11

【図 1 2】

図 12

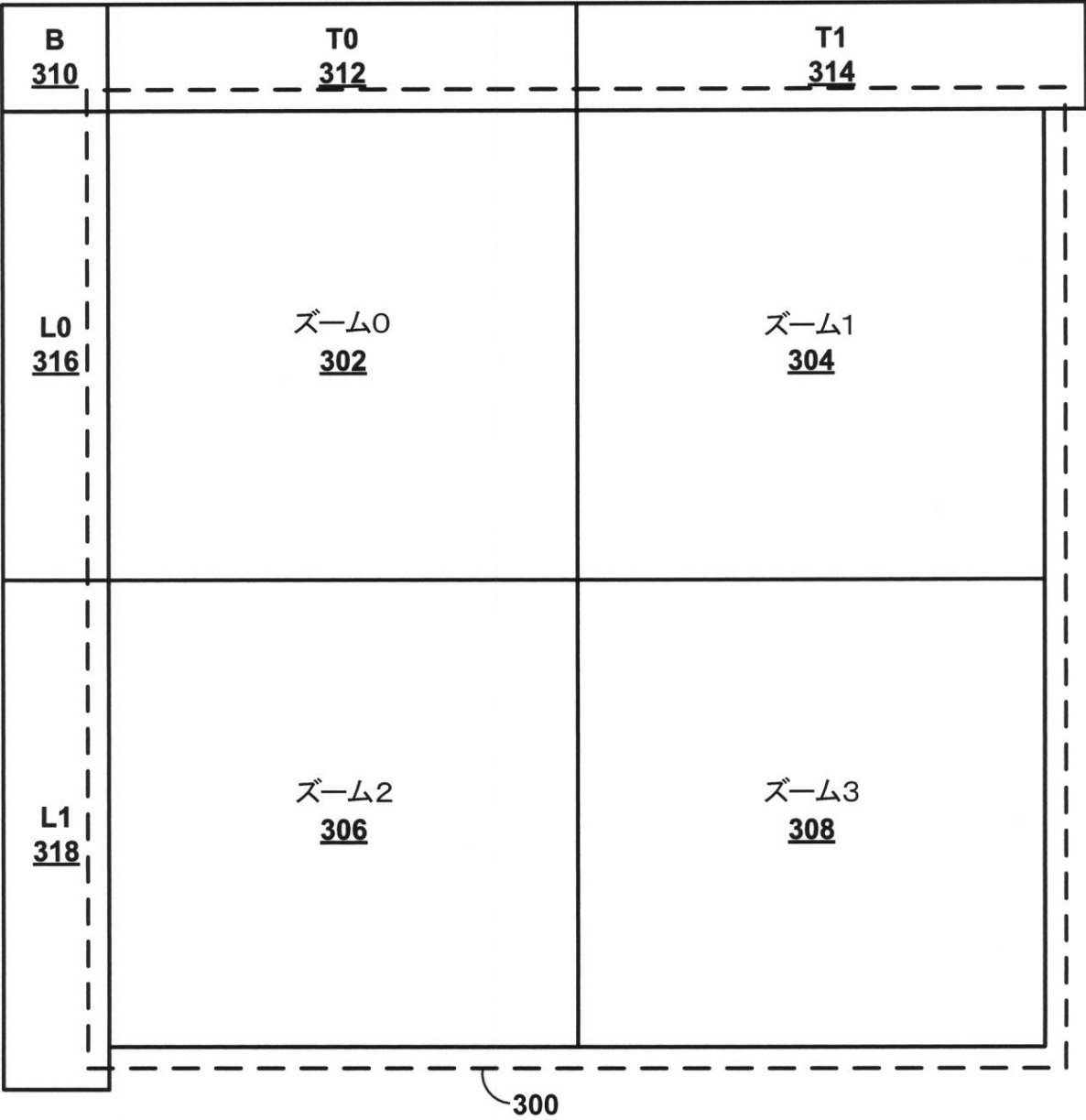


FIG. 12

【図 13】

図 13

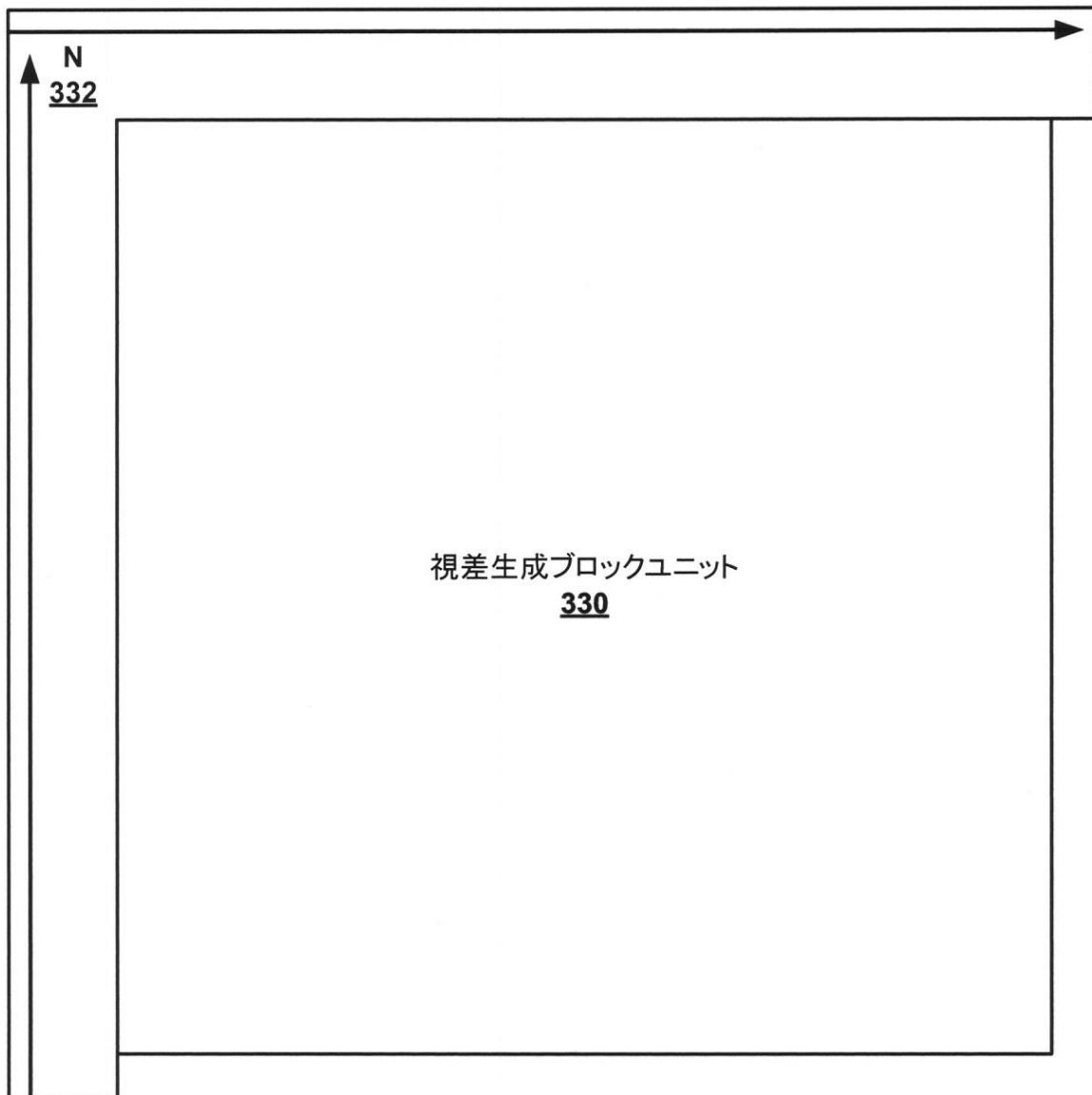


FIG. 13

【図 14】

図 14

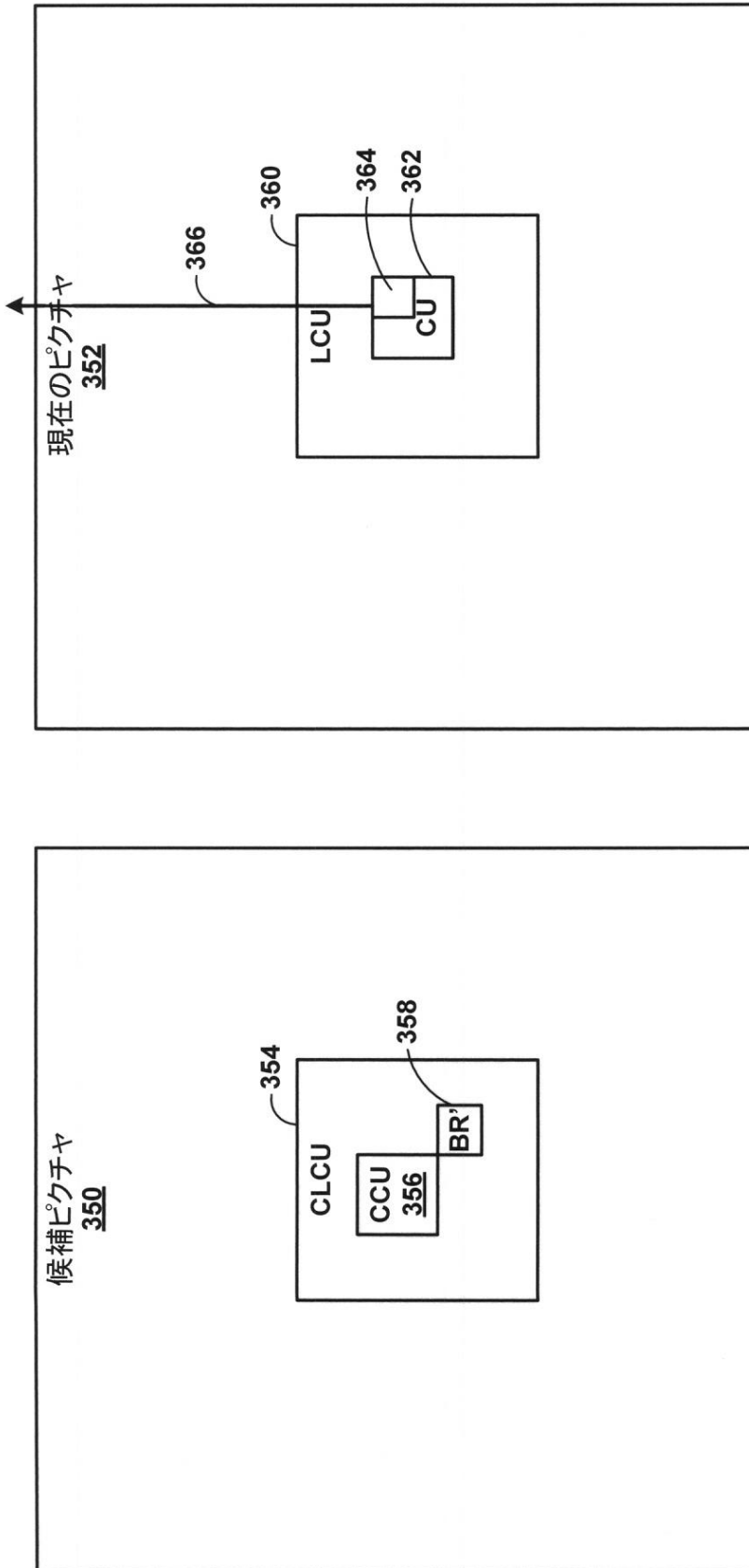


FIG. 14

【図 15】

図 15

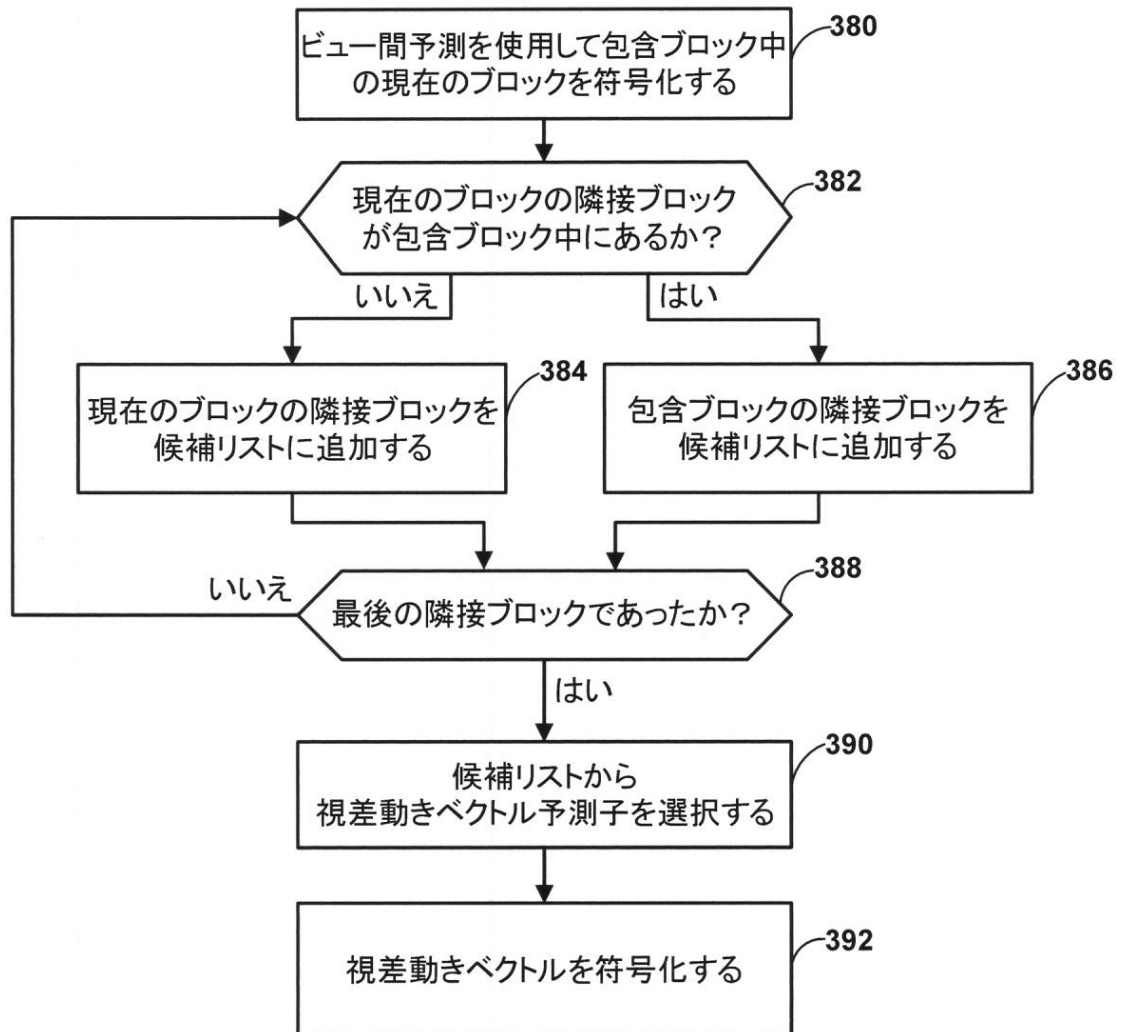


FIG. 15

【図 16】

図 16

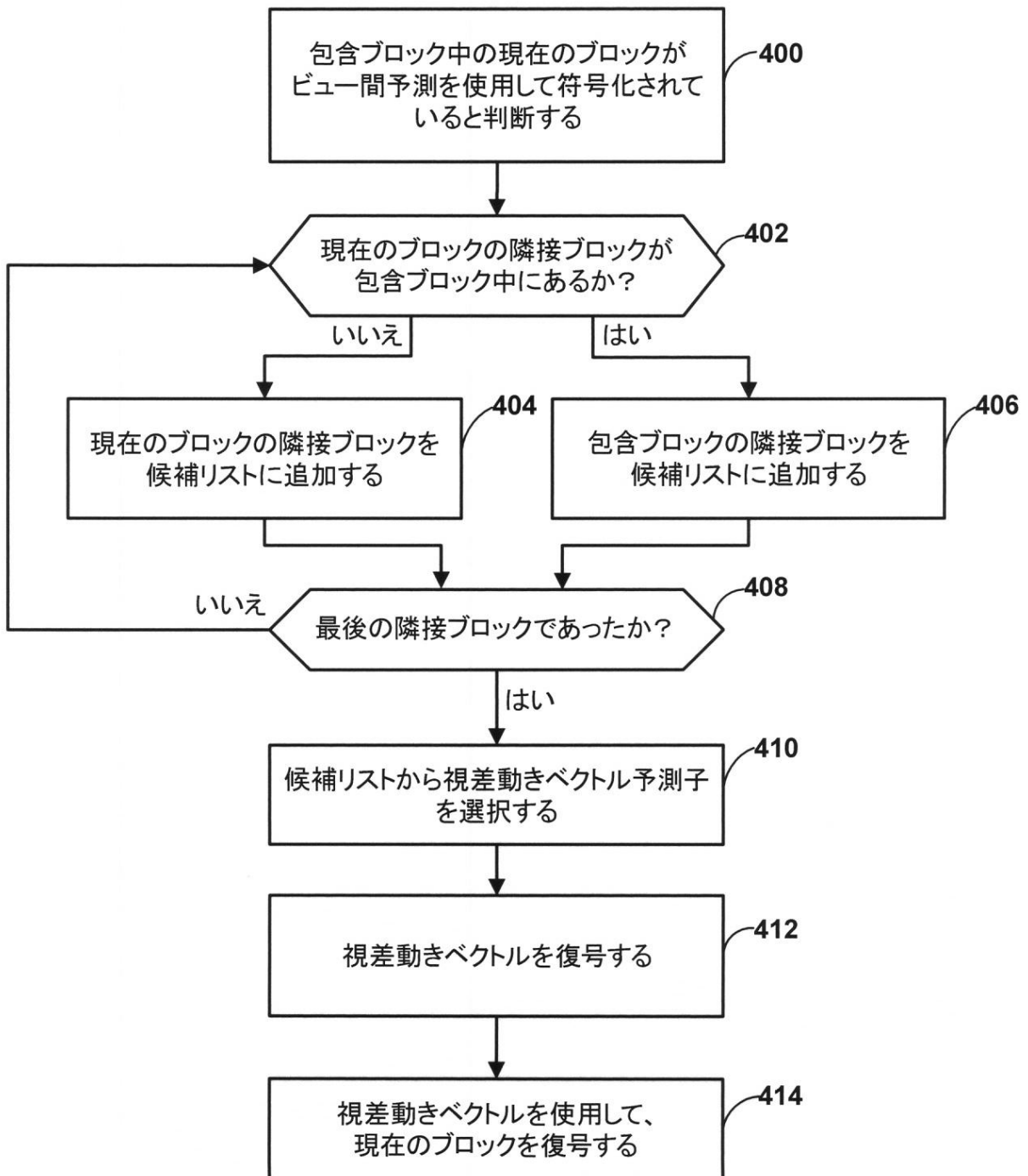


FIG. 16

【提出日】平成27年3月25日(2015.3.25)

【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】

ビデオデータを復号する方法であって、

コーディングユニット(CU)内にある現在の予測ユニット(PU)が視差動きベクトルを使用して符号化されていると判断することと、

前記現在のPUの隣接PUも前記CU内にあるとの判断に基づいて、候補リスト中で前記隣接PUを、前記CUの外にあって前記CUに隣接するブロックで代替することと、

前記候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、

前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号することとを備える方法。

【請求項2】

代替することが、前記隣接PUが前記現在のPUの左下隣接PUに対応するときに、前記現在のPUの前記左下隣接PUを前記CUの左下隣接PUで代替することを備える、請求項1に記載の方法。

【請求項3】

代替することが、前記隣接PUが前記現在のPUの左下隣接PUに対応するときに、前記現在のPUの前記左下隣接PUを前記CUの左隣接PUで代替することを備え、結果的に前記CUの前記左隣接PUが前記現在のPUの前記左下隣接PUと同じ垂直位置を有するようになる、請求項1に記載の方法。

【請求項4】

代替することが、前記隣接PUが前記現在のPUの右上隣接PUに対応するときに、前記現在のPUの前記右上隣接PUを前記CUの右上隣接PUで代替することを備える、請求項1に記載の方法。

【請求項5】

代替することが、前記隣接PUが前記現在のPUの右上隣接PUに対応するときに、前記現在のPUの前記右上隣接PUを前記CUの上隣接PUで代替することを備え、結果的に前記CUの前記上隣接PUが前記現在のPUの前記右上隣接PUと同じ水平位置を有するようになる、請求項1に記載の方法。

【請求項6】

代替することが、前記隣接PUが前記現在のPUの左上隣接PUに対応するときに、前記現在のPUの前記左上隣接PUを前記CUの左上隣接PUで代替することを備える、請求項1に記載の方法。

【請求項7】

代替することが、前記隣接PUが前記現在のPUの左上隣接PUに対応するときに、前記現在のPUの前記左上隣接PUを前記CUの左隣接PUで代替することを備え、結果的に前記CUの前記左隣接PUが前記現在のPUの前記左上隣接PUと同じ垂直位置を有するようになる、請求項1に記載の方法。

【請求項8】

代替することが、前記隣接PUが前記現在のPUの左上隣接PUに対応するときに、前記現在のPUの前記左上隣接PUを前記CUの上隣接PUで代替ことを備え、結果的に前記CUの前記上隣接PUが前記現在のPUの前記左上隣接PUと同じ水平位置を有するようになる、請求項1に記載の方法。

【請求項9】

前記視差動きベクトルを使用してビデオデータの前記現在の P Uを復号することをさらに備える、請求項 1 に記載の方法。

【請求項 1 0】

前記視差動きベクトルを復号することが、
動きベクトル差分値を復号することと、

前記視差動きベクトル予測子に前記動きベクトル差分値を追加して前記視差動きベクトルを再生することと

を備える、請求項 1 に記載の方法。

【請求項 1 1】

ビデオデータを復号するためのデバイスであって、
ビデオデータを記憶するように構成されたメモリと、

ビデオデータのコーディングユニット(C U)内にある現在の予測ユニット(P U)が視差動きベクトルを使用して符号化されていると判断することと、前記現在の P U の隣接 P U も前記 C U 内にあるとの判断に基づいて、候補リスト中で前記隣接 P U を、前記 C U の外にあって前記 C U に隣接するブロックで代替することと、前記候補リスト中の複数のブロックのうちの 1 つから視差動きベクトル予測子を選択することと、前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号することとを行うように構成されたビデオデコーダを備えるデバイス。

【請求項 1 2】

前記ビデオデコーダが、前記隣接 P U が前記現在の P U の左下隣接 P U に対応するときに、前記現在の P U の前記左下隣接 P U を前記 C U の左下隣接 P U で代替するように構成される、請求項 1 1 に記載のデバイス。

【請求項 1 3】

前記ビデオデコーダが、前記隣接 P U が前記現在の P U の左下隣接 P U に対応するときに、前記現在の P U の前記左下隣接 P U を前記 C U の左隣接 P U で代替するように構成され、結果的に前記 C U の前記左隣接 P U が前記現在の P U の前記左下隣接 P U と同じ垂直位置を有するようになる、請求項 1 1 に記載のデバイス。

【請求項 1 4】

前記ビデオデコーダが、前記隣接 P U が前記現在の P U の右上隣接 P U に対応するときに、前記現在の P U の前記右上隣接 P U を前記 C U の右上隣接 P U で代替するように構成される、請求項 1 1 に記載のデバイス。

【請求項 1 5】

前記ビデオデコーダが、前記隣接 P U が前記現在の P U の右上隣接 P U に対応するときに、前記現在の P U の前記右上隣接 P U を前記 C U の上隣接 P U で代替するように構成され、結果的に前記 C U の前記上隣接 P U が前記現在の P U の前記右上隣接 P U と同じ水平位置を有するようになる、請求項 1 1 に記載のデバイス。

【請求項 1 6】

前記ビデオデコーダが、前記隣接 P U が前記現在の P U の左上隣接 P U に対応するときに、前記現在の P U の前記左上隣接 P U を前記 C U の左上隣接 P U で代替するように構成される、請求項 1 1 に記載のデバイス。

【請求項 1 7】

前記ビデオデコーダが、前記隣接 P U が前記現在の P U の左上隣接 P U に対応するときに、前記現在の P U の前記左上隣接 P U を前記 C U の左隣接 P U で代替するように構成され、結果的に前記 C U の前記左隣接 P U が前記現在の P U の前記左上隣接 P U と同じ垂直位置を有するようになる、請求項 1 1 に記載のデバイス。

【請求項 1 8】

前記ビデオデコーダが、前記隣接 P U が前記現在の P U の左上隣接 P U に対応するときに、前記現在の P U の前記左上隣接 P U を前記 C U の上隣接 P U で代替するように構成され、結果的に前記 C U の前記上隣接 P U が前記現在の P U の前記左上隣接 P U と同じ水平位置を有するようになる、請求項 1 1 に記載のデバイス。

【請求項 19】

前記ビデオデコーダが、前記視差動きベクトルを使用してビデオデータの前記現在の P Uを復号するようにさらに構成される、請求項 1 1 に記載のデバイス。

【請求項 20】

前記ビデオデコーダが、動きベクトル差分値を復号することと、前記視差動きベクトル予測子に前記動きベクトル差分値を追加して前記視差動きベクトルを再生することとを行うように構成される、請求項 1 1 に記載のデバイス。

【請求項 21】

前記デバイスが、
集積回路、
マイクロプロセッサ、および
ワイヤレス通信デバイス

のうちの少なくとも 1 つを備える、請求項 1 1 に記載のデバイス。

【請求項 22】

ビデオデータを復号するためのデバイスであって、
コーディングユニット (C U) 内にある現在の予測ユニット (P U) が視差動きベクトルを使用して符号化されていると判断するための手段と、

前記現在の P U の隣接 P U も前記 C U 内にあるとの判断に基づいて、候補リスト中で前記隣接 P U を、前記 C U の外にあって前記 C U に隣接するブロックで代替するための手段と、

前記候補リスト中の複数のブロックのうちの 1 つから視差動きベクトル予測子を選択するための手段と、

前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号するための手段とを備えるデバイス。

【請求項 23】

実行されたとき、1 つまたは複数のプロセッサに、

コーディングユニット (C U) 内にある現在の予測ユニット (P U) が視差動きベクトルを使用して符号化されていると判断することと、

前記現在の P U の隣接 P U も前記 C U 内にあるとの判断に基づいて、候補リスト中で前記隣接 P U を、前記 C U の外にあって前記 C U に隣接するブロックで代替することと、

前記候補リスト中の複数のブロックのうちの 1 つから視差動きベクトル予測子を選択することと、

前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号することと
を行わせる命令を記憶したコンピュータ可読記憶媒体。

【請求項 24】

ビデオデータを符号化する方法であって、

視差動きベクトルを使用して、コーディングユニット (C U) 内にある現在の予測ユニット (P U) を符号化することと、

前記現在の P U の隣接 P U も前記 C U 内にあるとの判断に基づいて、候補リスト中で前記隣接 P U を、前記 C U の外にあって前記 C U に隣接するブロックで代替することと、

前記候補リスト中の複数のブロックのうちの 1 つから視差動きベクトル予測子を選択することと、

前記視差動きベクトル予測子に基づいて前記視差動きベクトルを符号化することとを備える方法。

【請求項 25】

代替することが、前記隣接 P U が前記現在の P U の左下隣接 P U に対応するときに、前記現在の P U の前記左下隣接 P U を前記 C U の左下隣接 P U で代替することを備える、請求項 2 4 に記載の方法。

【請求項 26】

代替することが、前記隣接 P U が前記現在の P U の左下隣接 P U に対応するときに、前

記現在の P U の前記左下隣接 P U を前記 C U の左隣接 P U で代替することを備え、結果的に前記 C U の前記左隣接 P U が前記現在の P U の前記左下隣接 P U と同じ垂直位置を有するようになる、請求項 2 4 に記載の方法。

【請求項 2 7】

代替することが、前記隣接ブロックが前記現在の P U の右上隣接 P U に対応するときに、前記現在の P U の前記右上隣接 P U を前記 C U の右上隣接 P U で代替することを備え、請求項 2 4 に記載の方法。

【請求項 2 8】

代替することが、前記隣接ブロックが前記現在の P U の右上隣接 P U に対応するときに、前記現在の P U の前記右上隣接 P U を前記 C U の上隣接 P U で代替することを備え、結果的に前記 C U の前記上隣接 P U が前記現在の P U の前記右上隣接 P U と同じ水平位置を有するようになる、請求項 2 4 に記載の方法。

【請求項 2 9】

代替することが、前記隣接ブロックが前記現在の P U の左上隣接 P U に対応するときに、前記現在の P U の前記左上隣接 P U を前記 C U の左上隣接 P U で代替することを備え、請求項 2 4 に記載の方法。

【請求項 3 0】

代替することが、前記隣接ブロックが前記現在の P U の左上隣接 P U に対応するときに、前記現在の P U の前記左上隣接 P U を前記 C U の左隣接 P U で代替することを備え、結果的に前記 C U の前記左隣接 P U が前記現在の P U の前記左上隣接 P U と同じ垂直位置を有するようになる、請求項 2 4 に記載の方法。

【請求項 3 1】

代替することが、前記隣接ブロックが前記現在の P U の左上隣接 P U に対応するときに、前記現在の P U の前記左上隣接 P U を前記 C U の上隣接 P U で代替することを備え、結果的に前記 C U の前記上隣接 P U が前記現在の P U の前記左上隣接 P U と同じ水平位置を有するようになる、請求項 2 4 に記載の方法。

【請求項 3 2】

前記視差動きベクトルを符号化することが、
前記視差動きベクトルと前記視差動きベクトル予測子との間の差分を計算することと、
前記差分を符号化することと
を備える、請求項 2 4 に記載の方法。

【請求項 3 3】

ビデオデータを符号化するためのデバイスであって、
ビデオデータを記憶するように構成されたメモリと、
視差動きベクトルを使用して、ビデオデータのコーディングユニット(C U)内にある現在の予測ユニット(P U)を符号化することと、前記現在の P U の隣接 P U も前記 C U 内にあるとの判断に基づいて、候補リスト中で前記隣接 P U を、前記 C U の外にあって前記 C U に隣接するブロックで代替することと、前記候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、前記視差動きベクトル予測子に基づいて前記視差動きベクトルを符号化することとを行うように構成されたビデオエンコーダを備えるデバイス。

【請求項 3 4】

前記ビデオエンコーダが、前記隣接 P U が前記現在のブロックの左下隣接 P U に対応するときに、前記現在の P U の前記左下隣接 P U を前記 C U の左下隣接 P U で代替するように構成される、請求項 3 3 に記載のデバイス。

【請求項 3 5】

前記ビデオエンコーダが、前記隣接 P U が前記現在のブロックの左下隣接 P U に対応するときに、前記現在の P U の前記左下隣接 P U を前記 C U の左隣接 P U で代替するように構成され、結果的に前記 C U の前記左隣接 P U が前記現在の P U の前記左下隣接 P U と同じ垂直位置を有するようになる、請求項 3 3 に記載のデバイス。

【請求項 36】

前記ビデオエンコーダが、前記隣接 P U が前記現在の P U の右上隣接 P U に対応するときに、前記現在の P U の前記右上隣接 P U を前記 C U の右上隣接 P U で代替するように構成される、請求項 33 に記載のデバイス。

【請求項 37】

前記ビデオエンコーダが、前記隣接 P U が前記現在の P U の右上隣接 P U に対応するときに、前記現在の P U の前記右上隣接ブロックを前記 C U の上隣接 P U で代替するように構成され、結果的に前記 C U の前記上隣接 P U が前記現在の P U の前記右上隣接 P U と同じ水平位置を有するようになる、請求項 33 に記載のデバイス。

【請求項 38】

前記ビデオエンコーダが、前記隣接 P U が前記現在の P U の左上隣接 P U に対応するときに、前記現在の P U の前記左上隣接 P U を前記 C U の左上隣接 P U で代替するように構成される、請求項 33 に記載のデバイス。

【請求項 39】

前記ビデオエンコーダが、前記隣接 P U が前記現在の P U の左上隣接 P U に対応するときに、前記現在の P U の前記左上隣接 P U を前記 C U の左隣接 P U で代替するように構成され、結果的に前記 C U の前記左隣接 P U が前記現在の P U の前記左上隣接 P U と同じ垂直位置を有するようになる、請求項 33 に記載のデバイス。

【請求項 40】

前記ビデオエンコーダが、前記隣接 P U が前記現在の P U の左上隣接 P U に対応するときに、前記現在の P U の前記左上隣接ブロックを前記 C U の上隣接 P U で代替するように構成され、結果的に前記 C U の前記上隣接 P U が前記現在の P U の前記左上隣接 P U と同じ水平位置を有するようになる、請求項 33 に記載のデバイス。

【請求項 41】

集積回路、
マイクロプロセッサ、および
ワイヤレス通信デバイス

のうちの少なくとも1つを備える、請求項 41 に記載のデバイス。

【手続補正 2】

【補正対象書類名】明細書

【補正対象項目名】0206

【補正方法】変更

【補正の内容】

【0206】

[0221] 様々な例について説明した。これらおよび他の例は以下の特許請求の範囲内に入る。

以下に、本願出願の当初の特許請求の範囲に記載された発明を付記する。

[C 1] ビデオデータを復号する方法であって、

包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断することと、

前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替することと、

前記候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、

前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号することとを備える方法。

[C 2] 前記包含ブロックが動き推定領域 (MER) を備え、前記現在のブロックがコーディングユニット (CU) および予測ユニット (PU) のうちの1つを備える、C 1 に記載の方法。

[C 3] 代替することが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記MERの左下隣接ブロックで代替することを備える、C 2に記載の方法。

[C 4] 代替することが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記MERの左隣接ブロックで代替することを備え、結果的に前記MERの前記左隣接ブロックが前記現在のブロックの前記左下隣接ブロックと同じ垂直位置を有するようになる、C 2に記載の方法。

[C 5] 代替することが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記MERの右上隣接ブロックで代替することを備える、C 2に記載の方法。

[C 6] 代替することが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記MERの上隣接ブロックで代替することを備え、結果的に前記MERの前記上隣接ブロックが前記現在のブロックの前記右上隣接ブロックと同じ水平位置を有するようになる、C 2に記載の方法。

[C 7] 代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの左上隣接ブロックで代替することを備える、C 2に記載の方法。

[C 8] 代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの左隣接ブロックで代替することを備え、結果的に前記MERの前記左隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ垂直位置を有するようになる、C 2に記載の方法。

[C 9] 代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの上隣接ブロックで代替することを備え、結果的に前記MERの前記上隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ水平位置を有するようになる、C 2に記載の方法。

[C 10] 前記包含ブロックがコーディングユニット(CU)を備え、前記現在のブロックが予測ユニット(PU)を備える、C 1に記載の方法。

[C 11] 前記視差動きベクトルを使用してビデオデータの前記現在のブロックを復号することをさらに備える、C 1に記載の方法。

[C 12] 前記視差動きベクトルを復号することが、

動きベクトル差分値を復号することと、

前記視差動きベクトル予測子に前記動きベクトル差分値を追加して前記視差動きベクトルを再生することと
を備える、C 1に記載の方法。

[C 13] ビデオデータを復号するためのデバイスであって、包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断することと、前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替することと、前記候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号することとを行うように構成されたビデオデコーダを備えるデバイス。

[C 14] 前記包含ブロックが動き推定領域(MER)を備え、前記現在のブロックがコーディングユニット(CU)および予測ユニット(PU)のうちの1つを備える、C 13に記載のデバイス。

[C 15] 前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記MERの左下隣接ブロックで代替するように構成される、C 14に記載のデバイス。

[C 16] 前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記MERの左

隣接ブロックで代替するように構成され、結果的に前記MERの前記左隣接ブロックが前記現在のブロックの前記左下隣接ブロックと同じ垂直位置を有するようになる、C14に記載のデバイス。

[C17] 前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記MERの右上隣接ブロックで代替するように構成される、C14に記載のデバイス。

[C18] 前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記MERの上隣接ブロックで代替するように構成され、結果的に前記MERの前記上隣接ブロックが前記現在のブロックの前記右上隣接ブロックと同じ水平位置を有するようになる、C14に記載のデバイス。

[C19] 前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの左上隣接ブロックで代替するように構成される、C14に記載のデバイス。

[C20] 前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの左隣接ブロックで代替するように構成され、結果的に前記MERの前記左隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ垂直位置を有するようになる、C14に記載のデバイス。

[C21] 前記ビデオデコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの上隣接ブロックで代替するように構成され、結果的に前記MERの前記上隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ水平位置を有するようになる、C14に記載のデバイス。

[C22] 前記包含ブロックがコーディングユニット(CU)を備え、前記現在のブロックが予測ユニット(PU)を備える、C13に記載のデバイス。

[C23] 前記ビデオデコーダが、前記視差動きベクトルを使用してビデオデータの前記現在のブロックを復号するようにさらに構成される、C13に記載のデバイス。

[C24] 前記ビデオデコーダが、動きベクトル差分値を復号することと、前記視差動きベクトル予測子に前記動きベクトル差分値を追加して前記視差動きベクトルを再生することを行うように構成される、C13に記載のデバイス。

[C25] 前記デバイスが、

集積回路、

マイクロプロセッサ、および

ワイヤレス通信デバイス

のうちの少なくとも1つを備える、C13に記載のデバイス。

[C26] ビデオデータを復号するためのデバイスであって、

包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断するための手段と、

前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替するための手段と、

前記候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択するための手段と、

前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号するための手段とを備えるデバイス。

[C27] 前記包含ブロックが動き推定領域(MER)を備え、前記現在のブロックがコーディングユニット(CU)および予測ユニット(PU)のうちの1つを備える、C26に記載のデバイス。

[C28] 実行されたとき、1つまたは複数のプロセッサに、

包含ブロック内にあるビデオデータの現在のブロックが視差動きベクトルを使用して符号化されていると判断することと、

前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替することと、

前記候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、

前記視差動きベクトル予測子に基づいて前記視差動きベクトルを復号することとを行わせる命令を記憶したコンピュータ可読記憶媒体。

[C 2 9] 前記包含ブロックが動き推定領域 (M E R) を備え、前記現在のブロックがコーディングユニット (C U) および予測ユニット (P U) のうちの1つを備える、C 2 8 に記載のコンピュータ可読記憶媒体。

[C 3 0] ビデオデータを符号化する方法であって、

視差動きベクトルを使用して、包含ブロック内のビデオデータの現在のブロックを符号化することと、

前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替することと、

前記候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、

前記視差動きベクトル予測子に基づいて前記視差動きベクトルを符号化することとを備える方法。

[C 3 1] 前記包含ブロックが動き推定領域 (M E R) を備え、前記現在のブロックがコーディングユニット (C U) および予測ユニット (P U) のうちの1つを備える、C 3 0 に記載の方法。

[C 3 2] 代替することが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記 M E R の左下隣接ブロックで代替することを備える、C 3 1 に記載の方法。

[C 3 3] 代替することが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記 M E R の左隣接ブロックで代替することを備え、結果的に前記 M E R の前記左隣接ブロックが前記現在のブロックの前記左下隣接ブロックと同じ垂直位置を有するようになる、C 3 1 に記載の方法

。

[C 3 4] 代替することが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記 M E R の右上隣接ブロックで代替することを備える、C 3 1 に記載の方法。

[C 3 5] 代替することが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記 M E R の上隣接ブロックで代替することを備え、結果的に前記 M E R の前記上隣接ブロックが前記現在のブロックの前記右上隣接ブロックと同じ水平位置を有するようになる、C 3 1 に記載の方法

。

[C 3 6] 代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記 M E R の左上隣接ブロックで代替することを備える、C 3 1 に記載の方法。

[C 3 7] 代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記 M E R の左隣接ブロックで代替することを備え、結果的に前記 M E R の前記左隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ垂直位置を有するようになる、C 3 1 に記載の方法

。

[C 3 8] 代替することが、前記隣接ブロックが前記現在のブロックの左上隣接ブロック

に対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの上隣接ブロックで代替することを備え、結果的に前記MERの前記上隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ水平位置を有するようになる、C31に記載の方法。

[C39] 前記包含ブロックがコーディングユニット(CU)を備え、前記現在のブロックが予測ユニット(PU)を備える、C30に記載の方法。

[C40] 前記視差動きベクトルを符号化することが、
前記視差動きベクトルと前記視差動きベクトル予測子との間の差分を計算することと、
前記差分を符号化することと
を備える、C30に記載の方法。

[C41] ビデオデータを符号化するためのデバイスであって、視差動きベクトルを使用して、包含ブロック内のビデオデータの現在のブロックを符号化することと、前記現在のブロックの隣接ブロックも前記包含ブロック内にあるとの判断に基づいて、候補リスト中で前記隣接ブロックを、前記包含ブロックの外にあって前記包含ブロックに隣接するブロックで代替することと、前記候補リスト中の複数のブロックのうちの1つから視差動きベクトル予測子を選択することと、前記視差動きベクトル予測子に基づいて前記視差動きベクトルを符号化することとを行うように構成されたビデオエンコーダを備えるデバイス。

[C42] 前記包含ブロックが動き推定領域(MER)を備え、前記現在のブロックがコーディングユニット(CU)および予測ユニット(PU)のうちの1つを備える、C41に記載のデバイス。

[C43] 前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記MERの左下隣接ブロックで代替するように構成される、C42に記載のデバイス。

[C44] 前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの左下隣接ブロックに対応するときに、前記現在のブロックの前記左下隣接ブロックを前記MERの左隣接ブロックで代替するように構成され、結果的に前記MERの前記左隣接ブロックが前記現在のブロックの前記左下隣接ブロックと同じ垂直位置を有するようになる、C42に記載のデバイス。

[C45] 前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記MERの右上隣接ブロックで代替するように構成される、C42に記載のデバイス。

[C46] 前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの右上隣接ブロックに対応するときに、前記現在のブロックの前記右上隣接ブロックを前記MERの上隣接ブロックで代替するように構成され、結果的に前記MERの前記上隣接ブロックが前記現在のブロックの前記右上隣接ブロックと同じ水平位置を有するようになる、C42に記載のデバイス。

[C47] 前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの左上隣接ブロックで代替するように構成される、C42に記載のデバイス。

[C48] 前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの左隣接ブロックで代替するように構成され、結果的に前記MERの前記左隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ垂直位置を有するようになる、C42に記載のデバイス。

[C49] 前記ビデオエンコーダが、前記隣接ブロックが前記現在のブロックの左上隣接ブロックに対応するときに、前記現在のブロックの前記左上隣接ブロックを前記MERの上隣接ブロックで代替するように構成され、結果的に前記MERの前記上隣接ブロックが前記現在のブロックの前記左上隣接ブロックと同じ水平位置を有するようになる、C42に記載のデバイス。

[C50] 集積回路、

マイクロプロセッサ、および
ワイヤレス通信デバイス
のうちの少なくとも１つを備える、Ｃ４１に記載のデバイス。

【 国際調査報告 】

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2013/037442

A. CLASSIFICATION OF SUBJECT MATTER

INV. H04N7/36 H04N7/32 H04N7/26
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, COMPENDEX, INSPEC, 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	<p>JEON Y ET AL: "AHG10: Unified design on parallel merge/skip", 8. JCT-VC MEETING; 99. MPEG MEETING; 1-2-2012 - 10-2-2012; SAN JOSE; (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-H0090, 30 January 2012 (2012-01-30), XP030111117, figure 1 section 2. "Algorithm description" ----- -/--</p>	1-50

☒ 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

19 July 2013

Date of mailing of the international search report

26/07/2013

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

Stoufs, Maryse

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2013/037442

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	<p>JEON Y ET AL: "Non-CE9: improvement on parallelized merge/skip mode", 7. JCT-VC MEETING; 98. MPEG MEETING; 21-11-2011 - 30-11-2011; 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-G164, 8 November 2011 (2011-11-08), XP030110148, abstract figure 1(b) section 2 "Proposed method" -----</p>	1-50
X	<p>KIM (ETRI) H Y ET AL: "CU-based Merge Candidate List Construction", 7. JCT-VC MEETING; 98. MPEG MEETING; 21-11-2011 - 30-11-2011; 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-G416, 8 November 2011 (2011-11-08), XP030110400, section 2. "CU-based Merge Candidate List Construction" -----</p>	1-50
X	<p>YU Q ET AL: "Parallel AMVP candidate list construction", 9. JCT-VC MEETING; 100. MPEG MEETING; 27-4-2012 - 7-5-2012; 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-I0036, 17 April 2012 (2012-04-17), XP030111799, the whole document -----</p>	1-50
A	<p>JUNGHAK NAM ET AL: "Advanced motion and disparity prediction for 3D video coding", 98. MPEG MEETING; 28-11-2011 - 2-12-2011; GENEVA; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11),, no. m22560, 23 November 2011 (2011-11-23), XP030051123, section 2. "Proposed advanced motion and disparity prediction method" -----</p>	1-50

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, 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, 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, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, 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, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC

(74)代理人 100153051
弁理士 河野 直樹

(74)代理人 100140176
弁理士 砂川 克

(74)代理人 100158805
弁理士 井関 守三

(74)代理人 100179062
弁理士 井上 正

(74)代理人 100124394
弁理士 佐藤 立志

(74)代理人 100112807
弁理士 岡田 貴志

(74)代理人 100111073
弁理士 堀内 美保子

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

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

(72)発明者 カークゼウィックス、マルタ
アメリカ合衆国、カリフォルニア州 9 2 1 2 1 - 1 7 1 4、サン・ディエゴ、モアハウス・ドライブ 5 7 7 5

F ターム(参考) 5C159 LC09 MA00 MA04 MA05 MA16 MA21 MC11 MC38 ME01 NN10
NN11 NN21 NN28 PP03 PP05 PP06 PP07 PP13 SS26 UA02
UA05 UA33