

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6337025号
(P6337025)

(45) 発行日 平成30年6月6日 (2018.6.6)

(24) 登録日 平成30年5月11日 (2018.5.11)

(51) Int.Cl.	F I
HO 4 N 19/52 (2014.01)	HO 4 N 19/52
HO 4 N 19/597 (2014.01)	HO 4 N 19/597
HO 4 N 13/10 (2018.01)	HO 4 N 13/00 4 8 0

請求項の数 29 (全 54 頁)

(21) 出願番号	特願2015-559301 (P2015-559301)	(73) 特許権者	595020643
(86) (22) 出願日	平成26年2月26日 (2014.2.26)		クゥアルコム・インコーポレイテッド
(65) 公表番号	特表2016-511604 (P2016-511604A)		QUALCOMM INCORPORATED
(43) 公表日	平成28年4月14日 (2016.4.14)		ED
(86) 国際出願番号	PCT/US2014/018694		アメリカ合衆国、カリフォルニア州 92
(87) 国際公開番号	W02014/134174		121-1714、サン・ディエゴ、モア
(87) 国際公開日	平成26年9月4日 (2014.9.4)		ハウス・ドライブ 5775
審査請求日	平成29年1月27日 (2017.1.27)	(74) 代理人	100108855
(31) 優先権主張番号	61/769,716		弁理士 蔵田 昌俊
(32) 優先日	平成25年2月26日 (2013.2.26)	(74) 代理人	100109830
(33) 優先権主張国	米国 (US)		弁理士 福原 淑弘
(31) 優先権主張番号	61/770,263	(74) 代理人	100158805
(32) 優先日	平成25年2月27日 (2013.2.27)		弁理士 井関 守三
(33) 優先権主張国	米国 (US)	(74) 代理人	100194814
			弁理士 奥村 元宏

最終頁に続く

(54) 【発明の名称】 3Dビデオコード化における隣接ブロック視差ベクトル導出

(57) 【特許請求の範囲】

【請求項1】

ビデオデータをコード化する方法であって、
テクスチャ優先コード化を使用して、ビデオデータをコード化することと、
前記ビデオデータのブロックのための視差ベクトルを導出するために、複数の隣接ブロックを使用して、前記ビデオデータの前記ブロックに対して、隣接ブロックベース視差ベクトル (NB DV) 導出プロセスを実行することと
を備え、前記実行することは、
前記複数の隣接ブロックのいずれかの隣接ブロックがビュー間予測を使用してコード化されているかどうかを決定するために、又は前記複数の隣接ブロックのいずれかの隣接ブロックがブロックベースビュー合成予測 (BVSP) モードを使用してコード化されているかどうかを決定するために、前記複数の隣接ブロックをチェックすることと、
前記チェックすることに基づいて、
前記複数の隣接ブロックのうち特定の隣接ブロックが前記 BVSP モードを使用してコード化されていると決定することに基づいて、前記 BVSP モードでコード化された前記複数の隣接ブロックのうち前記特定の隣接ブロックに関連付けられた動きベクトルを、1つ以上の利用可能な視差動きベクトルのうち利用可能な視差動きベクトルとして指定すること、又は、
前記複数の隣接ブロックのうち前記特定の隣接ブロックが前記ビュー間予測を使用してコード化されていると決定することに基づいて、前記ビュー間予測モードを使用し

10

20

てコード化された前記複数の隣接ブロックのうちの前記特定の隣接ブロックに関連付けられた前記動きベクトルを前記 1 つ以上の利用可能な視差動きベクトルのうちの前記利用可能な視差動きベクトルとして指定すること、

のうちの 1 つを実行することと、

前記特定の隣接ブロックが前記 B V S P モードを使用してコード化されていることに
基づいて、又は、前記特定の隣接ブロックが前記ビュー間予測モードを使用してコード化
されていることに基づいて、前記ビデオデータの前記ブロックのための前記視差ベクトル
を導出することと

によって、実行することである、

方法。

10

【請求項 2】

前記ビデオデータの前記ブロックのための前記視差ベクトルを導出することが、

前記 1 つ以上の利用可能な視差動きベクトルから、前記ビデオデータの前記ブロックの
ための前記視差ベクトルを導出することを備える、請求項 1 に記載の方法。

【請求項 3】

前記複数の隣接ブロックをチェックすることが、

前記複数の隣接ブロックのいずれかが前記ビュー間予測を使用してコード化されている
かどうかを決定するために、前記複数の隣接ブロックをチェックすることと、

前記複数の隣接ブロックのいずれも前記ビュー間予測を使用してコード化されていない
場合のみ、前記複数の隣接ブロックのいずれかが前記 B V S P モードを使用してコード化
されているかどうかを決定するために、前記複数の隣接ブロックをチェックすることと
を備える、請求項 1 に記載の方法。

20

【請求項 4】

前記複数の隣接ブロックをチェックすることが、

前記複数の隣接ブロックのいずれかが前記 B V S P モードを使用してコード化されてい
るかどうかを決定するために、前記複数の隣接ブロックをチェックすることと、

前記複数の隣接ブロックのいずれも前記 B V S P モードを使用してコード化されてい
ない場合のみ、前記複数の隣接ブロックのいずれかが前記ビュー間予測を使用してコード化
されているかどうかを決定するために、前記複数の隣接ブロックをチェックすることと
を備える、請求項 1 に記載の方法。

30

【請求項 5】

前記 N B D V 導出プロセスを実行することが、

深度参照ビュー中の深度ブロックから 1 つの深度値を選択することと、

前記深度値を更新された視差ベクトルに変換することと、

前記更新された視差ベクトルを前記ビデオデータの前記ブロックに適用することと

を更に備える、請求項 1 に記載の方法。

【請求項 6】

前記ビデオデータの前記ブロックがコード化された後、前記ビデオデータの
前記ブロックのための動きベクトルとして、前記更新された視差ベクトルを記憶することを更に備え
る、請求項 5 に記載の方法。

40

【請求項 7】

前記導出された視差ベクトルを記憶するために、追加のメモリを割り振ることを更に備
える、請求項 1 に記載の方法。

【請求項 8】

前記ビデオデータの前記ブロックが、マクロブロックである、請求項 1 に記載の方法。

【請求項 9】

前記ビデオデータの前記ブロックが、マクロブロックの区分又は前記区分のサブブロッ
クである、請求項 1 に記載の方法。

【請求項 10】

前記ビデオデータの前記ブロックが、コード化単位又は予測単位である、請求項 1 に記

50

載の方法。

【請求項 1 1】

前記 B V S P モードと前記導出された視差ベクトルとを使用して、前記ビデオデータの前記ブロックをコード化することを更に備える、請求項 1 に記載の方法。

【請求項 1 2】

深度優先コード化順序に起因して利用可能である、現在のビュー中の関連付けられた深度マップデータを組み込む動きベクトル予測方法を指す深度ベース動きベクトル予測 (D - M V P) と前記導出された視差ベクトルとを使用して、前記ビデオデータの前記ブロックをコード化することを更に備える、請求項 2 に記載の方法。

【請求項 1 3】

ビデオデータをコード化するように構成された装置であって、前記装置は、
前記ビデオデータのブロックを記憶するように構成されたメモリと、
前記メモリと通信するビデオコードと
を備え、前記ビデオコードは、
テクスチャ優先コード化を使用して、ビデオデータをコード化することと、
前記ビデオデータの前記ブロックのための視差ベクトルを導出するために、複数の隣接ブロックを使用して、前記ビデオデータの前記ブロックに対して、隣接ブロックベース視差ベクトル (N B D V) 導出プロセスを実行することと
を行うように構成され、ここにおいて、前記 N B D V 導出プロセスを実行するために、前記ビデオコードは、

前記複数の隣接ブロックのいずれかの隣接ブロックがビュー間予測を使用してコード化されているかどうかを決定するために、又は前記複数の隣接ブロックのいずれかの隣接ブロックがブロックベースビュー合成予測 (B V S P) モードを使用してコード化されているかどうかを決定するために、前記複数の隣接ブロックをチェックすることと、

前記複数の隣接ブロックの前記チェックからの、前記複数の隣接ブロックのうちの特定の隣接ブロックが前記 B V S P モードを使用してコード化されているという決定に基づいて、前記 B V S P モードでコード化された前記複数の隣接ブロックのうちの前記特定の隣接ブロックに関連付けられた動きベクトルを、1 つ以上の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして指定することと、

前記複数の隣接ブロックの前記チェックからの、前記複数の隣接ブロックのうちの前記特定の隣接ブロックが前記ビュー間予測を使用してコード化されているという決定に基づいて、前記ビュー間予測モードを使用してコード化された前記複数の隣接ブロックのうちの前記特定の隣接ブロックに関連付けられた動きベクトルを、前記 1 つ以上の利用可能な視差動きベクトルのうちの前記利用可能な視差動きベクトルとして指定することと、

前記特定の隣接ブロックが前記 B V S P モードを使用してコード化されていることに基づいて、又は、前記特定の隣接ブロックが前記ビュー間予測モードを使用してコード化されていることに基づいて、前記ビデオデータの前記ブロックのための前記視差ベクトルを導出することと

を行うように構成される、

装置。

【請求項 1 4】

前記ビデオデータの前記ブロックのための前記視差ベクトルを導出するために、前記ビデオコードが、

前記 1 つ以上の利用可能な視差動きベクトルから、前記ビデオデータの前記ブロックのための前記視差ベクトルを導出することを行うように更に構成される、請求項 1 3 に記載の装置。

【請求項 1 5】

前記複数の隣接ブロックをチェックするために、前記ビデオコードが、
前記複数の隣接ブロックのいずれかが前記ビュー間予測を使用してコード化されているかどうかを決定するために、前記複数の隣接ブロックをチェックすることと、

前記複数の隣接ブロックのいずれも前記ビュー間予測を使用してコード化されていない場合のみ、前記複数の隣接ブロックのいずれかが前記 B V S P モードを使用してコード化されているかどうかを決定するために、前記複数の隣接ブロックをチェックすることとを行うように更に構成される、請求項 1 3 に記載の装置。

【請求項 1 6】

前記複数の隣接ブロックをチェックするために、前記ビデオコードが、
前記複数の隣接ブロックのいずれかが前記 B V S P モードを使用してコード化されているかどうかを決定するために、前記複数の隣接ブロックをチェックすることと、
前記複数の隣接ブロックのいずれも前記 B V S P モードを使用してコード化されていない場合のみ、前記複数の隣接ブロックのいずれかが前記ビュー間予測を使用してコード化されているかどうかを決定するために、前記複数の隣接ブロックをチェックすることとを行うように更に構成される、請求項 1 3 に記載の装置。

10

【請求項 1 7】

前記ビデオコードが、
深度参照ビュー中の深度ブロックから 1 つの深度値を選択し、前記深度値を更新された視差ベクトルに変換することと、
前記深度値を更新された視差ベクトルに変換することと、
前記更新された視差ベクトルを前記ビデオデータの前記ブロックに適用することとを行うように更に構成される、請求項 1 3 に記載の装置。

【請求項 1 8】

前記ビデオコードが、
前記ビデオデータの前記ブロックがコード化された後、前記ビデオデータの前記ブロックのための動きベクトルとして、前記更新された視差ベクトルを、前記メモリに記憶することを行うように更に構成される、請求項 1 7 に記載の装置。

20

【請求項 1 9】

前記ビデオコードが、
前記導出された視差ベクトルを記憶するために、追加のメモリを割り振ることを行うように更に構成される、請求項 1 3 に記載の装置。

【請求項 2 0】

前記ビデオデータの前記ブロックが、マクロブロックである、請求項 1 3 に記載の装置。

30

【請求項 2 1】

前記ビデオデータの前記ブロックが、マクロブロックの区分又は前記区分のサブブロックである、請求項 1 3 に記載の装置。

【請求項 2 2】

前記ビデオデータの前記ブロックが、コード化単位又は予測単位である、請求項 1 3 に記載の装置。

【請求項 2 3】

前記ビデオコードが、
B V S P モードと前記導出された視差ベクトルとを使用して、前記ビデオデータの前記ブロックをコード化することを行うように更に構成される、請求項 1 3 に記載の装置。

40

【請求項 2 4】

前記ビデオコードが、
深度優先コード化順序に起因して利用可能である、現在のビュー中の関連付けられた深度マップデータを組み込む動きベクトル予測方法を指す深度ベース動きベクトル予測 (D - M V P)と前記導出された視差ベクトルとを使用して、前記ビデオデータの前記ブロックをコード化することを行うように更に構成される、請求項 1 4 に記載の装置。

【請求項 2 5】

実行されたとき、ビデオデータをコード化するように構成された 1 つ以上のプロセッサに、

50

テクスチャ優先コード化を使用して、ビデオデータをコード化することと、
前記ビデオデータの前記ブロックのための視差ベクトルを導出するために、複数の隣接ブロックを使用して、前記ビデオデータの前記ブロックに対して、隣接ブロックベース視差ベクトル（NB DV）導出プロセスを実行することと、

を行わせる命令を記憶する、非一時的コンピュータ可読記憶媒体であって、前記1つ以上のプロセッサに前記NB DV導出プロセスを実行させる前記命令は、実行されたとき、前記1つ以上のプロセッサに、

前記複数の隣接ブロックのいずれかの隣接ブロックがビュー間予測を使用してコード化されているかどうかを決定するために、又は前記複数の隣接ブロックのいずれかの隣接ブロックがブロックベースビュー合成予測（BVSP）モードを使用してコード化されているかどうかを決定するために、ある順序に従って前記複数の隣接ブロックをチェックすることと、

前記複数の隣接ブロックの前記チェックに基づいて、

前記複数の隣接ブロックのうちの特定の隣接ブロックが前記BVSPモードを使用してコード化されているという決定に基づいて、前記BVSPモードでコード化された前記複数の隣接ブロックのうちの前記特定の隣接ブロックに関連付けられた動きベクトルを、1つ以上の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして指定することと、

前記複数の隣接ブロックのうちの前記特定の隣接ブロックが前記ビュー間予測を使用してコード化されているという決定に基づいて、前記ビュー間予測モードを使用してコード化された前記複数の隣接ブロックのうちの前記特定の隣接ブロックに関連付けられた動きベクトルを、前記1つ以上の利用可能な視差動きベクトルのうちの前記利用可能な視差動きベクトルとして指定することと

のうちの1つを実行することと、

前記特定の隣接ブロックが前記BVSPモードを使用してコード化されていることに基づいて、又は、前記特定の隣接ブロックが前記ビュー間予測モードを使用してコード化されていることに基づいて、前記ビデオデータの前記ブロックのための前記視差ベクトルを導出することと

を行わせる命令を備える、非一時的コンピュータ可読記憶媒体。

【請求項26】

前記1つ以上のプロセッサに、前記ビデオデータの前記ブロックのための前記視差ベクトルを導出することを行わせる前記命令が、前記1つ以上のプロセッサに、

前記1つ以上の利用可能な視差動きベクトルから、前記ビデオデータの前記ブロックのための前記視差ベクトルを導出することを行わせる、請求項25に記載の非一時的コンピュータ可読記憶媒体。

【請求項27】

1つ以上の集積回路、

1つ以上のマイクロプロセッサ、

1つ以上のデジタル信号プロセッサ（DSP）、

1つ以上のフィールドプログラマブルゲートアレイ（FPGA）、

デスクトップコンピュータ、

ラップトップコンピュータ、

タブレットコンピュータ、

電話

テレビジョン、

カメラ、

表示装置、

デジタルメディアプレーヤ、

ビデオゲームコンソール、

ビデオゲーム機器、

10

20

30

40

50

ビデオストリーミング機器、又は、
ワイヤレス通信機器

のうちの少なくとも１つを更に備える、請求項１３に記載の機器。

【請求項２８】

前記ビデオデータの少なくとも一部分を撮影するように構成されたカメラを更に備える、請求項１３に記載の機器。

【請求項２９】

前記ビデオデータの少なくとも一部分を出力するように構成された表示器を更に備える、請求項１３に記載の機器。

【発明の詳細な説明】

10

【技術分野】

【０００１】

[0001]本出願は、各々の内容がその全体として本明細書に組み込まれる、
2013年2月26日に提出された米国仮出願第61/769,716号、
2013年2月27日に提出された米国仮出願第61/770,263号、
2013年2月27日に提出された米国仮出願第61/770,268号、
2013年3月4日に提出された米国仮出願第61/772,321号、
2013年3月19日に提出された米国仮出願第61/803,384号、及び
2013年4月24日に提出された米国仮出願第61/815,656号の利益を主張する。

20

【０００２】

[0002]本開示は、ビデオ符号化及び復号に関する。

【背景技術】

【０００３】

[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)、ラップトップ又はデスクトップコンピュータ、デジタルカメラ、デジタル記録機器、デジタルメディアプレーヤ、ビデオゲーム機器、ビデオゲームコンソール、携帯電話又は衛星無線電話、ビデオ遠隔会議機器などを含む、広範囲にわたる機器に組み込まれ得る。デジタルビデオ機器は、デジタルビデオ情報をより効率的に送信、受信及び記憶するための、MP

30

EG-2、MP EG-4、ITU-T H.263、ITU-T H.264/MP EG-4、Part 10、アドバンスドビデオコード化(AVC: Advanced Video Coding)によって定義された規格、現在開発中の高効率ビデオコード化(HEVC: High Efficiency Video Coding)規格、及びそのような規格の拡張に記載されているビデオ圧縮技法など、ビデオ圧縮技法を実装する。

[0004]H.264/AVCを含む前述の規格のうちの幾つかの拡張は、ステレオ又は3次元(「3D」)ビデオを生成するためにマルチビュービデオコード化用の技法を提供する。具体的には、マルチビューコード化用の技法は、(H.264/AVCに対するスケーラブル拡張である)スケーラブルビデオコード化(SVC)規格、及び(H.264/AVCに対するマルチビュー拡張になった)マルチビュービデオコード化(MVC)規格とともに、AVCにおいて使用するために提案されている。

40

【０００５】

[0005]一般に、ステレオビデオは、２つのビュー、例えば、左ビューと右ビューとを使用して実現される。左ビューのピクチャは右ビューのピクチャと実質的に同時に表示されて、3次元ビデオ効果が実現され得る。例えば、ユーザは、左ビューを右ビューからフィルタ処理する偏光パッシブ眼鏡を着用し得る。代替的に、２つのビューのピクチャは高速で連続して示され得、ユーザは、同じ頻度で、但し位相が90度シフトして左眼と右眼とを迅速に閉じるアクティブ眼鏡を着用し得る。

【発明の概要】

50

【 0 0 0 6 】

[0006]概して、本開示では、3Dビデオコード化用の技法について説明する。詳細には、本開示は、3Dビデオコード化における、隣接ブロック視差ベクトル(NBDV: neighboring block disparity vector)導出とブロックベースビュー合成予測(BVSP: block-based view synthesis prediction)とに関する。

【 0 0 0 7 】

[0007]本開示の一例では、ビデオデータをコード化する方法は、テクスチャ優先(texture-first)コード化を使用して、ビデオデータをコード化することと、複数の隣接ブロックを使用して、ビデオデータのブロックに対して、NBDV導出プロセスを実行することと、ここにおいて、NBDV導出プロセスは視差ベクトルを導出する、を備え、ここにおいて、NBDVプロセスを実行することは、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ブロックベースビュー合成予測(BVSP)モードでコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ビュー間予測モードを使用してコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することとを備える。

10

【 0 0 0 8 】

[0008]本開示の別の例では、ビデオデータをコード化するように構成された装置は、ビデオデータのブロックを記憶するように構成されたメモリと、テクスチャ優先コード化を使用して、ビデオデータをコード化することと、複数の隣接ブロックを使用して、ビデオデータのブロックに対して、隣接ブロックベース視差ベクトル(NBDV: neighboring block-based disparity vector)導出プロセスを実行することと、ここにおいて、NBDV導出プロセスは視差ベクトルを導出する、を行うように構成されたビデオコード化装置とを備え、ここにおいて、NBDVプロセスを実行することは、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ブロックベースビュー合成予測(BVSP)モードでコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ビュー間予測モードを使用してコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することとを備える。

20

30

【 0 0 0 9 】

[0009]本開示の別の例では、ビデオデータをコード化するように構成された装置は、テクスチャ優先コード化を使用して、ビデオデータをコード化するための手段と、複数の隣接ブロックを使用して、ビデオデータのブロックに対して、隣接ブロックベース視差ベクトル(NBDV)導出プロセスを実行することと、ここにおいて、NBDV導出プロセスは視差ベクトルを導出する、を行うための手段とを備え、ここにおいて、NBDVプロセスを実行するための手段は、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ブロックベースビュー合成予測(BVSP)モードでコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定するための手段と、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ビュー間予測モードを使用してコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定するための手段とを備える。

40

【 0 0 1 0 】

[0010]別の例では、本開示は、実行されたとき、ビデオデータをコード化するように構成された1つ又は複数のプロセッサに、テクスチャ優先コード化を使用して、ビデオデータをコード化することと、複数の隣接ブロックを使用して、ビデオデータのブロックに対して、隣接ブロックベース視差ベクトル(NBDV)導出プロセスを実行することと、ここにおいて、NBDV導出プロセスは視差ベクトルを導出する、を行わせる命令を記憶する、コンピュータ可読記憶媒体について説明し、ここにおいて、NBDVプロセスを実行

50

することは、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ブロックベースビュー合成予測 (BVS P) モードでコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ビュー間予測モードを使用してコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することとを備える。

【0011】

[0011]本開示の別の例では、ビデオデータをコード化する方法は、ビデオデータのブロック上でブロックベースビュー合成予測 (BVS P) プロセスを実行することを備え、BVS P プロセスは、視差ベクトルを導出するために、NB D V 導出プロセスを実行することと、ビデオデータのブロックの下位領域 (sub-region) について、導出視差ベクトルを精密化する (refining) ことと、BVS P を使用して、ビデオデータのブロックをコード化することとを備え、ここにおいて、NB D V 導出プロセスを実行することは、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ブロックベースビュー合成予測 (BVS P) モードでコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ビュー間予測モードを使用してコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、1つ又は複数の利用可能な視差動きベクトルから、視差ベクトルを導出することとを備える。

【0012】

[0012]本開示の別の例では、ビデオデータをコード化するように構成された装置は、ビデオデータのブロックを記憶するように構成されたメモリと、ビデオデータのブロック上でブロックベースビュー合成予測 (BVS P) プロセスを実行することを行うように構成されたビデオコーダとを備え、BVS P プロセスは、視差ベクトルを導出するために、NB D V 導出プロセスを実行することと、ビデオデータのブロックの下位領域について、導出視差ベクトルを精密化することと、BVS P を使用して、ビデオデータのブロックをコード化することとを備え、ここにおいて、NB D V 導出プロセスを実行することは、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ブロックベースビュー合成予測 (BVS P) モードでコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ビュー間予測モードを使用してコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、1つ又は複数の利用可能な視差動きベクトルから、視差ベクトルを導出することとを備える。

【0013】

[0013]本開示の別の例では、ビデオデータをコード化するように構成された装置は、ビデオデータのブロック上でブロックベースビュー合成予測 (BVS P) プロセスを実行するための手段を備え、BVS P プロセスを実行するための手段は、視差ベクトルを導出するために、NB D V 導出プロセスを実行するための手段と、ビデオデータのブロックの下位領域について、導出視差ベクトルを精密化するための手段と、BVS P を使用して、ビデオデータのブロックをコード化するための手段とを備え、ここにおいて、NB D V 導出プロセスを実行するための手段は、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ブロックベースビュー合成予測 (BVS P) モードでコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定するための手段と、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ビュー間予測モードを使用してコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定するための手段と、1つ又は複数の利用可能な視差動きベクトルから、視差ベクトルを導出するための手段とを備える。

【 0 0 1 4 】

[0014]別の例では、本開示は、実行されたとき、ビデオデータをコード化するように構成された1つ又は複数のプロセッサに、ビデオデータのブロック上でブロックベースビュー合成予測(BVSP)プロセスを実行することを行わせる命令を記憶する、コンピュータ可読記憶媒体について説明し、BVSPプロセスは、視差ベクトルを導出するために、NB DV導出プロセスを実行することと、ビデオデータのブロックの下位領域について、導出視差ベクトルを精密化することと、BVSPを使用して、ビデオデータのブロックをコード化することとを備え、ここにおいて、NB DV導出プロセスを実行することは、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ブロックベースビュー合成予測(BVSP)モードでコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ビュー間予測モードを使用してコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、1つ又は複数の利用可能な視差動きベクトルから、視差ベクトルを導出することとを備える。

10

【 0 0 1 5 】

[0015]1つ又は複数の例の詳細が、添付の図面及び以下の説明に記載されている。他の特徴、目的、及び利点は、その説明及び図面から、ならびに特許請求の範囲から明らかになる。

【図面の簡単な説明】

20

【 0 0 1 6 】

【図1】[0016]本開示で説明される技法を利用し得る例示的なビデオ符号化及び復号システムを示すブロック図。

【図2】[0017]例示的なマルチビュー復号順序を示す概念図。

【図3】[0018]マルチビューコード化のための例示的な予測構造を示す概念図。

【図4】[0019]後方ワーピングに基づくブロックベースビュー合成予測の概念の視覚化の図。

【図5】[0020]隣接ブロック視差ベクトル導出のために使用される空間隣接ブロックを示す概念図。

【図6】[0021]本開示で説明される技法を実装し得る例示的なビデオエンコーダを示すブロック図。

30

【図7】[0022]本開示で説明される技法を実装し得る例示的なビデオデコーダを示すブロック図。

【図8】[0023]深度ベース動きベクトル予測のための隣接ブロックを示す概念図。

【図9】[0024]本開示の例示的な方法を示すフローチャート。

【図10】[0025]本開示の別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】

【 0 0 1 7 】

[0026]概して、本開示は、(例えば、H.264/AVCの3D拡張では、3D-AVCと呼ばれることがある)H.264/アドバンストビデオコード化(AVC)コーデックを用いた2つ以上のビューのコード化を含む、アドバンストコーデックに基づくマルチビュープラス深度(例えば、3D)ビデオコード化のための技法について説明する。幾つかの例では、3D-AVCベースマルチビュービデオコード化における、ビュー合成予測と視差ベクトル導出とに関係する技法が提案される。但し、本開示の技法は、一般的に、高効率ビデオコード化(HEVC)規格のマルチビュー拡張及び新たに出現した3D拡張を含む、他のマルチビュー及び/又は3Dビデオコード化技法に適用可能であり得る。

40

【 0 0 1 8 】

[0027]テクスチャ優先復号を採用するとき、3D-AVCのための現在の提案には、正確な視差ベクトルを導出するための技法が欠けている。具体的には、隣接ブロック視差ベクトル(NB DV)導出プロセスのための現在の提案から導出された視差ベクトルは、不

50

正確な視差ベクトルを生成し得る。更に、テクスチャ優先コード化を採用するとき、ブロックベースビュー合成予測(BVSP)コード化のために、導出された視差ベクトルを使用するための現在の技法はない。

【0019】

[0028]これらの欠点に鑑みて、本開示は、非ベーステクスチャビューコンポーネントが、対応する非ベース深度ビューコンポーネントの前にコード化されるとき、3D-AVC準拠ビデオエンコーダ及びビデオデコーダのためのBVSPを可能にするための技法を提案する。加えて、他のインターコード化モードのコード化利得もまた、本開示の技法によって提供されるような、精密な視差ベクトルの導出により、改善される。

【0020】

[0029]図1は、本開示で説明されるビュー合成予測及び視差ベクトル導出のための技法を実行するように構成され得る、例示的なビデオ符号化及び復号システム10を示すブロック図である。図1に示されるように、システム10は、宛先機器14によって後の時間で復号されるべき符号化されたビデオデータを生成する発信源機器12を含む。発信源機器12及び宛先機器14は、デスクトップコンピュータ、ノートブック(即ち、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、所謂「スマートフォン」などの電話ハンドセット、所謂「スマート」パッド、テレビジョン、カメラ、表示装置、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機器などを含む、広範囲にわたる機器のいずれかを備え得る。場合によっては、発信源機器12及び宛先機器14は、ワイヤレス通信のために装備され得る。

【0021】

[0030]システム10は、異なるビデオコード化規格、プロプライエタリ規格、又はマルチビューコード化の任意の他の方法に従って動作し得る。下記は、ビデオコード化規格の数例について説明しており、限定と見なされるべきではない。ビデオコード化規格は、ITU-T H.261、ISO/IEC MPEG-1 Visual、ITU-T H.262又はISO/IEC MPEG-2 Visual、ITU-T H.263、ISO/IEC MPEG-4 Visual、及びそのスケラブルビデオコード化(SVC)拡張とマルチビュービデオコード化(MVC)拡張とを含む(ISO/IEC MPEG-4 AVCとしても知られる)ITU-T H.264を含む。MVCの最新のジョイントドラフトは、「Advanced video coding for generic audiovisual services」、ITU-T勧告H.264、2010年3月に記載されており、その内容全体は参照により本明細書に組み込まれる。MVCの別のジョイントドラフトは、「Advanced video coding for generic audiovisual services」、ITU-T勧告H.264、2011年6月に記載されており、その内容全体は参照により本明細書に組み込まれる。幾つかの追加のビデオコード化規格には、AVCに基づく、MVC+D及び3D-AVCがある。加えて、新しいビデオコード化規格、即ち高効率ビデオコード化(HEVC)が、ITU-T Video Coding Experts Group(VCEG)及びISO/IEC Motion Picture Experts Group(MPEG)のJoint Collaboration Team on Video Coding(JCT-VC)によって開発された。

【0022】

[0031]単に例示のために、本開示で説明される技法は、3D-AVCなど、H.264規格による例とともに説明される。しかしながら、本開示で説明される技法は、これらの例示的な規格に限定されると見なされるべきではなく、マルチビューコード化若しくは3Dビデオコード化のための他のビデオコード化規格(例えば、3D-HEVC)、又は必ずしも特定のビデオコード化規格に基づくとは限らないマルチビューコード化若しくは3Dビデオコード化に関連する技法に拡張可能であり得る。例えば、本開示で説明される技法は、マルチビューコード化のためのビデオエンコーダ/デコーダ(コーデック)によって実装され、ここでマルチビューコード化は、2つ以上のビューのコード化を含む。

【 0 0 2 3 】

[0032]宛先機器 1 4 は、リンク 1 6 を介して、復号されるべき符号化されたビデオデータを受信し得る。リンク 1 6 は、発信源機器 1 2 から宛先機器 1 4 に符号化されたビデオデータを移動することが可能な任意のタイプの媒体又は機器を備え得る。一例では、リンク 1 6 は、発信源機器 1 2 が、符号化されたビデオデータをリアルタイムで宛先機器 1 4 に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。符号化されたビデオデータは、ワイヤレス通信プロトコルなどの通信規格に従って変調され、宛先機器 1 4 に送信され得る。通信媒体は、高周波 (R F) スペクトル又は 1 つ以上の物理伝送線路など、任意のワイヤレス又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、又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、発信源機器 1 2 から宛先機器 1 4 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、又は任意の他の機器を含み得る。

10

【 0 0 2 4 】

[0033]代替的に、符号化されたデータは、出力インターフェース 2 2 から記憶装置 3 4 に出され得る。同様に、符号化されたデータは、入力インターフェースによって記憶装置 3 4 からアクセスされ得る。記憶装置 3 4 は、ハードドライブ、Blu-ray (登録商標) ディスク、DVD、CD-ROM、フラッシュメモリ、揮発性又は不揮発性メモリ、若しくは符号化された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、様々な分散した又はローカルでアクセスされるデータ記憶媒体のいずれかを含み得る。更なる一例では、記憶装置 3 4 は、発信源機器 1 2 によって生成された符号化されたビデオを保持できるファイルサーバ又は別の中間記憶装置に対応し得る。宛先機器 1 4 は、ストリーミング又はダウンロードを介して、記憶装置 3 4 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、符号化されたビデオデータを記憶し、その符号化されたビデオデータを宛先機器 1 4 に送信することが可能な任意のタイプのサーバとすることができる。例示的なファイルサーバとしては、(例えば、ウェブサイト用の)ウェブサーバ、FTPサーバ、ネットワーク接続ストレージ (NAS) 機器、又はローカルディスクドライブがある。宛先機器 1 4 は、インターネット接続を含む任意の標準的なデータ接続を通じて、符号化されたビデオデータにアクセスし得る。これは、ファイルサーバに記憶された符号化されたビデオデータにアクセスするのに適してい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)、又は両方の組合せを含み得る。記憶装置 3 4 からの符号化されたビデオデータの送信は、ストリーミング送信、ダウンロード送信、又は両方の組合せであり得る。

20

30

【 0 0 2 5 】

[0034]ビュー合成予測及び視差ベクトル導出のための本開示の技法は、必ずしもワイヤレス適用例又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、例えばインターネットを介したストリーミングビデオ送信、データ記憶媒体に記憶するためのデジタルビデオの符号化、データ記憶媒体に記憶されたデジタルビデオの復号、又は他の適用例のような、種々の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ド化に適用され得る。幾つかの例では、システム 1 0 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、及び / 又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、一方向又は双方向のビデオ送信をサポートするように構成され得る。

40

【 0 0 2 6 】

[0035]図 1 の例では、発信源機器 1 2 は、ビデオ発信源 1 8 と、ビデオエンコーダ 2 0 と、出力インターフェース 2 2 とを含む。場合によっては、出力インターフェース 2 2 は、変調器 / 復調器 (モデム) 及び / 又は送信機を含み得る。発信源機器 1 2 において、ビデオ発信源 1 8 は、撮像装置、例えばビデオカメラ、以前に撮影されたビデオを含んでいるビデオアーカイブ、ビデオコンテンツプロバイダからビデオを受信するためのビデオフ

50

ードインターフェース、及び／又は発信源ビデオとしてコンピュータグラフィックスデータを生成するためのコンピュータグラフィックスシステムのような発信源、又はそのような発信源の組合せを含み得る。一例として、ビデオ発信源１８がビデオカメラである場合、発信源機器１２及び宛先機器１４は、所謂カメラフォン又はビデオフォンを形成し得る。但し、本開示で説明される技法は、概してビデオコード化に適用可能であり得、ワイヤレス及び／又はワイヤード適用例に適用され得る。

【００２７】

[0036]撮影されたビデオ、プリ撮影されたビデオ、又はコンピュータにより生成されたビデオは、ビデオエンコーダ２０によって符号化され得る。符号化されたビデオデータは、発信源機器１２の出力インターフェース２２を介して宛先機器１４に直接送信され得る。符号化されたビデオデータは、更に（又は代替的に）、復号及び／又は再生のための宛先機器１４又は他の機器による後のアクセスのために記憶装置３４上に記憶され得る。

10

【００２８】

[0037]宛先機器１４は、入力インターフェース２８と、ビデオデコーダ３０と、表示装置３２とを含む。場合によっては、入力インターフェース２８は、受信機及び／又はモデムを含み得る。宛先機器１４の入力インターフェース２８は、リンク１６を介して、符号化されたビデオデータを受信する。リンク１６を介して通信され、又は記憶装置３４上に提供された符号化されたビデオデータは、ビデオデータを復号する際にビデオデコーダ３０などのビデオデコーダが使用するのための、ビデオエンコーダ２０によって生成された様々なシンタックス要素を含み得る。そのようなシンタックス要素は、通信媒体上で送信され、記憶媒体上に記憶される符号化されたビデオデータとともに含まれ得、又はファイルサーバを記憶した。

20

【００２９】

[0038]表示装置３２は、宛先機器１４と一体であってよく、又はその外部にあり得る。幾つかの例では、宛先機器１４は、集積表示装置を含むことができ、また、外部表示装置とインターフェースするように構成され得る。他の例では、宛先機器１４は表示装置であり得る。一般に、表示装置３２は、復号されたビデオデータをユーザに対して表示し、液晶表示器（ＬＣＤ）、プラズマ表示器、有機発光ダイオード（ＯＬＥＤ）表示器、又は別のタイプの表示装置など、種々の表示装置のいずれかを備え得る。

【００３０】

30

[0039]図１には示されないが、幾つかの態様では、ビデオエンコーダ２０及びビデオデコーダ３０は、それぞれオーディオエンコーダ及びオーディオデコーダと統合されてよく、共通のデータストリーム又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するための、適切なＭＵＸ－ＤＥＭＵＸユニット、又は他のハードウェアとソフトウェアとを含み得る。適用可能な場合、幾つかの例では、ＭＵＸ－ＤＥＭＵＸユニットは、ＩＴＵ－Ｈ．２２３マルチプレクサプロトコル、又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。

【００３１】

[0040]ビデオエンコーダ２０及びビデオデコーダ３０はそれぞれ、１つ又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、ファームウェアなど、様々な好適なエンコーダ回路のいずれか、あるいはそれらの任意の組合せとして実装され得る。例えば、本開示で説明される技法は、装置又は機器の観点から説明され得る。一例として、装置又は機器は、ビデオデコーダ３０（例えば、ワイヤレス通信機器の一部としての宛先機器１４）を含んでよく、ビデオデコーダ３０は、本開示で説明される技法を実装する（例えば、本開示で説明される技法に従って、ビデオデータを復号する）ように構成された１つ又は複数のプロセッサを含んでよい。別の例として、装置又は機器は、ビデオデコーダ３０を含むマイクロプロセッサ又は集積回路（ＩＣ）を含んでよく、マイクロプロセッサ又はＩＣは、宛先機器１４又は別のタイプの機器の一部であり得る。同じことは、ビデオエンコーダ２０にも当てはまり

40

50

得る（即ち、発信源機器 12 のような装置又は機器、及び / 又はマイクロコントローラ若しくは IC は、ビデオエンコーダ 20 を含み、その場合、ビデオエンコーダ 20 は、本開示で説明される技法に従ってビデオデータを符号化するように構成される）。

【0032】

[0041] 本技法が部分的にソフトウェアで実装されるとき、機器は、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、本開示の技法を実行するために 1 つ又は複数のプロセッサを使用してハードウェアでその命令を実行し得る。ビデオエンコーダ 20 及びビデオデコーダ 30 の各々は 1 つ又は複数のエンコーダ又はデコーダ中に含まれてよく、そのいずれも、それぞれの機器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てよい。

10

【0033】

[0042] ビデオシーケンスは、一般に、ビューからの一連のビデオピクチャを含む。ピクチャオブグループ（GOP）は、概して、一連の 1 つ又は複数のビデオピクチャを備える。GOP は、GOP 中に含まれる幾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、GOP の 1 つ又は複数のピクチャのヘッダ中、又は他の場所に含み得る。各ピクチャは、それぞれのピクチャのための符号化モードを記述するピクチャ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、個々のビデオピクチャ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、H.264 規格において定義されるようなマクロブロック、マクロブロックの区分、及び場合によっては区分（partition）のサブブロック、又は、HEVC 規格において定義されるようなコード化単位（CU）、予測単位（PU）、又は変換単位（TU）に対応し得るが、本開示で説明される技法は、これらのブロックの例に限定されない。ビデオブロックは、固定サイズ又は可変サイズを有し得、指定されたコード化規格に従ってサイズが異なり得る。各ビデオピクチャは複数のスライスを含み得る。各スライスは複数のブロックを含み得る。

20

【0034】

[0043] 一例として、ITU-T H.264 規格は、ルーマ成分については 16×16 、 8×8 、又は 4×4 、及びクロマ成分については 8×8 など、様々なブロックサイズのイントラ予測をサポートし、並びにルーマ成分については 16×16 、 16×8 、 8×16 、 8×8 、 8×4 、 4×8 及び 4×4 、並びにクロマ成分については対応するスケール

リングされたサイズなど、様々なブロックサイズのインター予測をサポートする。本開示では、「 $N \times (x) N$ 」と「 $N \times (by) N$ 」は、垂直寸法及び水平寸法に関するブロックの画素寸法（例えば、 $16 \times (x) 16$ 画素又は $16 \times (by) 16$ 画素）を指すために互換的に使用され得る。一般に、 16×16 ブロックは、垂直方向に 16 画素を有し（ $y = 16$ ）、水平方向に 16 画素を有する（ $x = 16$ ）。同様に、 $N \times N$ ブロックは、概して、垂直方向に N 画素を有し、水平方向に N 画素を有し、但し、 N は非負整数値（nonnegative integer value）を表す。ブロック中の画素は行と列とに構成され得る。その上、ブロックは、必ずしも、水平方向において垂直方向と同じ数の画素を有する必要があるとは限らない。例えば、ブロックは $N \times M$ 画素を備えてよく、この場合に、 M は必ずしも N に等しいとは限らない。

30

40

【0035】

[0044] ブロックがイントラモード符号化される（例えば、イントラ予測される）とき、ブロックは、ブロック用のイントラ予測モードを記述するデータを含む場合がある。別の例として、ブロックがインターモード符号化される（例えば、インター予測される）とき、ブロックは、ブロックについての動きベクトルを定義する情報を含む場合がある。この動きベクトルは、同じビュー中の参照ピクチャを指す（例えば、時間動きベクトル）か、又は別のビュー中の参照ピクチャを指す（例えば、視差動きベクトル）。ブロックのための動きベクトルを定義するデータは、例えば、動きベクトルの水平成分と、動きベクトルの垂直成分と、動きベクトルの解像度（例えば、 $1/4$ 画素精度又は $1/8$ 画素精度）とを記述する。加えて、インター予測されるとき、ブロックは、動きベクトルが指す参照ピ

50

クチャなどの参照インデックス情報、及び/又は動きベクトル用の参照ピクチャリスト（例えば、RefPicList0若しくはRefPicList1）を含む場合がある。

【0036】

[0045] H.264規格では、イントラ予測又はインター予測コード化の後、ビデオエンコーダ20は、マクロブロックのための残差データを計算する。残差データは、符号化されていないピクチャの画素と、H.264におけるマクロブロックのための予測値との間の画素差分に対応し得る。

【0037】

[0046] 幾つかの例では、変換係数を生成するための任意の変換の後、ビデオエンコーダ20は、変換係数の量子化を実行する。量子化は、概して、係数を表すために使用されるデータの量をできるだけ低減するために変換係数が量子化され、更なる圧縮を行うプロセスを指す。量子化プロセスは、係数の一部又は全部に関連するビット深度を低減する。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、但し、nはmよりも大きい。

【0038】

[0047] 幾つかの例では、ビデオエンコーダ20は、エントロピー符号化され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する。他の例では、ビデオエンコーダ20は適応走査を実行する。1次元ベクトルを形成するために、量子化変換係数を走査した後、幾つかの例では、ビデオエンコーダ20は、幾つかの例として、コンテキスト適応型可変長コード化(CAVLC)、コンテキスト適応型バイナリ算術コード化(CABAC)、シンタックススペースコンテキスト適応型バイナリ算術コード化(SBAC)、確率間隔区分エントロピー(PIPE)コード化、又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する。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によって使用するための符号化された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する。

【0039】

[0048] CABACを実行するために、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0ではないかどうかに関係し得る。CAVLCを実行するために、ビデオエンコーダ20は、送信されるべきシンボルの可変長コードを選択し得る。VLC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、一方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、VLC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。確率決定は、シンボルに割り当てられたコンテキストに基づき得る。

【0040】

[0049] ビデオデコーダ30は、ビデオエンコーダ20の技法の逆を実施する。例えば、ビデオデコーダ30は、符号化されたビデオビットストリームを復号し、逆量子化及び逆変換によって残差ブロックを決定する。ビデオデコーダ30は、ピクチャ内のブロックのための画素値を決定するために、前に復号されたピクチャのブロックと残差ブロックとを合計する。

【0041】

[0050] 上記で説明されたように、本開示で説明される技法は、3D-AVCを対象とする。本技法をよりよく理解するために、下記は、幾つかのH.264/AVCコード化技法、H.264/MVC拡張及び高効率ビデオコード化(HEVC)規格の観点からのマルチビュービデオコード化、ならびに、3D-AVC技法について説明する。

【0042】

[0051] H.264/アドバンスドビデオコード化(AVC)では、ビデオ符号化又は復号(例えば、コード化)がマクロブロック上で実施され、但し、マクロブロックは、フレームの一部を表し、インター予測又はイントラ予測される(即ち、インター予測符号化

10

20

30

40

50

若しくは復号され、又はイントラ予測符号化若しくは復号される)。例えば、H.264/AVCでは、各インターマクロブロック(MB)(例えば、インター予測されたマクロブロック)は、1つの16×16MB区分、2つの16×8MB区分、2つの8×16MB区分、又は4つの8×8MB区分の4つの異なる方法に区分され得る。1つのMB中の異なるMB区分は、方向ごとに異なる参照インデックス値(即ち、RefPicList0又はRefPicList1)を有し得る。MBが複数の(1よりも多い)MB区分に区分されないとき、MBは、各方向に、MB区分全体のための1つの動きベクトルのみを有する。

【0043】

[0052]ビデオコード化(符号化又は復号)の一部として、ビデオエンコーダ20及びビデオデコーダ30は、RefPicList0及びRefPicList1と呼ばれる、1つ又は2つの参照ピクチャリストを構成するように構成される。参照ピクチャリスト(複数可)は、フレーム又はスライスのマクロブロックをインター予測するために使用され得る参照ピクチャを識別する。例えば、ビデオエンコーダ20は、参照インデックスと参照ピクチャリスト識別子とを信号伝達(signal)し得る。ビデオデコーダ30は、参照インデックスと参照ピクチャリスト識別子とを受信し、参照インデックスと参照ピクチャリスト識別子とから、現在のマクロブロックをインター予測復号するために使用されるべきである参照ピクチャを決定し得る。

【0044】

[0053]MBが4つの8×8MB区分に区分されるとき、各8×8MB区分は、サブブロックに更に区分され得る。8×8MB区分から、サブブロック、即ち、1つの8×8サブブロック、2つの8×4サブブロック、2つの4×8サブブロック、又は4つの4×4サブブロックを得るための、4つの異なる方法がある。各サブブロックは、各方向において異なる動きベクトルを有し得るが、各方向のためのエス同じ参照ピクチャインデックスを共有することができる。8×8MB区分がサブブロックに区分される方法は、サブブロック区分と称される。

【0045】

[0054]マルチビュービデオコード化では、複数の異なるビデオコード化規格がある。混乱を避けるために、本開示が一般的にマルチビュービデオコード化について説明するとき、本開示は「マルチビュービデオコード化」というフレーズを使用する。概して、マルチビュービデオコード化では、ベースビュー、及び、1つ又は複数の拡張ビュー又は依存ビューがある。ベースビューは、依存ビューのいずれにも関係なく、完全に復号可能である(即ち、ベースビューは、時間動きベクトルを用いてのみインター予測される)。これは、マルチビュービデオコード化のために構成されないコーデックが、完全に復号可能である少なくとも1つのビューをなお受信することを可能にする(即ち、ベースビューが抽出され、他のビューが破棄されて、マルチビュービデオコード化のために構成されないデコーダが、3D経験(3D experience)がないにもかかわらず、ビデオコンテンツをなお復号することが可能にされ得る)。1つ又は複数の拡張ビュー又は依存ビューは、ベースビューに関して、又は他の拡張ビュー(enhancement view)若しくは依存ビュー(dependent view)に関してインター予測され(即ち、視差補償予測され)、又は、同じビュー中の他のピクチャに関してインター予測され(即ち、動き補償予測され)得る。

【0046】

[0055]「マルチビュービデオコード化」が総称的に使用されるのに対して、頭文字MVCは、H.264/AVCの拡張に関連付けられる。従って、本開示が頭文字MVCを使用するとき、本開示は、特にH.264/AVCビデオコード化規格の拡張を指している。H.264/AVCのMVC拡張は、時間動きベクトルに加えて、別のタイプの動きベクトルとしての視差動きベクトルに依拠している。MVCプラス深度(MVC+D)と呼ばれる別のビデオコード化規格もまた、JCT-3V及びMPEGによって開発されている。MVC+Dは、テクスチャと深度の両方に対して、MVCのものと同一低レベルコード化ツールを適用し、深度の復号は、テクスチャの復号とは無関係であり、その逆も同様

10

20

30

40

50

である。例えば、MVCでは、フレームは、テキストビューコンポーネント又は単にテキストと呼ばれる、1つのビューコンポーネントのみによって表される。MVC + Dでは、テキストビューコンポーネント及び深度ビューコンポーネント、又は単にテキスト及び深度という、2つのビューコンポーネントがある。例えば、MVC + Dでは、各ビューは、テキストビューと深度ビューとを含み、但し、ビューは複数のビューコンポーネントを含み、テキストビューは複数のテキストビューコンポーネントを含み、深度ビューは複数の深度ビューコンポーネントを含む。

【0047】

[0056]各テキストビューコンポーネントは、ビューのビューコンポーネントを形成するために、深度ビューコンポーネントに関連付けられる。深度ビューコンポーネントは、テキストビューコンポーネント中のオブジェクトの相対深度を表す。MVC + Dでは、深度ビューコンポーネント及びテキストビューコンポーネントは、別個に復号可能である。例えば、ビデオデコーダ30は、第1のコーデックがテキストビューコンポーネントを復号し、第2のコード化されたものが深度ビューコンポーネントを復号する、MVCコーデックの2つのインスタンスを実装し得る。これらの2つのコーデックは、テキストビューコンポーネント及び深度ビューコンポーネントが別個に符号化されるので、互いに無関係に実行することができる。

10

【0048】

[0057]MVC + Dでは、深度ビューコンポーネントは常に、関連付けられた（例えば、対応する）テキストビューコンポーネントの直後にくる。このようにして、MVC + Dは、テキストビューコンポーネントが深度ビューコンポーネントより前に復号される、テキスト優先コード化をサポートする。

20

【0049】

[0058]テキストビューコンポーネント及びその関連付けられた（例えば、対応する）深度ビューコンポーネントは、同じピクチャ順序カウンタ（POC: picture order count）値とview_idとを含み得る（即ち、テキストビューコンポーネント及びその関連付けられた深度ビューコンポーネントのPOC値及びview_idは同じである）。POC値は、テキストビューコンポーネントの表示順序を示し、view_idは、テキストビューコンポーネント及び深度ビューコンポーネントが属する先のビューを示す。

30

【0050】

[0059]典型的なMVC復号順序（即ち、ビットストリーム順序）が図2に示される。復号順序構成は時間優先（time-first）コード化と呼ばれる。アクセス単位の復号順序は、出力又は表示の順序と同一ではない場合があることに留意されたい。図2では、S0 ~ S7は、それぞれマルチビュービデオの異なるビューを指す。T0 ~ T8は、それぞれ1つの出力時間インスタンスを表す。アクセス単位は、1つの出力時間インスタンスについての全てのビューのコード化されたピクチャを含み得る。例えば、第1のアクセス単位は時間インスタンスT0についてのビューS0 ~ S7の全てを含み得、第2のアクセス単位は時間インスタンスT1についてのビューS0 ~ S7の全てを含み得、以下同様である。

【0051】

[0060]簡潔にするために、本開示は以下の定義を使用し得る。

40

【0052】

ビューコンポーネント：単一のアクセス単位中のビューのコード化された表現。ビューが、コード化されたテキスト表現とコード化された深度表現の両方を含むとき、ビューコンポーネントは、テキストビューコンポーネントと深度ビューコンポーネントとを含み得る。

【0053】

テキストビューコンポーネント：単一のアクセス単位中のビューのテキストのコード化された表現。

【0054】

50

深度ビューコンポーネント：単一のアクセス単位中のビューの深度のコード化された表現。

【 0 0 5 5 】

[0061] 上記で説明されたように、本開示のコンテキストでは、ビューコンポーネント、テクスチャビューコンポーネント、及び深度ビューコンポーネントは、一般にレイヤと呼ばれることがある。図 2 では、ビューの各々はピクチャのセットを含む。例えば、ビュー S 0 はピクチャ 0、8、16、24、32、40、48、56、及び 64 のセットを含み、ビュー S 1 はピクチャ 1、9、17、25、33、41、49、57、及び 65 のセットを含み、以下同様である。各セットは 2 つのピクチャを含み、一方のピクチャはテクスチャビューコンポーネントと呼ばれ、他方のピクチャは深度ビューコンポーネントと呼ばれる。ビューのピクチャのセット内のテクスチャビューコンポーネント及び深度ビューコンポーネントは、互いに対応すると見なされ得る。例えば、ビューのピクチャのセット内のテクスチャビューコンポーネントは、そのビューのピクチャのセット内の深度ビューコンポーネントに対応すると見なされ、その逆も同様である（即ち、深度ビューコンポーネントはセット中のそのテクスチャビューコンポーネントに対応し、その逆も同様である）。本開示で使用する、深度ビューコンポーネントに対応するテクスチャビューコンポーネントは、単一のアクセス単位の同じビューの一部であるテクスチャビューコンポーネント及び深度ビューコンポーネントと見なされ得る。

10

【 0 0 5 6 】

[0062] テクスチャビューコンポーネントは、表示される実際の画像コンテンツを含む。例えば、テクスチャビューコンポーネントは、ルーマ（Y）成分と、クロマ（Cb 及び Cr）成分とを含み得る。深度ビューコンポーネントは、その対応するテクスチャビューコンポーネント中の画素の相対深度を示し得る。一例として、深度ビューコンポーネントは、ルーマ値のみを含むグレースケール画像である。言い換えれば、深度ビューコンポーネントは、画像コンテンツを搬送するのではなく、テクスチャビューコンポーネント中の画素の相対深度の測度を与え得る。

20

【 0 0 5 7 】

[0063] 例えば、深度ビューコンポーネント中の純白の画素は、対応するテクスチャビューコンポーネント中のその対応する 1 つ又は複数の画素が閲覧者から見てより近いことを示し、深度ビューコンポーネント中の純黒の画素は、対応するテクスチャビューコンポーネント中のその対応する 1 つ又は複数の画素が閲覧者から見てより遠いことを示す。黒と白との中間にあるグレーの様々な陰影は、異なる深度レベルを示す。例えば、深度ビューコンポーネント中の濃いグレーの画素は、テクスチャビューコンポーネント中のその対応する画素が、深度ビューコンポーネント中のより薄いグレーの画素よりも遠いことを示す。画素の深度を識別するためにグレースケールのみが必要とされるので、深度ビューコンポーネントの色値がいかなる目的も果たし得ないことから、深度ビューコンポーネントはクロマ成分を含む必要がない。

30

【 0 0 5 8 】

[0064] 深度を識別するためにルーマ値（例えば、強度値）のみを使用する深度ビューコンポーネントが説明のために提供され、限定するものと見なされるべきではない。他の例では、テクスチャビューコンポーネント中の画素の相対深度を示すために任意の技法が利用され得る。

40

【 0 0 5 9 】

[0065] マルチビュービデオコード化用の（各ビュー内のピクチャ間予測と、ビュー間のビュー間予測の両方を含む）典型的な M V C 予測構造が図 3 に示されている。予測方向は矢印によって示され、矢印の終点のオブジェクトは、予測参照として矢印の始点のオブジェクトを使用する。M V C では、H . 2 6 4 / A V C 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を可能にする視差動き補償によって、ビュー間予測がサポートされる。

【 0 0 6 0 】

50

【0066】図3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個の時間位置(「T0」～「T11」)がビューごとに示されている。即ち、図3中の各行はビューに対応し、一方各列は時間位置(temporal location)を示す。

【0061】

【0067】MVCがH.264/AVCデコーダによって復号可能である、所謂ベースビューを有し、また、ステレオビューペアがMVCによってもサポートされ得るが、MVC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リング装置(renderer)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。

10

【0062】

【0068】図3中のピクチャは、各行と各列の交点に示されている。H.264/AVC規格は、ビデオの一部分を表すためにフレームという用語を使用し得る。本開示では、ピクチャという用語とフレームという用語とを互換的に使用し得る。

【0063】

【0069】図3中のピクチャは、対応するピクチャがイントラコード化される(即ち、Iピクチャである)か、又は一方向に(即ち、Pピクチャとして)又は複数の方向に(即ち、Bピクチャとして)インターコード化されるかを指定する、文字を含むブロックを使用して示されている。概して、予測は矢印によって示され、ここで矢印の終点のピクチャは、予測参照のために矢印の始点のピクチャを使用する。例えば、時間位置T0にあるビューS2のPピクチャは、時間位置T0にあるビューS0のIピクチャから予測される。

20

【0064】

【0070】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ド化ビデオシーケンスのピクチャは、異なる時間位置にあるピクチャに関して予測符号化され得る。例えば、時間位置T1にあるビューS0のbピクチャは、時間位置T0にあるビューS0のIピクチャからそのbピクチャに向けられた矢印を有し、その矢印は、bピクチャがIピクチャから予測されることを示す。しかしながら、更に、マルチビュービデオ符号化のコンテキストにおいて、ピクチャはビュー間予測され得る。即ち、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することができる。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拡張において信号伝達され、インター予測又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。ビュー間予測は、3D-HEVC(マルチビュープラス深度)を含むHEVCの提案されたマルチビュー拡張の機能でもある。

30

【0065】

【0071】図3は、ビュー間予測の様々な例を提供する。図3の例では、ビューS1のピクチャは、ビューS1の異なる時間位置にあるピクチャから予測されるものとして、及び同じ時間位置にあるビューS0及びS2のピクチャ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、時間位置T1にあるビューS1のbピクチャは、時間位置T0及びT2にあるビューS1のBピクチャの各々、ならびに時間位置T1にあるビューS0及びS2のbピクチャから予測される。

40

【0066】

【0072】幾つかの例では、図3はテクスチャビューコンポーネントを示すものとして見なされ得る。例えば、図2に示されたIピクチャ、Pピクチャ、Bピクチャ、及びbピクチャは、ビューの各々のためのテクスチャビューコンポーネントと見なされ得る。本開示で説明される技法によれば、図3に示されたテクスチャビューコンポーネントの各々について、対応する深度ビューコンポーネントがある。幾つかの例では、深度ビューコンポーネントは、対応するテクスチャビューコンポーネントについて図3に示された方法と同様の方法で予測され得る。

50

【 0 0 6 7 】

[0073] 2つのビューのコード化もMVCによってサポートされ得る。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り得、MVCデコーダがそのようなマルチビュー表現を復号し得ることである。従って、MVCデコーダをもついかなるレンダラも、3つ以上のビューをもつ3Dビデオコンテンツを復号し得る。

【 0 0 6 8 】

[0074] 上記で説明されたように、MVCでは、(幾つかの事例では、同じ時間インスタンスをもつことを意味する) 同じアクセス単位中のピクチャ間で、ビュー間予測が可能にされる。非ベースビューのうちの1つの中のピクチャをコード化するとき、ピクチャが異なるビュー内にあるが同じ時間インスタンス内にある場合、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。図3に示されたように、ビューコンポーネントは、参照用に他のビュー中のビューコンポーネントを使用することができる。MVCでは、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。

10

【 0 0 6 9 】

[0075] MVCでは、同じアクセス単位中の(即ち、同じ時間インスタンスをもつ) ピクチャ間でビュー間予測が可能にされる。非ベースビューのうちの1つ中のピクチャをコード化するとき、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をもつ場合、そのピクチャは参照ピクチャリストに追加され得る。ビュー間予測参照ピクチャは、任意のインター予測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。

20

【 0 0 7 0 】

[0076] 図3に示されたように、ビューコンポーネントは、参照用に他のビュー中のビューコンポーネントを使用することができる。これはビュー間予測と呼ばれる。MVCでは、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。

【 0 0 7 1 】

[0077] マルチビュービデオコード化のコンテキストでは、2種類の動きベクトルが存在する。1つは、時間参照ピクチャを指す通常の動きベクトルである。対応する時間インター予測は、動き補償予測(MCP: motion-compensated prediction)である。他方のタイプの動きベクトルは、異なるビュー中のピクチャ(即ち、ビュー間参照ピクチャ)を指す視差動きベクトルである。対応するインター予測は、視差補償予測(DCP: disparity-compensated prediction)である。

30

【 0 0 7 2 】

[0078] 現在、VCEG及びMPEGのJoint Collaboration Team on 3D Video Coding(JCT-3V)は、H.264/AVCに基づく3DV規格、即ち、3D-AVCを開発中である。3D-AVCでは、MVCにおけるビュー間予測の他に、新しいコード化ツールが含まれ、サポートされている。3D-AVCのための最新のソフトウェア3D-ATMは、以下のリンクからダウンロード可能であり、即ち、[3D-ATMバージョン6.2]: <http://mpeg3dv.research.nokia.com/svn/mpeg3dv/tags/3DV-ATMv6.2/>である。

40

【 0 0 7 3 】

[0079] AVCベース3Dビデオ(3D-AVC)コード化規格は、JCT-3Vによって現在開発中であり、3D-AVCの最新のバージョンは、現在公に入手可能であり、即ち、M. M. Hannuksela、Y. Chen、T. Suzuki、J. -R. Ohm、G. J. Sullivan、「3D-AVC draft text 5」、JCT3V-C1002、ジュネーブ、スイス、2013年1月である。それは、以下のリンクから入手可能であり、参照により本明細書に組み込まれ、即ち、<http://mpeg3dv.research.nokia.com/svn/mpeg3dv/tags/3DV-ATMv6.2/>

50

: / / p h e n i x . i t - s u d p a r i s . e u / j c t 2 / d o c _ e n d _ u s e r / d o c u m e n t s / 3 _ G e n e v a / w g 1 1 / J C T 3 V - C 1 0 0 2 - v 3 . z i p である。

【 0 0 7 4 】

[0080] 3D - AVC は、ベースビューのテクスチャ部分が H . 2 6 4 / AVC デコーダにとって完全に復号可能である形で、H . 2 6 4 / AVC と互換性がある。例えば、ベースビューのビューコンポーネント中のテクスチャビューコンポーネントは、同じベースビュー中の他のテクスチャビューコンポーネントのみを用いてインター予測され得る。ベースビュー中のテクスチャビューコンポーネントは、ビュー間予測されなくてよい。また、ベースビュー中のテクスチャビューコンポーネントは、復号の目的のために対応する深度ビューコンポーネントを必要としなくてよい。

10

【 0 0 7 5 】

[0081] 3D - AVC における拡張ビューコンポーネントでは、幾つかの他の例示的な技法において、深度がテクスチャより前にコード化され得、テクスチャビューコンポーネントが、深度ビューコンポーネントからの情報に基づいてコード化され得、これは深度優先 (depth-first) コード化としても知られる。但し、各テクスチャビューコンポーネントは、上記で説明された MVC + D におけるような、テクスチャ優先コード化順序において、それぞれの深度ビューコンポーネントの前でコード化される。言い換えれば、幾つかの他の例示的な技法では、3D - AVC では、ベースビューのテクスチャビューコンポーネントが最初にコード化され、ベースビューの関連付けられた深度ビューコンポーネントによって後続され、第 1 の拡張ビュー又は依存ビューの深度ビューコンポーネントによって後続され、第 1 の拡張ビュー又は依存ビューの関連付けられたテクスチャビューコンポーネントによって後続され、第 2 の拡張ビュー又は依存ビューの深度ビューコンポーネントによって後続され、第 2 の拡張ビュー又は依存ビューの関連付けられたテクスチャビューコンポーネントによって後続され、以下同様である。

20

【 0 0 7 6 】

[0082] 例えば、3D - AVC におけるテクスチャビューコンポーネント及び深度ビューコンポーネントのコード化順序は、次のように例示される。以下の例では、T0 及び D0 は、それぞれ、ベースビューのテクスチャビューコンポーネントと深度ビューコンポーネントとを指し、Ti 及び Di は、それぞれ、i 番目の依存ビューのテクスチャビューコンポーネントと深度ビューコンポーネントとを指す。以下の例では、3つのビューが考慮される。

30

【 0 0 7 7 】

[0083] 第 1 の例では、考慮されるビューは、T0、D0、D1、D2、T1、及び T2 である。この例では、ベースビュー (T0 及び D0) は、テクスチャ優先コード化順序でコード化されるが、一方依存ビューは、深度優先コード化順序でコード化される。ハイブリッドコード化順序が、3D - AVC の共通試験条件において現在使用されている。別の例では、コード化の順序は、T0、D0、T1、D1、T2、及び D2 である。即ち、全てのビューコンポーネントが、テクスチャ優先コード化順序でコード化される。ビュー間予測が Ti に対して可能にされる場合、参照テクスチャビューは、ビュー間参照ピクチャを含むビューとして定義され、対応する深度ビューは、参照テクスチャビューのものと同一ビュー順序インデックスを有する参照深度ビューとして定義される。

40

【 0 0 7 8 】

[0084] テクスチャビューコンポーネントのブロックのための視差ベクトルを導出することは、対応する深度ビューコンポーネントを必要としたので、幾つかの他の 3D - AVC 技法は、深度優先コード化を必要とした。下記は、深度マップを介したそのような視差ベクトル導出について説明する。視差ベクトルを導出するための技法は、各低レベルコード化ツールとともに変化し得るが、一般に、依存ビューの深度データは、テクスチャビューコンポーネントコード化のための視差ベクトル導出のために使用される。これは、深度優先コード化順序のために、依存ビューの深度ビューが利用可能であるからである。使用さ

50

れる低レベルコード化ツールは、3D-AVCにおける、インループブロックベースビュー合成ビュー間予測(BVSP)及び深度ベース動きベクトル予測(D-MVP: depth-based motion vector prediction)である。ビデオコーデ、例えば、ビデオデコーデ30は、(依存フレームと呼ばれることがある)依存ビュー中の(深度マップと呼ばれることがある)深度ビューの深度値から変換された視差ベクトルを使用し得る。3D-AVC参照ソフトウェアでは、典型的には、実際の深度マップ値から特定のビューに対する視差への変換プロセスの結果が、カメラパラメータとともにルックアップテーブルに記憶される。

【0079】

[0085]図4は、後方ワーピングに基づくBVSPの概念図である。BVSPは、W. S. 他による「3DV-CE1. a: Block-Based View Synthesis Prediction for 3DV-ATM」(JCT3V-A0107)において最初に提案されており、それは、以下のリンクからダウンロード可能であり、参照により本明細書に組み込まれ、即ち、http://phenix.it-sudparis.eu/jct2/doc_end_user/documents/1__Stockholm/wg11/JCT3V-A0107-v1.zipである。図4を参照すると、以下のコード化順序、即ち、(T0、D0、D1、T1)が利用されると仮定する。テクスチャコンポーネントT0は、ベースビューであり、T1は、VSP(ビュー合成予測)でコード化された依存ビューである。深度マップコンポーネントD0及びD1は、T0及びT1に関連付けられたそれぞれの深度マップである。

【0080】

[0086]依存ビューT1では、現在コード化されているブロックCbのサンプル値が、ベースビューT0のサンプル値からなる参照エリアR(Cb)から予測される(VSP予測)。コード化されるべき現在のサンプルと参照サンプルとの間の変位ベクトル(Disp_vec)は、現在コード化されているテクスチャサンプルに関連付けられた深度マップ値からのT1とT0との間の導出された視差ベクトルとして示される。

【0081】

[0087]深度値から視差ベクトルへの変換のプロセスは、例えば以下の式を用いて実行され得る。

【数1】

$$Z(Cb(j,i)) = \frac{1}{\frac{d(Cb(j,i))}{255} \cdot \left(\frac{1}{Z_{near}} - \frac{1}{Z_{far}} \right) + \frac{1}{Z_{far}}}; \quad (1)$$

【数2】

$$D(Cb(j,i)) = \frac{f \cdot b}{Z(Cb(j,i))}; \quad (2)$$

【0082】

但し、j及びiは、Cb内のローカル空間座標であり、dCb(j,i)は、ビュー1の深度マップ画像における深度マップ値であり、Zは、実際の対応する深度値であり、Dは、特定のビュー0への導出された視差ベクトルの水平成分である。パラメータf、b、Znear及びZfarは、カメラセットアップを指定するパラメータであり、即ち、使用される焦点距離(f)、ビュー#1とビュー#0との間のカメラ分離(b)、及び、深度マップ変換のパラメータを表す深度範囲(Znear、Zfar)である。

【0083】

[0088]幾つかの例では、導出された視差ベクトルの垂直成分が0に設定されることに留意されたい。また、幾つかの3DV-ATM実装形態では、式(1)及び(2)は、あら

ゆる深度マップ値 (0 . . . 2 5 5) についてすでに事前計算され、ルックアップテーブルとして記憶されている。

【 0 0 8 4 】

[0089] 次のセクションは、B V S P の幾つかの実装問題について説明する。1 つの問題は、B V S P ブロックの指示を含む。B V S P ブロックは、次のように示される。

【 0 0 8 5 】

- M B レベルにおける 1 つのフラグが、現在の M B が従来のスキップ / 直接モードでコード化されるか、スキップ / 直接モードでコード化されるが、合成参照コンポーネントから予測されるかを、信号伝達するために使用される。

【 0 0 8 6 】

- M B 区分 (1 6 × 1 6 から 8 × 8 まで) ごとに、各参照ピクチャリスト中の参照インデックス (又は、3 D - A V C のための幾つかの提案と同様に、フラグ) が、参照ピクチャを信号伝達するために使用される。区分が B V S P モードでコード化されるとき、B V S P コード化されたブロックのための動きベクトルがないので、動きベクトル差分は信号伝達されない。

【 0 0 8 7 】

[0090] フラグ又は参照インデックスのいずれかが合成参照コンポーネントを示すとき、以下の項目において説明されるような 1 つの区分の予測が呼び出される。そのサイズが N × M によって示される (但し、N 又は M は、8 又は 1 6 とする)、M B 区分ごとに、その M B 区分が B V S P モードでコード化される場合、現在の M B 区分は、K × K (但し、K は、3 D - A V C のための幾つかの提案と同様に 8 × 8 であるか、4 × 4、2 × 2、又は 1 × 1 であり得る) に等しいサイズをもつ幾つかの下位領域に更に区分される。下位領域ごとに、別個の視差ベクトルが導出され、各下位領域が、ビュー間参照ピクチャ中の導出された視差ベクトルによって位置を特定された 1 つのブロックから予測され、即ち、図 4 における R (c b) である。幾つかの例示的な共通試験条件では、K は 4 になるように定義される。導出された視差ベクトルは、B V S P コード化されたブロックについては記憶されず、その理由は、そのようなベクトルを使用するコード化ツールがないからであることに留意されたい。

【 0 0 8 8 】

[0091] 別の実装問題は、視差ベクトル導出プロセスを含む。深度優先コード化順序が適用されるとき、導出された視差ベクトルは、図 4 に示されるように、対応する非ベース深度ビュー中の対応する深度ブロックの深度値を変換することによって、取得され得る。幾つかの技法が、深度ブロックの中心位置の深度値、1 つの深度ブロック内の全ての深度値の最大値、1 つの深度ブロック内の 4 隅の画素の最大値、及び深度ブロック / 深度 M B の右下画素の深度値など、1 つの深度ブロックの深度値を選択するために適用され得る。テクスチャ優先コード化順序が適用されるとき、非ベーステクスチャビューを復号するときに対応する非ベース深度ビューが利用不可能であるので、B V S P モードが無効化されることになる。

【 0 0 8 9 】

[0092] 通常のインターモードのための 3 D - A V C における深度ベース動きベクトル予測 (D - M V P) について、次に説明する。D - M V P は、深度優先コード化順序のために利用可能である、現在のビュー中の関連付けられた深度マップデータを組み込む、動きベクトル予測方法を指す。その方法は、依存ビュー中のテクスチャビューコンポーネントとともに適用される。

【 0 0 9 0 】

[0093] 3 D - A V C では、D - M V P 方法が、H . 2 6 4 / A V C の従来の中央値関数ベース動きベクトル予測に組み込まれる。具体的には、予測されるべき動きベクトルのタイプ (即ち、時間動きベクトルか視差動きベクトルか) が、隣接ブロック中の動きベクトルの参照インデックスから最初に識別され、従って、動き予測のタイプが決定される。図 8 に示されるように、現在の区分のための隣接ブロックは、現在のブロックに対して、順

10

20

30

40

50

に、(「A」と示される)左ブロックと、(「B」と示される)上ブロックと、(「C」と示される)右上ブロックと、(「D」と示される)左上ブロックとを含み得る。左上ブロック中の動きベクトルは、他の3つの隣接ブロックのうちの1つが動きベクトルを含んでおらず、従って利用不可能であると見なされるとき、使用され得る。

【0091】

[0094] 3つの隣接ブロックの動きベクトルが利用可能であると仮定すると、3つの隣接ブロック中の動きベクトルが、現在のブロックの動きベクトル予測のために採用される。時間的予測では、それらの動きベクトルが全て同じタイプを有し、同じ参照インデックスを有する場合、H.264/AVCと同様に、メディアンフィルタが直接使用される。他の場合、動きベクトルが異なるタイプに属しており、異なる参照インデックスを有する場合、現在のブロックのための動きベクトルが更に導出される。現在の参照ピクチャがビュー間参照ピクチャであるとき、隣接ブロック位置における動きベクトルタイプ及びそれらの参照インデックスがチェックされる。隣接ブロックが全て同じタイプと同じ参照インデックスとを有する場合、メディアンフィルタが適用される。どちらの場合も、3つよりも少ない隣接ブロックが利用可能である場合、3つの隣接ブロックが利用可能になるように、利用不可能なブロックのための動きベクトルが更に導出される。

【0092】

[0095] 隣接ブロックのために導出された動きベクトルは、導出された動きベクトルと呼ばれ、次のように生成される。現在の動きベクトルが視差動きベクトルであり、隣接ブロックの動きベクトルが現在の動きベクトルのものとは異なるタイプを有する(又は、利用不可能である)場合、隣接ブロックの導出された動きベクトルは、対応する深度ビューコンポーネントから変換される、視差動きベクトルになるように設定される。同じビューの深度ビューコンポーネントの対応するブロックが使用され、この対応するブロックの4隅の深度値の最大値が視差値に変換され、その視差値が、導出された動きベクトルの水平成分になる。導出された動きベクトルの垂直成分は、ゼロになるように設定される。

【0093】

[0096] 現在の動きベクトルが時間動きベクトルである場合、(上記で説明されたように導出された)視差値は、参照(ベース)ビュー中の参照ブロックの時間動きベクトルを決定するために使用され、導出された動きベクトルは、時間動きベクトルになるように設定される。時間動きベクトルが利用不可能であると見なされる場合(例えば、イントラブロックの場合、又は、動きベクトルが、現在の参照ピクチャとともに整合された参照ビュー中の参照ピクチャを指さない場合)、導出された動きベクトルは、ゼロに設定される。

【0094】

[0097] スキップモード及び直接モードのための3D-AVCにおけるビュー間動き予測について、次に説明する。3D-AVCにおけるビュー間動き予測は、Pスキップ、Bスキップ、B-16×16直接モード、及びB-8×8直接モードで実行される。視差ベクトルは、隣接ブロック、ならびに、同じビューの深度ビューコンポーネントの深度値から変換された視差ベクトルを導出することから、最初に導出され得る。

【0095】

[0098] 1つの利用可能な空間隣接ブロックが、視差動きベクトルを含んでいる場合、この視差動きベクトルが、現在のブロックのための視差ベクトルになる。他の場合、視差動きベクトルを含んでいない隣接ブロックでは、現在のブロックのために使用されるべき視差動きベクトルが、(D-MVPにおける変換と同様に)同じビューに対応する深度値から変換される。幾つかの例では、視差ベクトルを取得するために、メディアンフィルタが3つの隣接ブロックに適用される。

【0096】

[0099] 導出されたベクトルは、参照(ベース)ビュー中の参照ブロックに対する時間動きベクトルを取得するために使用され得る。時間動きベクトルが利用不可能である場合、参照インデックスが最初に導出され得、上記で説明されたD-MVPプロセスが、動きベクトル予測子を生成するために適用される。

【 0 0 9 7 】

[0100]隣接ブロックベース視差ベクトル (N B D V) 導出について、次に説明する。N B D V は、テクスチャ優先コード化順序が全てのビューに対して使用されるとき、3 D - H E V C において視差ベクトル導出方法として使用される。現在の3 D - H E V C 設計では、N B D V 導出はまた、参照ビューの深度マップから深度データを取り出すためにも使用される。

【 0 0 9 8 】

[0101]参照ソフトウェア記述の1つのバージョン、及び3 D - H E V C のワーキングドラフトは、次のように入手可能となり、参照により本明細書に組み込まれ、即ち、Gerhard Tech、Krzysztof Wegner、Ying Chen、Sehoon Yea、「3 D - H E V C Test Model Description draft 2」、JCT3V - B1005、ITU - T SG16 WP3とISO / IEC JTC1 / SC29 / WG11とのJoint Collaborative Team on 3D Video Coding Extension Development、第2回会合：上海、中国、2012年10月である。

【 0 0 9 9 】

[0102]視差ベクトル (D V) は、2つのビューの間の視差を推定するものとして使用される。即ち、視差ベクトルは、現在のピクチャ中のブロックに関する、同じ時間インスタンス中のすでにコード化されたピクチャ中の対応するブロックへの、ポインタである。隣接ブロックは、ビデオコード化においてほぼ同じ動き / 視差情報を共有するので、現在のブロックは、良い予測子として、隣接ブロック中の動きベクトル情報を使用することができる。この考えに従って、N B D V 導出プロセスは、異なるビュー中の視差ベクトルを推定するために、隣接ブロックの視差情報を使用する。

【 0 1 0 0 】

[0103]N B D V 導出を実行するために、候補隣接ブロックが最初に定義される。2つのセットの隣接候補ブロックが利用される。一方のセットは、空間隣接ブロックからのものであり、他方のセットは、時間隣接ブロックからのものである。空間及び時間隣接候補ブロックの各々が、次いで、現在のブロックと候補ブロックとの間の相関の優先度によって決定された、予め定義された順序でチェックされる。視差動きベクトル (即ち、動きベクトルがビュー間参照ピクチャを指す) が候補中で発見されると、視差動きベクトルが視差ベクトルに変換される。

【 0 1 0 1 】

[0104]3 D - H E V C におけるN B D V 導出の具体例について、次に説明する。3 D - H E V C は、JCT3V - A0097において提案された隣接ブロック (ベース) 視差ベクトル (N B D V) 方法を、最初に採用した。暗黙的視差ベクトル (implicit disparity vector) が、JCTVC - A0126において簡略化されたN B D V 導出プロセスとともに含まれた。それに加えて、JCT3V - B0047では、N B D V 導出は、復号されたピクチャバッファに記憶された暗黙的視差ベクトルを除去することによって、更に簡略化されるが、また、ランダムアクセスポイント (R A P) ピクチャ選択を用いてコード化利得も改善した。

【 0 1 0 2 】

JCT3V - A0097 : 3 D - CE5 . h : Disparity vector generation results、L . Zhang、Y . Chen、M . Karczewicz (Qualcomm) は、参照により本明細書に組み込まれる。

【 0 1 0 3 】

JCT3V - A0126 : 3 D - CE5 . h : Simplification of disparity vector derivation for HEVC - based 3D video coding、J . Sung、M . Koo、S . Yea (LG) は、参照により本明細書に組み込まれる。

【 0 1 0 4 】

JCT3V-B0047:3D-CE5.h related:Improvements for disparity vector derivation、J. Kang、Y. Chen、L. Zhang、M. Karczewicz (Qualcomm) は、参照により本明細書に組み込まれる。

【0105】

[0105]NB DV 導出のための幾つかの提案では、5つの空間隣接ブロックが、視差ベクトル導出のために使用される。図5に示されるように、5つの空間隣接ブロックは、A0、A1、B0、B1又はB2によって示されるような、現在のPU500の左下ブロック、左ブロック、右上ブロック、上ブロック、及び左上ブロックである。それらは、HEVCにおけるマージモードにおいて使用されたものと同じであることに留意されたい。従って、追加のメモリアクセスが必要とされない。

10

【0106】

[0106]時間隣接ブロックをチェックする前に、候補ピクチャリストの構成プロセスが最初に行われる。現在のビューからの全ての参照ピクチャが、候補ピクチャとして扱われ得る。同一位置配置された参照ピクチャが最初に候補ピクチャリストに挿入され、参照インデックスの昇順に候補ピクチャの残りによって後続される。両方の参照ピクチャリスト中で同じ参照インデックスをもつ参照ピクチャが利用可能であるとき、同一位置配置ピクチャ (co-located picture) と同じ参照ピクチャリスト中にある参照ピクチャが、一致する参照インデックスを有する他の参照ピクチャに先行する。候補ピクチャリスト中の候補ピクチャごとに、時間隣接ブロックを導出するために3つの候補領域が決定される。

20

【0107】

[0107]ブロックがビュー間動き予測でコード化されるとき、異なるビュー中の対応するブロックを選択するために、視差ベクトルが導出される。ビュー間動き予測において導出された視差ベクトルは、暗黙的視差ベクトル (IDV) と呼ばれることがある。ブロックが動き予測でコード化されるときも、IDVは、後続のブロックをコード化する目的のために破棄されない。

【0108】

[0108]典型的には、NB DV 導出プロセスは、時間隣接ブロック中の視差動きベクトルと、空間隣接ブロック中の視差動きベクトルと、次いでIDVとを、その順序でチェックすることを伴う。視差ベクトルが発見されると、プロセスは終了させられる。

30

【0109】

[0109]3D-HEVCにおける後方VSPについて、次に説明する。3D-HEVCでは、テクスチャ優先コード化順序が適用されるとき、予測単位 (PU) ごとに、参照深度ビュー中の深度値を考慮するかもしれないにもかかわらず、視差ベクトルがNB DV 導出プロセスから導出され得る。視差ベクトルが取得された後、視差ベクトルは、1つのPUがBVSPモードでコード化される場合、そのPUのM×N (但し、M/Nは、例えば、8又は4に等しくなり得る) 下位領域ごとに更に精密化されることになる。

【0110】

[0110]精密化プロセスは、2つのステップを含み、即ち、1) 導出された視差ベクトルによって位置を特定される参照深度ビュー中のM×N深度ブロックから1つの最大深度値を選択し、2) 正確な視差ベクトルの垂直成分を0になるように保ちながら、深度値を正確な視差ベクトルの水平成分に変換する。視差ベクトルが1つのPUの1つのM×N下位領域について精密化された後、正確な視差ベクトルが、動き補償のために参照テクスチャビュー中で1つのブロックの位置を特定するために使用される。

40

【0111】

[0111]3D-AVCにおけるNB DV 導出プロセスについて、次に説明する。参照により本明細書に組み込まれる、2014年2月25日に出願された、同時係属米国特許出願第14/189,177号に記載されているように、MBレベルNB DV 導出は、現在のMBのための視差ベクトルを導出するために使用され得る。導出された視差ベクトルは、動きベクトル予測のために更に使用され得る。視差動きベクトルが識別されると、即ち、

50

時間又は空間隣接ブロックのうちの1つがビュー間参照ピクチャを使用すると、それが現在のMBのための視差ベクトルとして返される。

【0112】

[0112]下記は、米国特許出願第14/189,177号の技法について、より詳細に説明する。幾つかの以前の3D-AVC技法は、テクスチャビューコンポーネントの深度ビューコンポーネントが視差ベクトル導出のために利用可能であることを必要としており（即ち、依存ビュー又は拡張ビューのための深度優先コード化を必要とする）、そのことが、復号待ち時間、実装の複雑さ、他のビデオコード化規格へのスケーラビリティの欠如、深度ビューコンポーネントが必要とされない場合の帯域幅の非効率性、及び他の潜在的な欠点などの問題につながる。

10

【0113】

[0113]米国特許出願第14/189,177号に記載された技法は、対応する深度ビューコンポーネントに依拠することを必要としない視差ベクトル導出を可能にする。このようにして、それらの技法は、視差ベクトル導出を用いた依存ビューのための3D-AVCにおけるテクスチャ優先コード化を可能にする。視差ベクトル導出を実現するために、米国特許出願第14/189,177号に記載された技法は、隣接ブロックの動きベクトル情報に依拠する。一例として、隣接ブロックのための動きベクトルが視差動きベクトルである場合、それらの技法は、現在のブロックのための視差ベクトルとして、隣接ブロックの視差動きベクトルを利用する。このようにして、ビデオエンコーダ20及びビデオデコーダ30は、対応する深度ビューコンポーネントに依拠する必要なしに、テクスチャビューコンポーネントの現在のマクロブロックのための視差ベクトルを決定し得る。

20

【0114】

[0114]ビデオエンコーダ20及びビデオデコーダ30は、米国特許出願第14/189,177号に記載された技法を実装するように構成され得る。例えば、ビデオエンコーダ20及びビデオデコーダ30は、テクスチャビューがビューコンポーネントごとに最初にコード化されることを可能にすることによって、3D-AVCの効率的なコード化を可能にする技法を実装するように構成され得る。ビデオエンコーダ20及びビデオデコーダ30は、テクスチャ優先コード化順序のために、対応する深度データが3D-AVCにおいて利用可能ではない（又は、まだ利用可能ではない）とき、現在のブロックの空間/時間隣接ブロックからの2つ以上の利用可能な視差動きベクトルを考慮するNB-DVの概念を使用して、視差ベクトルを導出し得る。

30

【0115】

[0115]一例として、ビデオデコーダ30は、依存ビューのテクスチャ優先コード化を用いて生成された、3D-AVC準拠ビデオコード化プロセスにおけるコード化されたビットストリームを受信し得る。この例では、3D-AVC準拠ビデオコード化プロセスは、3D-AVCビデオコード化規格において定義されたビデオコード化ツールを使用するビデオコード化プロセスを指す。依存ビューのテクスチャ優先コード化は、テクスチャビューコンポーネントが対応する深度ビューコンポーネントより前にコード化される（即ち、T0、D0、T1、D1など）場合を指す。

【0116】

40

[0116]ビデオデコーダ30は、3D-AVC準拠ビデオコード化プロセスにおいて、依存ビューのうちの依存ビューのテクスチャビューコンポーネントを復号し得る。この例では、テクスチャビューコンポーネントを復号するために、ビデオデコーダ30は、少なくとも1つの隣接ブロックが、依存ビュー以外のビュー中のビュー間参照ピクチャを参照する視差動きベクトルを用いてビュー間予測されるかどうかを決定するために、テクスチャビューコンポーネント中の現在のブロックの1つ又は複数の隣接ブロックの動き情報を評価するように構成され得る。また、テクスチャビューコンポーネントを復号するために、ビデオデコーダ30は、隣接ブロックのうちの1つのための視差動きベクトルに基づいて、現在のブロックのための視差ベクトルを導出するように構成され得る。ビデオデコーダ30は、テクスチャビューコンポーネントを復号することに続いて、テクスチャビューコ

50

ンポーネントに対応する深度ビューコンポーネントを復号し得る。

【0117】

[0117]別の例として、ビデオエンコーダ20は、3D-AVC準拠ビデオコード化プロセスにおいて、依存ビューのテクスチャビューコンポーネントを符号化し得る。この例では、テクスチャビューコンポーネントを符号化するために、ビデオエンコーダ20は、少なくとも1つの隣接ブロックが、依存ビュー以外のビュー中のビュー間参照ピクチャを参照する視差動きベクトルを用いてビュー間予測されるかどうかを決定するために、テクスチャビューコンポーネント中の現在のブロックの1つ又は複数の隣接ブロックの動き情報を評価するように構成され得る。また、テクスチャビューコンポーネントを符号化するために、ビデオエンコーダ20は、隣接ブロックのうちの1つのための視差動きベクトルに基づいて、現在のブロックのための視差ベクトルを導出するように構成され得る。

10

【0118】

[0118]ビデオエンコーダ20は、テクスチャビューコンポーネントを符号化することについて、テクスチャビューコンポーネントに対応する深度ビューコンポーネントを符号化し得る。ビデオエンコーダ20はまた、符号化されたテクスチャビューコンポーネントと符号化された深度ビューコンポーネントとを含む、依存ビューのテクスチャ優先コード化を用いてコード化されたビットストリームを、出力のために生成し得る。

【0119】

[0119]3D-AVCのための現在の提案は、以下の問題を呈している。米国特許出願第14/189,177号に記載されたNBDV方法を利用するとき、BVSPは、主に、視差ベクトルが常に十分に正確であるとは限らないという理由のために、より効率的でなくなる。また、BVSPブロックのための(例えば、NBDVからの)導出された視差ベクトルは、コード化されるべきブロックのためにより正確な視差ベクトルを提供し得る。但し、BVSPにおけるそのような導出された視差ベクトルの使用は、NBDV導出とともに以前に採用されていなかった。

20

【0120】

[0120]これらの欠点に鑑みて、本開示は、非ベーステクスチャビューコンポーネントが、対応する非ベース深度ビューコンポーネントの前にコード化されるとき、3D-AVC準拠ビデオエンコーダ及びビデオデコーダのためのBVSPを可能にするための解決策を提供する。加えて、他のインターコード化モードのコード化利得もまた、本開示の技法によって提供されるような、精密な視差ベクトルの導出により、改善される。

30

【0121】

[0121]最初に、改善されたNBDV導出プロセスが、本開示で提案される。本開示のNBDV導出プロセスは、BVSPを組み込むように変更されるが、本開示の幾つかの態様は、必ずしもBVSPを必要とするとは限らない。以下の説明は、ビデオデコーダ30に関して説明されるが、以下の技法の各々がビデオエンコーダ20によっても同様に実装され得ることを理解されたい。ビデオデコーダ30とビデオエンコーダ20の両方は、本開示の技法を実行するように構成された1つ又は複数のプロセッサとともに実装され得る。幾つかの例では、ビデオデコーダ30及びビデオエンコーダ20の1つ又は複数のプロセッサは、1つ又は複数の非一時的コンピュータ可読記憶媒体上に記憶されたソフトウェアを実行するように構成され得る。また、本開示のコンテキストでは、「ビデオコード」は、ビデオエンコーダとビデオデコーダの両方に適用される総称語であることも理解されたい。同様に、「ビデオコード化」という用語は、ビデオ符号化又はビデオ復号のいずれかを指すことがある。

40

【0122】

[0122]本開示の第1の例示的なNBDV導出プロセスとして、ビデオデコーダ30は、空間又は時間隣接ブロックがBVSPモードでコード化されるかどうかを決定するように構成される。このチェックは、NBDV導出プロセスのために定義された各空間又は時間隣接ブロックに対して実行される。隣接ブロックがBVSPモードでコード化される場合、ビデオデコーダ30は、隣接ブロックが現在のピクチャ中に位置するか、異なるピクチャ

50

ャ中に位置するかにかかわらず、視差動きベクトルとして、B V S Pモードでコード化された隣接ブロックに属する動きベクトルを指定する。言い換えれば、ビデオデコーダ30は、N B D V導出プロセス中に、利用可能な視差動きベクトルとして、B V S Pでコード化された隣接ブロック中の動きベクトルと、ビュー間予測でコード化された隣接ブロック中の動きベクトルの両方を指定する。

【0123】

[0123]本開示の第2の例示的なN B D V導出プロセスでは、米国特許出願第14/189,177号の技法を使用する、1つのMB区分のためのMBレベルN B D V導出プロセスは、隣接ブロック中のB V S Pモードの使用をチェックすることによって改善され得る。

10

【0124】

[0124]第2の例の第1の態様では、ビデオデコーダ30は、以下の2つの下記の条件のいずれかが真であるとき、各時間及び/又は空間隣接ブロックを使用して、N B D V導出を使用して、視差ベクトルを導出するように構成され得、それらの条件は即ち、(1)隣接ブロックがビュー間予測を使用してコード化される場合、又は(2)隣接ブロックがB V S Pモードを使用してコード化される場合である。隣接ブロックがビュー間予測を使用してコード化される場合、ビデオデコーダ30は、現在のブロックのための視差ベクトルとして、隣接ブロックに関連付けられた視差動きベクトルを指定する。隣接ブロックがB V S Pモードを使用してコード化された場合、ビデオデコーダ30は、現在のブロックのための視差ベクトルとして、隣接ブロックの復号中に生成された視差ベクトルを指定する。

20

【0125】

[0125]本開示の第2の例の第2の態様では、ビデオデコーダ30は、2段チェック手順を採用することになり得る。最初に、ビデオデコーダ30は、空間及び/又は時間隣接ブロックのうちの少なくとも1つが、ビュー間予測を使用してコード化されるかどうかをチェックするように構成される。そうでない場合、ビデオデコーダ30は、次いで、空間及び/又は時間隣接ブロックのうちの少なくとも1つが、B V S Pモードを使用してコード化されるかどうかをチェックする。

【0126】

[0126]本開示の第2の例の第3の態様では、上記で説明された2段チェックプロセスが入れ替えられる。即ち、ビデオデコーダ30は、隣接ブロックのうちの少なくとも1つが、B V S Pモードを使用してコード化されるかどうかを、最初にチェックするように構成される。そうでない場合、ビデオデコーダ30は、次いで、全ての空間(及び/又は)時間隣接ブロックのうちの少なくとも1つが、ビュー間予測を使用してコード化されるかどうかをチェックするように構成される。

30

【0127】

[0127]上記で説明された第2の例の態様の各々は、区分レベルN B D V導出アルゴリズム、ならびにMBレベルN B D V導出アルゴリズムに適用され得る。

【0128】

[0128]本開示の第3の例示的なN B D V導出プロセスでは、MBレベルN B D V導出プロセスが、参照深度ビューにアクセスすることを含む追加のステップを、プロセスの最後に追加することによって、更に精密化され得る。幾つかの例では、ビデオデコーダ30は、B V S Pモードを使用してコード化されたブロックに、B V S Pモードを使用してコード化されていない、インター予測されたブロックに、又は、全てのインター予測されたブロックに、この追加のステップを採用するように構成され得る。

40

【0129】

[0129]第3の例の第1の態様では、ビデオデコーダ30は、深度参照ビュー中の1つの深度ブロックから1つの深度値を選択し、その深度値を、更新された視差ベクトルに変換するように構成され得る。この更新された視差ベクトルは、次いで、現在のMB内の全てのMB区分に適用される。幾つかの例では、ビデオデコーダ30は、現在のMB又はMB

50

区分の最終的な視差動きベクトルとして、この更新 / 正確な視差ベクトルを記憶するように更に構成され得る。

【 0 1 3 0 】

[0130]本開示の第 4 の例示的な N B D V 導出プロセスでは、ビデオデコーダ 3 0 は、現在の M B が復号された後、現在の M B のための動きベクトルとして、M B の導出された視差ベクトル（即ち、N B D V 導出を使用して導出された）を記憶するように構成され得る。

【 0 1 3 1 】

[0131]本開示の第 5 の例示的な N B D V 導出プロセスでは、ビデオデコーダ 3 0 は、B V S P モードを使用してコード化された少なくとも 1 つの M B 区分を含む、M B のための追加のメモリを割り振るように構成され得る。この場合、M B の視差ベクトルが記憶され得、M B 区分のいかなる復号された動きベクトルをも上書きする必要はない。

【 0 1 3 2 】

[0132]第 5 の例の第 1 の態様では、少なくとも 1 つの M B 区分が、B V S P モードを使用してコード化される場合に、N B D V 導出プロセスから導出された M B の視差ベクトルを記憶するために、M B ごとに 1 つの追加の動きベクトルが割り振られる。

【 0 1 3 3 】

[0133]第 5 の例の第 2 の態様では、N B D V 導出プロセスを採用するとき、ビデオデコーダ 3 0 は、更に構成され得、隣接ブロックが B V S P モードを使用してコード化される時、現在の M B のための視差ベクトルとして、この隣接ブロックを含んでいる M B に関連付けられた視差ベクトルを使用する。

【 0 1 3 4 】

[0134]第 6 の例では、本開示の N B D V 導出プロセス、ビデオデコーダ 3 0 は、通常のインターモードにおける D - M V P、ならびにスキップモード及び直接モードにおけるビュー間動きベクトル予測など、視差ベクトルに依存するコード化ツールに、上記の例示的な技法のいずれかの組合せを採用するように構成され得る。この例では、改善された N B D V 導出プロセスの結果が、コード化ツールのために使用される。例えば、D - M V P プロセス中に、本開示の改善された N B D V 導出プロセスから生成された、得られた視差ベクトルは、場合によっては、他の N B D V 導出プロセスの結果（例えば、米国特許出願第 1 4 / 1 8 9 , 1 7 7 号中において記載された N B D V 導出プロセスからの結果）に取って代わるために使用した。

【 0 1 3 5 】

[0135]上記の例及び例の態様のいずれもが任意の組合せにおいて一緒に実行され得ることを理解されたい。

【 0 1 3 6 】

[0136]改善された B V S P プロセスもまた、上記で説明されたような改善された N B D V 導出プロセスを使用して、本開示で提案される。但し、改善された B V S P プロセスがより正確な視差ベクトルを生成するという理由のために、N B D V 導出結果もまた同様に改善され得る（即ち、導出された視差ベクトルの精度が改善され得る）。

【 0 1 3 7 】

[0137]本開示の第 1 の例示的な B V S P プロセスでは、B V S P プロセス中に、M B レベル N B D V 導出プロセスと下位領域レベル視差ベクトル精密化プロセスとが組み合わされる。まず、ビデオデコーダ 3 0 は、各 M B のための視差ベクトルを導出するように構成され得る。ビデオデコーダ 3 0 は、上記で説明された改善された N B D V 導出プロセスを使用して、又は、米国特許出願第 1 4 / 1 8 9 , 1 7 7 号に記載されたような N B D V プロセスを用いて、視差ベクトルを導出するように構成され得る。

【 0 1 3 8 】

[0138]次に、ビデオデコーダ 3 0 は、現在の M B 又は M B 区分の 8×8 下位領域ごとに視差ベクトルを精密化するように構成され得る。ビデオデコーダ 3 0 は、B V S P モードでコード化された現在の M B 又は M B 区分の下位領域ごとの動き補償のために、正確な視

10

20

30

40

50

差ベクトルを使用し得る。MB又はMB区分の下位領域ごとの視差ベクトルの精密化は、参照ビューの深度ビューコンポーネントに依存する。一例では、下位領域ごとに、参照ビューの深度コンポーネント中の対応する深度ブロックが、NB DV導出プロセスからの視差ベクトルによって識別される。対応する深度ブロック中の4隅画素の最大値が、正確な視差ベクトルの水平成分に変換される。正確な視差ベクトルの垂直成分は、0に設定される。本明細書では、正確な視差ベクトルは、上記で説明されたように、BVSPモードでコード化された下位領域のための導出された視差ベクトルと同じであることに留意されたい。

【0139】

[0139]第1のBVSP例の第1の態様では、下位領域のサイズは $K \times K$ であってよく、 K は8とは異なり、例えば、 16×16 、 4×4 、 2×2 、又は 1×1 である。第1のBVSP例の第2の態様では、ビデオデコーダ30は、1つのMB区分に対して1回、視差ベクトルを精密化し、又は、 K が8よりも小さいときでも、1つのMB区分内の 8×8 領域ごとに視差ベクトルを精密化するように構成される。

【0140】

[0140]第1のBVSP例の第3の態様では、ビデオデコーダ30は、参照ビューの深度ビューコンポーネント中の(NB DV導出プロセスによって生成された)視差ベクトルによって識別された参照ブロックの1つ又は複数の深度画素から、深度値を選択するように構成され得る。幾つかの例では、ビデオデコーダ30は、(NB DV導出プロセスからの導出された視差ベクトルによって位置を特定される)参照深度ビュー中の識別された深度ブロックの中心内又はその近くのフル深度画素を選択しないように構成され得る。代わりに、一例では、ビデオデコーダ30は、識別された深度ブロックの隅の位置にある深度画素を選択するように構成され得る。

【0141】

[0141]第1のBVSP例の第4の態様では、ビデオデコーダ30は、正確な視差ベクトルの垂直成分のために、NB DV導出プロセスから導出された視差ベクトルの垂直成分を継承するように構成され得る。第1のBVSP例の第5の態様では、ビデオデコーダ30は、正確な視差ベクトルの垂直成分を0に等しくなるように設定するように構成され得る。

【0142】

[0142]本開示の第2の例示的なBVSPプロセスでは、ビデオデコーダ30は、BVSPモードを使用してコード化されたMB区分のために、現在のMBがコード化された後、MB区分のための動きベクトルとして、各下位領域の正確な視差ベクトルを記憶するように構成される。一例では、このプロセスは、MB下位領域のサイズが 4×4 以上である場合に実行される。記憶された視差ベクトルは、上記で説明された本開示のNB DV導出プロセスにおいて使用され得る。

【0143】

[0143]第2のBVSP例の第1の態様では、下位領域の定義されたサイズが 8×8 であるとき(即ち、 K が8に等しいとき)、ビデオデコーダ30は、MB区分サイズが 8×8 に等しいときにのみ、MB区分の動きベクトルとして、正確な視差ベクトルを記憶するように構成され得る。他のMB区分では、ビデオデコーダ30は、現在のMBがコード化された後、視差動きベクトルとして、NB DV導出プロセスからのMBの導出された視差ベクトルを記憶するように構成される。

【0144】

[0144]第2のBVSP例の第2の態様では、ビデオデコーダ30は、ビュー間参照ピクチャを示すインデックスによって、参照ピクチャインデックスを置き換えるように更に構成され得る。

【0145】

[0145]第2のBVSP例の第3の態様では、上記で説明されたBVSP技法の各々は、BVSPモードでコード化されたMB区分がコード化されると、直接適用され得る。即ち

、例えば、ビデオデコーダ 30 は、MB 全体が完全に復号されるまで待機する必要がない。

【0146】

[0146]本開示の第3の例示的なBVSPプロセスでは、ビデオデコーダ30は、BVSPモードを使用してコード化される少なくとも1つのMB区分を含む、MBのための追加のメモリを割り振るように構成され得る。この場合、BVSP区分の動き補償のために使用された各下位領域の正確な視差ベクトルが記憶され得、上書きされる必要がない。

【0147】

[0147]第3のBVSP例の第1の態様では、ビデオデコーダ30は、区分がBVSPモードを使用してコード化される場合、正確な視差ベクトルを記憶するために、1つのMB区分のために、MBにつき最大 $(16/K) * (16/K)$ の追加の動きベクトルを割り振るように構成され得る。

10

【0148】

[0148]第3のBVSP例の第2の態様では、NB DV導出プロセスにおいて、1つの隣接ブロックがBVSPモードを使用してコード化されるとき、この隣接ブロックを含んでいる下位領域に関連付けられた正確な視差ベクトルが、現在のMBのための視差ベクトルとして使用されることになる。

【0149】

[0149]上記の方法の各々はまた、MBがコード化単位(CU)によって置き換えられ、MB区分が予測単位(PU)によって置き換えられて、3D-HEVCにも適用され得る。

20

【0150】

[0150] 1 例示的な実装形態

[0151]本開示の次のセクションは、例示的な実装形態について説明する。テクスチャ優先コード化順序が可能にされる方法で、3D-AVCを構成するとき、BVSPの復号プロセスは、以下のステップを含み得る。代替的に、以下のセクション1.2で説明される方法は、視差ベクトルを必要とする他のコード化ツール、例えば、3D-AVCにおけるD-MVPに適用され得る。

【0151】

[0152]セクション1.1 BVSPモードの例示的な指示

30

[0153]一例では、深度優先コード化順序を使用する3D-AVCのための以前の提案において使用されるような、1つ又は複数のMB区分がBVSPモードを使用してコード化されるといふ指示が、再使用され得る。別の例では、BVSP又は従来のインター予測(時間予測若しくはビュー間予測)が使用されるかどうかを示すために、参照インデックスの代わりに、1つのフラグが、各MB区分中で信号伝達され得る。

【0152】

[0154]セクション1.2 例示的な視差ベクトル導出プロセス

[0155]NB DV導出プロセスについて、以下のサブセクション1.2.1及び1.2.2で説明する。BVSPコード化された区分中の下位領域ごとの動き補償のために使用される、正確な視差ベクトルの生成プロセスについて、以下のサブセクション1.2.3で説明する。

40

【0153】

[0156]NB DV導出プロセスから視差ベクトルを導出するプロセスは、1つのマクロブロックを復号する前に呼び出され得る。別の例では、更に、視差ベクトルを精密化するプロセスが、BVSPモードでコード化されたMB区分に対して呼び出され得る。別の例では、1つのMB区分がBVSPモード及び/又は他のインターモードでコード化されるとき、NB DVから視差ベクトルを導出するプロセスが呼び出され得る。

【0154】

[0157]別の例では、以下のサブセクション1.2.1及び1.2.3の技法が順に実行され、次いで、サブセクション1.2.3の技法から生成された下位領域の正確な視差ベ

50

クトルが、B V S Pコード化されたM B区分の動き補償のために使用される。更に、以下のサブセクション1.2.1の技法から生成された視差ベクトルが、D - M V Pにおいて適用され得る。

【0155】

[0158]別の例では、サブセクション1.2.1、1.2.2及び1.2.3の技法が、順に呼び出される。以下のセクション1.2.2の技法から生成された、更新された視差ベクトルが、次いで、B V S Pモード又は他のインターモードのいずれかでコード化された全てのM B区分に対して使用される。B V S Pモードでコード化されたM B区分に対して、B V S Pコード化されたM B区分の動き補償のために使用される下位領域の正確な視差ベクトルを取得するために、更新された視差ベクトルが次いで使用される。

10

【0156】

[0159]サブセクション1.2.1 深度情報なしに、1つのマクロブロックに対するN B D V導出プロセスから導出された視差ベクトル

[0160]米国特許出願第14/189,177号に記載されているように、M BレベルN B D V導出プロセスは、いかなる深度情報を考慮することもなしに適用され得る。最初に、ビデオデコーダ30は、ある数の時間ピクチャ候補を選択するように構成されてよく、幾つかの予め定義された時間隣接ブロックが、候補ピクチャ中でチェックされ、予め定義された空間隣接ブロックによって後続される。各隣接ブロックについて、各隣接ブロックが異なるビュー、又はR e f P i c L i s t 0中の合成参照コンポーネント（即ち、B V S Pモードを使用してコード化された）のいずれかから予測される場合、隣接ブロックの視差動きベクトル又は関連付けられた導出された視差ベクトル/正確な視差ベクトルは、その導出された視差ベクトル/正確な視差ベクトルの水平成分が0に等しくない場合、現在のM Bの視差ベクトルとして返される。

20

【0157】

[0161]別の例では、ビデオデコーダ30は、任意の順序で、参照ピクチャリスト0（R e f P i c L i s t 0）と参照ピクチャリスト1（R e f P i c L i s t 1）の両方をチェックするように構成され得る。

【0158】

[0162]別の例では、ビデオデコーダ30は、全ての空間（及び/又は）時間隣接ブロックがビュー間予測を使用するかどうか、全ての空間（及び/又は）時間隣接ブロックをチェックするように構成され得る。そうでない場合、ビデオデコーダ30は、隣接ブロックをチェックする異なるラウンドにおいて、隣接ブロックがB V S Pモードを使用してコード化されるかどうかをチェックする。

30

【0159】

[0163]別の例では、ビデオデコーダ30は、全ての隣接ブロックをチェックする最初のラウンドにおいて、隣接ブロックがB V S Pモードを使用してコード化されるかどうかを最初にチェックするように構成される。どのブロックもB V S Pモードを使用してコード化されない場合、ビデオデコーダ30は、次いで、全ての空間（及び/又は）時間隣接ブロックが、ビュー間予測を使用してコード化されるかどうかを決定するために、全ての空間（及び/又は）時間隣接ブロックをチェックする。

40

【0160】

[0164]別の例では、1つの隣接ブロックがビュー間予測又はB V S Pモードのいずれかでコード化されないとき、ビデオデコーダ30は、視差ベクトルとして、利用可能である場合、その隣接ブロックを含んでいるM Bの視差ベクトルを返す。M Bの視差ベクトルは、M B区分のうちの少なくとも1つがB V S Pモードを使用してコード化されるときのみ、利用可能である。

【0161】

[0165]サブセクション1.2.1.1 時間/空間隣接ブロックの選択

[0166]米国特許出願第14/189,177号及び上記に記載されているような同様の方法は、どの時間/空間隣接ブロックをN B D V導出のために使用するべきかの選択のた

50

めに使用され得る。時間ピクチャ中の時間隣接ブロックはまた、B V S Pを用いて予測され得、B V S Pコード化されたM B又はM B区分の視差ベクトルもまた、利用可能であると見なされることに留意されたい。

【 0 1 6 2 】

[0167]サブセクション 1 . 2 . 2 深度情報を用いたM Bレベル視差ベクトル更新プロセス

[0168]ビデオデコーダ 3 0 は、以下の技法を使用して、上記で説明されたN B D V導出プロセスを使用して導出された視差ベクトルを更新するように構成され得る。最初に、M Bのサイズを、 $K \times K$ (但し、 K は16であり得る)として示し、現在のピクチャに対する現在のM B区分の左上位置を、 (x, y) として示し、現在のM Bのためのサブセクション 1 . 2 . 1におけるN B D Vからの導出された視差ベクトルを、 $(DV[0], DV[1])$ として示し、但し、 $DV[0]$ 及び $DV[1]$ は、視差ベクトルの水平成分と垂直成分を示す。1つの深度値(D)は、参照深度ビュー中の4隅画素から選択される。

【 0 1 6 3 】

$D = \max(D0, D1, D2, D3)$

関数 $\max(\cdot)$ は、 D_i (i は0から3である)の最大値を返し、 D_i は、

$i=0: ((x + (DV[0] \gg P)) \gg \text{reduced_resolution_flag}, (y + (DV[1] \gg P)) \gg \text{reduced_resolution_flag})$

$i=1: ((x + (DV[0] \gg P) + K-1) \gg \text{reduced_resolution_flag}, (y + (DV[1] \gg P)) \gg \text{reduced_resolution_flag})$

$i=2: ((x + (DV[0] \gg P)) \gg \text{reduced_resolution_flag}, (y + (DV[1] \gg P) + K-1) \gg \text{reduced_resolution_flag})$

$i=3: ((x + (DV[0] \gg P) + K-1) \gg \text{reduced_resolution_flag}, (y + (DV[1] \gg P) + K-1) \gg \text{reduced_resolution_flag})$

に位置する i 番目の画素値を示す。

【 0 1 6 4 】

1に等しいシンタックス要素 $\text{reduced_resolution_flag}$ は、あるビューコンポーネントペアの深度ビューコンポーネントが同じビューコンポーネントペアのテクスチャビューコンポーネントのルーマ成分よりも低い空間解像度を有すること、ならびに、深度ビューコンポーネントの幅と高さの両方が、全てのテクスチャビューコンポーネントの幅及び高さの半分であることを指定する。0に等しいシンタックス要素 $\text{reduced_resolution_flag}$ は、深度ビューコンポーネントとテクスチャビューコンポーネントの両方が存在するとき、それらが同じ空間解像度を有することを指定する。 P は、視差ベクトルが1/4画素精度であるときは2に、1/2画素精度では1に、整数画素精度では0に等しい、視差ベクトルの精度を示す。

【 0 1 6 5 】

[0169]別の例では、深度値を選択するために、 $\max(D0, D3)$ が使用され得る。

【 0 1 6 6 】

[0170]別の例では、参照深度ビュー中の同一位置配置されたM B内の他の画素が使用され得る。

【 0 1 6 7 】

[0171]ビデオデコーダ 3 0 は、次いで、現在のM B領域内の下位領域のための選択された深度値から、更新された視差ベクトルの水平成分を変換するように構成され得る。更新された視差ベクトルの垂直成分は、ゼロに設定される。

【 0 1 6 8 】

[0172]別の例では、更新された視差ベクトルの垂直成分は、N B D Vからの導出された視差ベクトルの垂直成分に設定され得る。

【 0 1 6 9 】

[0173]別の例では、 K は、8、4又は2に等しくなり得る。

【 0 1 7 0 】

[0174]更新された視差ベクトルは、現在のMB内の全ての区分のために使用され得る。

【0171】

[0175]セクション1.2.3 B V S Pモードでコード化された各MB区分のための視差ベクトル精密化

[0176]ビデオデコーダ30はまた、MB区分がB V S Pモードでコード化される（即ち、合成参照コンポーネントから予測される）場合、各MB区分の下位領域ごとに1つの正確な視差ベクトルを導出するようにも構成され得る。

【0172】

[0177]ビデオデコーダ30は、以下の技法を使用して、上記で説明されたN B D V導出プロセスを使用して導出された視差ベクトルを精密化するように構成され得る。最初に、
下位領域のサイズを、 $K \times K$ （但し、 K は8であり得る）として示し、現在のピクチャに
対する現在のMB区分内の1つの下位領域の左上位置を、 (x, y) として示し、現在の
MBのためのN B D V導出プロセスからの導出された視差ベクトル（又は、サブセクショ
ン1.2.2の技法を実行した後に生成された、更新された視差ベクトル）を、 $(DV[0], DV[1])$
として示し、但し、 $DV[0]$ 及び $DV[1]$ は、視差ベクトルの水平成分と垂直成分とを示す。1つの深度値（ D ）は、参照深度ビュー中の4隅画素から選
択される。

【0173】

$D = \max(D0, D1, D2, D3)$

関数 $\max(\cdot)$ は、 D_i （ i は0から3である）の最大値を返し、 D_i は、

$i=0: ((x + (DV[0] \gg P)) \gg \text{reduced_resolution_flag}, (y + (DV[1] \gg P)) \gg \text{reduced_resolution_flag})$

$i=1: ((x + (DV[0] \gg P) + K-1) \gg \text{reduced_resolution_flag}, (y + (DV[1] \gg P)) \gg \text{reduced_resolution_flag})$

$i=2: ((x + (DV[0] \gg P)) \gg \text{reduced_resolution_flag}, (y + (DV[1] \gg P) + K-1) \gg \text{reduced_resolution_flag})$

$i=3: ((x + (DV[0] \gg P) + K-1) \gg \text{reduced_resolution_flag}, (y + (DV[1] \gg P) + K-1) \gg \text{reduced_resolution_flag})$

に位置する i 番目の画素値を示す。

【0174】

[0178]このサブセクションでは、シンタックス要素 $\text{reduced_resolution_flag}$ 及び P は、セクション1.2.2と同様に定義される。

【0175】

[0179]別の例では、深度値を選択するために、 $\max(D0, D3)$ が使用され得る。

【0176】

[0180]別の例では、参照深度ビュー中の同一位置配置されたMB内の他の画素が使用され得る。

【0177】

[0181]ビデオデコーダ30は、現在のMB領域内の下位領域のための選択された深度値から、正確な視差ベクトルの水平成分を変換するように更に構成され得る。正確な視差ベクトルの垂直成分は、0に設定される。別の例では、正確な視差ベクトルの垂直成分は、N B D Vからの導出された視差ベクトルの垂直成分に設定され得る。

【0178】

[0182]別の例では、 K は、4、2又は1に等しくなり得る。

【0179】

[0183]セクション1.3 B V S Pモードでコード化された1つのMB区分の予測

[0184]B V S Pモードでコード化された現在のMB区分内の下位領域ごとに、ビデオデコーダ30は、参照テクスチャビュー中の予測ブロックを取得するために、正確な視差ベクトルを使用するように構成され得る。別の例では、B V S Pモードでコード化されたMB区分ごとに、ビデオデコーダ30は、参照テクスチャビュー中の予測ブロックを取得す

10

20

30

40

50

るために、NBDV導出プロセスからの導出された視差ベクトルを使用するように構成され得る。現在のMB区分の残差ブロックと予測ブロックとが一緒に、MB区分を再構成するために使用される。

【0180】

[0185] 1. 4 BVS Pモードでコード化されたMB区分のための動きベクトル割当て

[0186] 非ベーステクスチャビュー中の1つのMBが完全に復号された後、ビデオデコーダ30は、サブセクション1. 2. 3において説明されたように、正確な視差ベクトルを記憶することができ、BVS Pモードを使用してコード化されるMB区分の下位領域ごとの動きベクトルとして記憶される。

【0181】

[0187] 別の例では、非ベーステクスチャビュー中の1つのMBが完全に復号された後、ビデオデコーダ30は、BVS Pモードを使用してコード化される全てのMB区分のための動きベクトルとして、サブセクション1. 2. 1において説明されたようなNBDV導出プロセスから導出された視差ベクトル、又は、サブセクション1. 2. 2において説明されたように深度情報が考慮に入れられた、更新された視差ベクトルを記憶し得る。

【0182】

[0188] 別の例では、非ベーステクスチャビュー中のBVS Pモードでコード化された1つのMB区分が復号された後、ビデオデコーダ30は、このMB区分のための動きベクトルとして、説明されたサブセクション1. 2. 1のようなMBレベルNBDV導出プロセスからの導出された視差ベクトル、又は、サブセクション1. 2. 2において説明されたように深度情報が考慮に入れられた、更新された視差ベクトルを直接記憶し得る。

【0183】

[0189] 別の例では、非ベーステクスチャビュー中のBVS Pモードでコード化された1つのMB区分が復号された後、ビデオデコーダ30は、この下位領域のための動きベクトルとして、サブセクション1. 2. 3において説明されたような、下位領域ごとの正確な視差ベクトルを直接記憶し得る。

【0184】

[0190] 別の例では、サブセクション1. 2. 2において説明されたように（可能な精密化とともに）NBDV導出プロセスからの導出された視差ベクトルを記憶するか、又は、BVS Pコード化された区分のための動きベクトルとして、正確な視差ベクトルを記憶するのではなく、ビデオデコーダ30は、この情報を記憶するために追加のメモリを割り振り得る。

【0185】

[0191] 別の例では、ビデオデコーダ30は、現在のMBがBVS Pモードでコード化された少なくとも1つの区分を有する場合、サブセクション1. 2. 2において説明されたように（可能な精密化とともに）NBDV導出プロセスからの導出された視差ベクトルを記憶するために、MBごとに1つの動きベクトルを割り振る。NBDV導出プロセス中に、1つの隣接ブロックがBVS Pモードを使用するとき、この隣接ブロックを含んでいるMBに関連付けられた視差ベクトルが、現在のMBのための視差ベクトルとして使用されることになる。

【0186】

[0192] 別の例では、下位領域のサイズが 8×8 に等しいとき、下位領域ごとの正確な視差ベクトルを記憶するために、4つの動きベクトルがMBごとに割り振られる。NBDV導出プロセス中に、1つの隣接ブロックがBVS Pモードを使用するとき、ビデオデコーダ30は、現在のMBのための視差ベクトルとして、この隣接ブロックを含んでいる下位領域に関連付けられた視差ベクトルを使用する。

【0187】

[0193] 図6は、本開示で説明される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である。例えば、図6は、ビデオスライス内のビデオブロックのイントラコード化とインターコード化とを実行することができる、ビデオエンコーダ20を示す。例えば

10

20

30

40

50

、ビデオエンコーダ20は、インター予測符号化又はイントラ予測符号化を実行できる。イントラコード化は、所与のビデオフレーム又はピクチャ内のビデオの空間的冗長性を低減又は除去するために、空間的予測に依拠する。インターコード化は、ビデオシーケンスの隣接フレーム若しくはピクチャ内の時間的冗長性、又は、異なるビュー中のピクチャ間の冗長性を低減又は除去するために、時間的予測又はビュー間予測に依拠する。イントラモード（Iモード）は、幾つかの空間ベースの圧縮モードのいずれかを指すことがある。単方向予測（Pモード）又は双方向予測（Bモード）などのインターモードは、幾つかの時間ベースの圧縮モードのいずれかを指すことがある。

【0188】

[0194]図6の例では、ビデオエンコーダ20は、ビデオデータメモリ40と、予測処理ユニット42と、参照ピクチャメモリ64と、加算器50と、変換処理ユニット52と、量子化処理ユニット54と、エントロピー符号化ユニット56とを含む。予測処理ユニット42は、動き推定ユニット44と、動き補償ユニット46と、イントラ予測ユニット48とを含む。ビデオブロック再構成のために、ビデオエンコーダ20はまた、逆量子化処理ユニット58と、逆変換処理ユニット60と、加算器62とを含む。再構成されたビデオからブロックネスアーティファクトを除去するためにブロック境界をフィルタ処理するために、デブロッキングフィルタ（図6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62の出力をフィルタ処理することになる。デブロッキングフィルタに加えて、（ループ内又はループ後の）追加ループフィルタも使用され得る。

【0189】

[0195]ビデオデータメモリ40は、ビデオエンコーダ20の構成要素によって符号化されるべきビデオデータを記憶し得る。ビデオデータメモリ40に記憶されたビデオデータは、例えば、ビデオ発信源18から取得され得る。参照ピクチャメモリ64は、（例えば、イントラ予測コード化モード又はインター予測コード化モードとも呼ばれる、イントラコード化モード又はインターコード化モードで）ビデオエンコーダ20によってビデオデータを符号化する際に使用するための、参照ビデオデータを記憶する復号ピクチャバッファ（DPBの一例である。ビデオデータメモリ40及び参照ピクチャメモリ64は、同期DRAM（SDRAM）、磁気抵抗RAM（MRAM）、抵抗RAM（RRAM（登録商標））、又は他のタイプの記憶装置を含む、ダイナミックランダムアクセスメモリ（DRAM）など、様々な記憶装置のいずれかによって形成され得る。ビデオデータメモリ40及び参照ピクチャメモリ64は、同じ記憶装置又は別個の記憶装置によって提供され得る。様々な例では、ビデオデータメモリ40は、ビデオエンコーダ20の他の構成要素とともにオンチップであるか、又はそれらの構成要素に対してオフチップであり得る。

【0190】

[0196]ビデオエンコーダ20は、ビデオデータを受信し、区分ユニット（図示せず）は、データをビデオブロックに区分する。この区分は、スライス、タイル、又は他のより大きいユニットへの区分、ならびにビデオブロック区分（例えば、マクロブロック区分、及び区分のサブブロック）をも含み得る。ビデオエンコーダ20は、概して、符号化されるべきビデオスライス内のビデオブロックを符号化する構成要素を示している。スライスは、複数のビデオブロックに（及び、場合によっては、タイルと呼ばれるビデオブロックのセットに）分割され得る。予測処理ユニット42は、誤差結果（例えば、コード化レート及びひずみレベル）に基づいて現在のビデオブロックのために、複数のイントラコード化モード（イントラ予測コード化モード）のうちの1つ、又は複数のインターコード化モード（インター予測コード化モード）のうちの1つなど、複数の可能なコード化モードのうちの1つを選択し得る。予測処理ユニット42は、得られたイントラコード化されたブロック又はインターコード化されたブロックを、残差ブロックデータを生成するために加算器50に与え、参照ピクチャとして使用するための符号化されたブロックを再構成するために加算器62に与え得る。

【0191】

[0197]予測処理ユニット42内のイントラ予測ユニット48は、空間圧縮を行うために、コード化されるべき現在のブロックと同じフレーム又はスライス中の1つ又は複数の隣接ブロックに対する現在のビデオブロックのイントラ予測コード化を実行し得る。予測処理ユニット42内の動き推定ユニット44及び動き補償ユニット46は、時間圧縮を行うために、1つ又は複数の参照ピクチャ中の1つ又は複数の予測ブロックに対する現在のビデオブロックのインター予測コード化を実行する。

【0192】

[0198]動き推定ユニット44は、ビデオシーケンスの所定のパターンに従ってビデオスライスのためのインター予測モードを決定するように構成され得る。所定のパターンは、シーケンス中のビデオスライスをPスライス又はBスライスに指定し得る。動き推定ユニット44と動き補償ユニット46とは、高度に統合され得るが、概念的な目的のために別々に示されている。動き推定ユニット44によって実行される動き推定は、ビデオブロック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。動きベクトルは、例えば、参照ピクチャ内の予測ブロックに対する、現在のビデオフレーム又はピクチャ内のビデオブロックの変位を示し得る。

【0193】

[0199]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)、又は他の差分メトリックによって決定され得る画素差分に関して、コード化されるべきビデオ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。幾つかの例では、ビデオエンコーダ20は、参照ピクチャメモリ64に記憶された参照ピクチャのサブ整数画素位置の値を計算し得る。例えば、ビデオエンコーダ20は、参照ピクチャの1/4画素位置、1/8画素位置、又は他の分数画素位置の値を補間し得る。従って、動き推定ユニット44は、フル画素位置と分数画素位置とに対する動き探索を実行し、分数画素精度で動きベクトルを出力し得る。

【0194】

[0200]動き推定ユニット44は、ビデオブロック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、インターコード化された(インター予測コード化された)スライスにおけるビデオブロック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(RefPicList0)又は第2の参照ピクチャリスト(RefPicList1)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つ又は複数の参照ピクチャを識別する。動き推定ユニット44は、エントロピー符号化ユニット56と動き補償ユニット46とに計算された動きベクトルを送る。

【0195】

[0201]動き補償ユニット46によって実行される動き補償は、動き推定によって決定された動きベクトルに基づいて予測ブロックをフェッチ又は生成すること、場合によってはサブ画素精度への補間を実行することを伴い得る。現在のビデオブロックの動きベクトルを受信すると、動き補償ユニット46は、動きベクトルが参照ピクチャリストのうちの1つにおいて指す予測ブロックの位置を特定し得る。ビデオエンコーダ20は、コード化されている現在のビデオブロックの画素値から予測ブロックの画素値を減算し、画素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。画素差分値は、ブロックの残差データを形成し、ルーマ差分成分とクロマ差分成分の両方を含み得る。加算器50は、この減算演算を実行する1つ又は複数の構成要素を表す。動き補償ユニット46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が使用するのための、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。

【0196】

[0202]イントラ予測ユニット48は、上記で説明されたように、動き推定ユニット44と動き補償ユニット46とによって実行されるインター予測の代替として、現在のブロックをイントラ予測し得る。特に、イントラ予測ユニット48は、現在のブロックを符号化するために使用するべきイントラ予測モードを決定し得る。幾つかの例では、イントラ予

10

20

30

40

50

測ユニット48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48（又は、幾つかの例では、モード選択ユニット）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。例えば、イントラ予測ユニット48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することができる。レートひずみ分析は、概して、符号化されたブロックと、符号化された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（又は誤差）の量、ならびに符号化されたブロックを生成するために使用されるビットレート（即ち、ビット数）を決定する。イントラ予測ユニット48は、どのイントラ予測モードがブロックについて最良のレートひずみ値を呈するかを決定するために、様々な符号化されたブロックのひずみ及びレートから比率を計算し得る。

10

【0197】

[0203]いずれの場合も、ブロック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48は、ブロックについて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56に提供し得る。エントロピー符号化ユニット56は、本開示の技法に従って、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20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及び複数の修正されたイントラ予測モードインデックステーブル（コードワードマッピングテーブルとも呼ばれる）と、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のために使用すべき、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、及び修正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。

20

【0198】

[0204]予測処理ユニット42が、インター予測又はイントラ予測のいずれかを介して、現在のビデオブロックのための予測ブロックを生成した後、ビデオエンコーダ20は、現在のビデオブロックから予測ブロックを減算することによって残差ビデオブロックを形成する。残差ブロックにおける残差ビデオデータは、変換処理ユニット52に適用され得る。変換処理ユニット52は、離散コサイン変換（DCT）又は概念的に同様の変換のような変換を使用して、残差ビデオデータを残差変換係数に変換する。変換処理ユニット52は、残差ビデオデータを画素領域から周波数領域などの変換領域に変換し得る。

30

【0199】

[0205]変換処理ユニット52は、得られた変換係数を量子化処理ユニット54に送り得る。量子化処理ユニット54は、ビットレートを更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、係数の一部又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。幾つかの例では、量子化処理ユニット54は、次いで、量子化変換係数を含む行列の走査を実行し得る。代替的に、エントロピー符号化ユニット56が走査を実行し得る。

【0200】

[0206]量子化の後、エントロピー符号化ユニット56は量子化変換係数をエントロピー符号化する。例えば、エントロピー符号化ユニット56は、コンテキスト適応型可変長コード化（CAVLC）、コンテキスト適応型バイナリ算術コード化（CABAC）、シンタックススペースコンテキスト適応型バイナリ算術コード化（SBAC）、確率間隔区分エントロピー（PIPE）コード化又は別のエントロピー符号化方法若しくは技法を実行し得る。エントロピー符号化ユニット56によるエントロピー符号化の後、符号化されたビットストリームは、ビデオデコーダ30に送信されるか、又はビデオデコーダ30が後で送信するか若しくは取り出すためにアーカイブされ得る。エントロピー符号化ユニット56はまた、コード化されている現在のビデオスライスのための動きベクトルと他のシンタックス要素とをエントロピー符号化することができる。

40

【0201】

50

[0207]逆量子化処理ユニット58及び逆変換処理ユニット60は、参照ピクチャの参照ブロックとして後で使用する目的で画素領域において残差ブロックを再構成するために、それぞれ逆量子化及び逆変換を適用する。動き補償ユニット46は、残差ブロックを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46はまた、動き推定において使用するためのサブ整数画素値を計算するために、再構成された残差ブロックに1つ又は複数の補間フィルタを適用し得る。加算器62は、参照ピクチャメモリ64に記憶するための参照ブロックを生成するために、再構成された残差ブロックを動き補償ユニット46によって生成された動き補償予測ブロックに加算する。参照ブロックは、後続のビデオフレーム又はピクチャ中のブロックをインター予測するために、動き推定ユニット44と動き補償ユニット46とによって参照ブロックとして使用され得る。

10

【0202】

[0208]このようにして、ビデオエンコーダ20は、本開示で説明される1つ又は複数の例示的な技法を実装するように構成されるビデオエンコーダの一例である。例えば、ビデオデータメモリ40は、ビデオデータを記憶する。ビデオデータは、依存ビューのテクスチャビデオコンポーネントと、そのテクスチャビューコンポーネントに対応する深度ビューコンポーネントとを含んでよく、その各々を、ビデオエンコーダ20は、3D-AVC準拠ビデオコード化プロセスにおいて符号化することになる。

【0203】

[0209]本開示で説明される技法では、ビデオエンコーダ20は、3D-AVC準拠ビデオコード化プロセスにおいて、ビデオデータの依存ビューのテクスチャビューコンポーネントを符号化するように構成される、1つ又は複数のプロセッサを含み得る。上記で説明されたように、3D-AVCにおける各ビューは、テクスチャビューコンポーネントと深度ビューコンポーネントとを含む。3D-AVCにおいて、1つのベースビューと、1つ又は複数の拡張ビュー又は依存ビューとがあり、但し、1つ又は複数の拡張ビュー又は依存ビューのテクスチャビューコンポーネントは、ビュー間予測され得る。

20

【0204】

[0210]テクスチャビューコンポーネントを符号化するために、ビデオエンコーダ20は、少なくとも1つの隣接ブロックが、依存ビュー以外のビュー中のビュー間参照ピクチャを参照する視差動きベクトルを用いてビュー間予測されるかどうかを決定するために、テクスチャビューコンポーネント中の現在のブロックの1つ又は複数の隣接ブロックの動き情報を評価するように構成され得る。ビデオエンコーダ20は、隣接ブロックのうちの1つのための視差動きベクトルに基づいて、現在のブロックのための視差ベクトルを導出し得る。テクスチャ優先コード化のために、ビデオエンコーダ20は、テクスチャビューコンポーネントを符号化することに続いて、テクスチャビューコンポーネントに対応する、ビデオデータの深度ビューコンポーネントを符号化し得る。

30

【0205】

[0211]幾つかの例では、ビデオエンコーダ20の予測処理ユニット42は、NB-DV導出及びBVSPコード化のための本開示で説明される例を実装するように構成されたプロセッサの一例であり得る。幾つかの例では、予測処理ユニット42以外のユニット（例えば、1つ又は複数のプロセッサ）が、上記で説明された例を実装することができる。幾つかの例では、予測処理ユニット42は、ビデオエンコーダ20の1つ又は複数の他のユニットとともに、上記で説明された例を実装することができる。幾つかの例では、ビデオエンコーダ20のプロセッサ（図6には図示せず）は、単独で、又はビデオエンコーダ20の他のプロセッサとともに、上記で説明された例を実装することができる。

40

【0206】

[0212]図7は、本開示で説明される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である。例えば、ビデオデコーダ30は、インター予測復号又はイントラ予測復号を実行することができる。図7は、ビデオデコーダ30を示す。図7の例では、ビデオデコーダ30は、ビデオデータメモリ69と、エンтроピー復号ユニット70と、予測処理

50

ユニット 71 と、逆量子化処理ユニット 76 と、逆変換処理ユニット 78 と、加算器 80 と、参照ピクチャメモリ 82 とを含む。予測処理ユニット 71 は、動き補償ユニット 72 と、イントラ予測ユニット 74 とを含む。ビデオデコーダ 30 は、幾つかの例では、図 6 のビデオエンコーダ 20 に関して説明された符号化パスとは概して逆の復号パスを実行し得る。

【0207】

[0213] ビデオデータメモリ 69 は、ビデオエンコーダ 30 の構成要素によって復号されるべき、符号化されたビデオビットストリームなどのビデオデータを記憶し得る。ビデオデータメモリ 69 に記憶されたビデオデータは、例えば、記憶装置 34 から、カメラなどのローカルビデオ発信源から、ビデオデータのワイヤード若しくはワイヤレスネットワーク通信を介して、又は物理データ記憶媒体にアクセスすることによって取得され得る。ビデオデータメモリ 69 は、符号化されたビデオビットストリームからの符号化されたビデオデータを記憶する、コード化されたピクチャバッファ (CPB) を形成し得る。

10

【0208】

[0214] 参照ピクチャメモリ 82 は、(例えば、イントラコード化モード又はインターコード化モードで) ビデオデコーダ 30 によってビデオデータを復号する際に使用するための、参照ビデオデータを記憶する、復号されたピクチャバッファ (DPB) の一例である。ビデオデータメモリ 69 及び参照ピクチャメモリ 82 は、同期 DRAM (SDRAM)、磁気抵抗 RAM (MRAM)、抵抗 RAM (RRAM)、又は他のタイプの記憶装置を含む、ダイナミックランダムアクセスメモリ (DRAM) など、様々な記憶装置のいずれかによって形成され得る。ビデオデータメモリ 69 及び参照ピクチャメモリ 82 は、同じ記憶装置又は別個の記憶装置によって提供され得る。様々な例では、ビデオデータメモリ 69 は、ビデオデコーダ 30 の他の構成要素とともにオンチップであるか、又はそれらの構成要素に対してオフチップであり得る。

20

【0209】

[0215] 復号プロセス中に、ビデオデコーダ 30 は、ビデオエンコーダ 20 から、符号化されたビデオスライスのビデオブロックと、関連付けられるシンタックス要素とを表す、符号化されたビデオビットストリームを受信する。ビデオデコーダ 30 のエンтроピー復号ユニット 70 は、量子化係数と、動きベクトルと、他のシンタックス要素とを生成するために、ビットストリームをエンтроピー復号する。エンтроピー復号ユニット 70 は、動きベクトルと他のシンタックス要素とを予測処理ユニット 71 に転送する。ビデオデコーダ 30 は、ビデオスライスレベル及び / 又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。

30

【0210】

[0216] ビデオスライスがイントラコード化された (I) スライスとしてコード化されるとき、予測処理ユニット 71 のイントラ予測ユニット 74 は、信号伝達されたイントラ予測モードと、現在のフレーム又はピクチャの、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された (即ち、B、又は P) スライスとしてコード化されるとき、予測処理ユニット 71 の動き補償ユニット 72 は、エンтроピー復号ユニット 70 から受信された動きベクトル及び他のシンタックス要素に基づいて、現在のビデオスライスのビデオブロック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 30 は、参照ピクチャメモリ 82 に記憶された参照ピクチャに基づいて、デフォルト構成技法を使用して参照ピクチャリスト (RefPicList0 及び RefPicList1) を構成し得る。

40

【0211】

[0217] 動き補償ユニット 72 は、動きベクトルと他のシンタックス要素とを解析することによって現在のビデオスライスのビデオブロックについての予測情報を決定し、復号されている現在のビデオブロックのための予測ブロックを生成するために、予測情報を使用

50

する。例えば、動き補償ユニット 72 は、ビデオスライスのビデオブロックをコード化するために使用される予測モード（例えば、イントラ予測又はインター予測）と、インター予測スライスタイプ（例えば、B スライス又は P スライス）と、スライスの参照ピクチャリストのうちの 1 つ又は複数についての構成情報と、スライスの各インター符号化されたビデオブロックのための動きベクトルと、スライスの各インターコード化されたビデオブロックについてのインター予測ステータスと、現在のビデオスライス中のビデオブロックを復号するための他の情報とを決定するために、受信されたシンタックス要素の幾つかを使用する。

【0212】

[0218] 動き補償ユニット 72 はまた、補間フィルタに基づいて補間を実行し得る。動き補償ユニット 72 は、参照ブロックのサブ整数画素の補間値を計算するために、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決定し、予測ブロックを生成するために、その補間フィルタを使用し得る。

【0213】

[0219] 逆量子化処理ユニット 76 は、ビットストリーム中で与えられ、エン트로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）、（即ち、逆量子化（de-quantize））する。逆量子化プロセスは、量子化の程度を決定するために、同様に、適用されるべき逆量子化の程度を決定するために、ビデオスライス中の各ビデオブロックについてビデオエンコーダ 20 によって計算される量子化パラメータを使用することを含み得る。逆変換処理ユニット 78 は、画素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、又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。

【0214】

[0220] 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、逆変換処理ユニット 78 からの残差ブロックを動き補償ユニット 72 によって生成された対応する予測ブロックと加算することによって、復号されたビデオブロックを形成する。加算器 80 は、この加算演算を実行する 1 つ又は複数の構成要素を表す。所望される場合、ブロックネスアーティファクトを除去するために復号されたブロックをフィルタ処理するデブロックフィルタも適用され得る。画素遷移を平滑化するために、又は場合によってはビデオ品質を改善するために、（コード化ループ内又はコード化ループ後のいずれかの）他のループフィルタも使用され得る。次いで、所与のピクチャ中の復号されたビデオブロックは、その後の動き補償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 の表示装置 32 などの表示装置上に後で提示するための、復号されたビデオを記憶する。

【0215】

[0221] このようにして、ビデオデコーダ 30 は、本開示で説明される 1 つ又は複数の例示的な技法を実装するように構成されるビデオデコーダの一例である。例えば、ビデオデータメモリ 69 は、ビデオデータを記憶する。ビデオデータは、そこからビデオデコーダ 30 が依存ビューのテクスチャビデオコンポーネントと、そのテクスチャビューコンポーネントに対応する深度ビューコンポーネントとを復号することができる情報を含んでよく、その各々を、ビデオエンコーダ 20 は、3D-AVC 準拠ビデオコード化プロセスにおいて符号化される。

【0216】

[0222] 本開示で説明される技法では、ビデオデコーダ 30 は、3D-AVC 準拠ビデオコード化プロセスにおいて、ビデオデータの依存ビューのテクスチャビューコンポーネントを復号するように構成される、1 つ又は複数のプロセッサを含み得る。テクスチャビューコンポーネントを復号するために、ビデオデコーダ 30 は、少なくとも 1 つの隣接プロ

ックが、依存ビュー以外のビュー中のビュー間参照ピクチャを参照する視差動きベクトルを用いてビュー間予測されるかどうかを決定するために、テクスチャビューコンポーネント中の現在のブロックの1つ又は複数の隣接ブロックの動き情報を評価するように構成され得る。ビデオエンコーダ30は、隣接ブロックのうちの1つのための視差動きベクトルに基づいて、現在のブロックのための視差ベクトルを導出し得る。テクスチャ優先コード化のために、ビデオエンコーダ30は、テクスチャビューコンポーネントを復号することについて、テクスチャビューコンポーネントに対応する、ビデオデータの深度ビューコンポーネントを復号し得る。

【0217】

[0223]幾つかの例では、ビデオデコーダ30の予測処理ユニット71は、NB DV導出及びB V S Pコード化のための本開示で説明される例を実装するように構成されたプロセッサの一例であり得る。幾つかの例では、予測処理ユニット71以外のユニット（例えば、1つ又は複数のプロセッサ）が、上記で説明された例を実装することができる。幾つかの例では、予測処理ユニット71は、ビデオデコーダ30の1つ又は複数の他のユニットとともに、上記で説明された例を実装することができる。更に幾つかの他の例では、ビデオデコーダ30のプロセッサ（図7には図示せず）は、単独で、又はビデオデコーダ30の他のプロセッサとともに、上記で説明された例を実装することができる。

【0218】

[0224]図9は、本開示の例示的な方法を示すフローチャートである。図9を参照しながら説明される技法は、予測処理ユニット42と予測処理ユニット71とを含む、ビデオエンコーダ20及びビデオデコーダ30の任意の構造的又は機能的要素によって実行され得る。以下の例については、上記で説明されたように、ビデオデコーダのビデオエンコーダのいずれか（例えば、ビデオエンコーダ20及びビデオデコーダ30）のための総称語である、「ビデオコーダ」に関して説明される。

【0219】

[0225]一例として、ビデオコーダは、テクスチャ優先コード化を使用して、ビデオデータをコード化するように構成され得る（900）。加えて、ビデオコーダは、複数の隣接ブロックを使用して、ビデオデータのブロックに対してNB DV導出プロセスを実行するように構成されてよく、ここにおいて、NB DV導出プロセスは、視差ベクトルを導出する。ビデオデコーダは、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ブロックベースビュー合成予測（B V S P）モードでコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定すること（910）と、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ビュー間予測モードを使用してコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定すること（920）とによって、NB DV導出プロセスを実行するように構成され得る。ビデオコーダは、1つ又は複数の利用可能な視差動きベクトルから、視差ベクトルを導出することになる（930）。

【0220】

[0226]本開示の別の例では、ビデオコーダは、ある順序で複数の隣接ブロックをチェックすること、及び、特定の隣接ブロックがB V S Pモードでコード化された場合、又は、特定の隣接ブロックがビュー間予測を使用してコード化された場合、視差ベクトルを導出することによって、NB DV導出プロセスを実行するように構成される。

【0221】

[0227]本開示の別の例では、ビデオコーダは、複数の隣接ブロックのいずれかがビュー間予測を使用してコード化されるかどうかを決定するために、複数の隣接ブロックをチェックすることと、チェックすることに基づいて、複数の隣接ブロックのうちの1つがビュー間予測を使用してコード化される場合、ビュー間予測を使用してコード化された隣接ブロックから、視差ベクトルを導出することと、ビュー間予測を使用してコード化された中の複数の隣接ブロックのいずれもない場合、複数の隣接ブロックのいずれかがB V S Pモ

ードを使用してコード化されるかどうかを決定するために、複数の隣接ブロックをチェックすることと、チェックすることに基づいて、複数の隣接ブロックのうちの1つがB V S Pモードを使用してコード化され、複数の隣接ブロックのいずれもビュー間予測を使用してコード化されない場合、B V S Pモードを使用してコード化された隣接ブロックから、視差ベクトルを導出することとによって、N B D V導出プロセスを実行するように構成される。

【0222】

[0228]本開示の別の例では、ビデオコードは、複数の隣接ブロックのいずれかがB V S Pモードを使用してコード化されるかどうかを決定するために、複数の隣接ブロックをチェックすることと、チェックすることに基づいて、複数の隣接ブロックのうちの1つがB V S Pモードを使用してコード化される場合、B V S Pモードを使用してコード化された隣接ブロックから、視差ベクトルを導出することと、B V S Pモードを使用してコード化された中の複数の隣接ブロックのいずれもない場合、複数の隣接ブロックのいずれかがビュー間予測を使用してコード化されるかどうかを決定するために、複数の隣接ブロックをチェックすることと、チェックすることに基づいて、複数の隣接ブロックのうちの1つがビュー間予測を使用してコード化され、複数の隣接ブロックのいずれもB V S Pモードを使用してコード化されない場合、ビュー間予測を使用してコード化された隣接ブロックから、視差ベクトルを導出することとによって、N B D V導出プロセスを実行するように構成される。

【0223】

[0229]本開示の別の例では、ビデオコードは、深度参照ビュー中の深度ブロックから1つの深度値を選択し、深度値を更新された視差ベクトルに変換することと、更新された視差ベクトルをビデオデータのブロックに適用することとによって、N B D V導出プロセスを実行するように構成される。

【0224】

[0230]本開示の別の例では、ビデオコードは、ビデオデータのブロックがコード化された後、ビデオデータのブロックのための動きベクトルとして、更新された視差ベクトルを記憶するように構成される。本開示の別の例では、ビデオコードは、導出された視差ベクトルを記憶するために、追加のメモリを割り振るように構成される。

【0225】

[0231]本開示の別の例では、ビデオデータのブロックは、マクロブロックである。本開示の別の例では、ビデオデータのブロックは、マクロブロックのサブ区分又は区分である。本開示の別の例では、ビデオデータのブロックは、コード化ユニット又は予測ユニットである。

【0226】

[0232]本開示の別の例では、ビデオコードは、B V S Pモードと導出された視差ベクトルとを使用して、ビデオデータのブロックをコード化するように構成される。本開示の別の例では、ビデオコードは、深度ベース動きベクトル予測(D - M V P)と導出された視差ベクトルとを使用して、ビデオデータのブロックをコード化するように構成される。

【0227】

[0233]図10は、本開示の別の例示的な方法を示すフローチャートである。図10を参照しながら説明される技法は、予測処理ユニット42と予測処理ユニット71とを含む、ビデオエンコーダ20及びビデオデコーダ30の任意の構造的又は機能的要素によって実行され得る。以下の例については、上記で説明されたように、ビデオデコーダのビデオエンコーダのいずれか(例えば、ビデオエンコーダ20及びビデオデコーダ30)のための総称語である、「ビデオコード」に関して説明される。

【0228】

[0234]一例では、ビデオコードは、ビデオデータのブロック上でB V S Pプロセスを実行するように構成され、B V S Pプロセスは、視差ベクトルを導出するためにN B D V導出プロセスを実行することを備える。この点については、ビデオコードは、1つ又は複数

の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ブロックベースビュー合成予測 (BVS P) モードでコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定すること (1010) と、1つ又は複数の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ビュー間予測モードを使用してコード化された複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定すること (1020) と、1つ又は複数の利用可能な視差動きベクトルから、視差ベクトルを導出すること (1030) とによって、NB D V 導出プロセスを実行するように構成される。ビデオコードは、ビデオデータのブロックの下位領域について、導出された視差ベクトルを精密化すること (1040) と、正確な視差ベクトルを使用して、BVS P を使用してビデオデータのブロックをコード化すること (1050) とを行うように更に構成される。

10

【0229】

[0235] 本開示の別の例では、複数の隣接ブロックのうちの少なくともは、現在のピクチャ内の空間隣接ブロック、又は異なるピクチャ中の時間隣接ブロックである。

【0230】

[0236] 本開示の別の例では、下位領域は、 8×8 下位領域である。本開示の別の例では、下位領域は、 16×16 、 4×4 、 2×2 、及び 1×1 下位領域のうちの1つである。

【0231】

[0237] 本開示の別の例では、ビデオコードは、ビデオデータのブロックの下位領域のために NB D V 導出プロセスによって生成された視差ベクトルによって識別された参照ブロックの1つ又は複数の深度画素から、深度値を選択し、1つ又は複数の深度画素は、参照ブロックの深度ビューコンポーネント中にあるが、視差ベクトルによって位置を特定される参照深度ビュー中の深度ブロックの中心の近接内のフル画素をしないように更に構成される。

20

【0232】

[0238] 本開示の別の例では、ビデオコードは、NB D V 導出プロセスによって生成された視差ベクトルの垂直成分から、正確な視差ベクトルの垂直成分を継承するように更に構成される。本開示の別の例では、ビデオコードは、正確な視差ベクトルの垂直成分をゼロになるように設定するように更に構成される。

【0233】

30

[0239] 本開示の別の例では、ビデオコードは、ビデオデータのブロックの下位領域ごとの正確な視差ベクトルを記憶するように更に構成され、ここにおいて、記憶された正確な視差ベクトルは、別のブロックに対する NB D V 導出プロセスのために使用される。本開示の別の例では、ビデオコードは、ビデオデータのブロックがある予め決定されたサイズよりも大きく、記憶された正確な視差ベクトルが NB D V 導出プロセスにおいて使用されることになる場合、ビデオデータのブロックのための動きベクトルとして、正確な視差ベクトルを記憶するように更に構成される。

【0234】

[0240] 本開示の別の例では、ビデオデータのブロックは、マクロブロックである。本開示の別の例では、ビデオデータのブロックは、マクロブロックのサブ区分又は区分である。本開示の別の例では、ビデオデータのブロックは、コード化ユニット又は予測ユニットである。

40

【0235】

[0241] 1つ又は複数の例では、説明された機能は、ハードウェア、ソフトウェア、ファームウェア、又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つ又は複数の命令又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、又はコンピュータ可読媒体を介して送信され、ハードウェアベースの処理ユニットによって実行され得る。コンピュータ可読媒体は、データ記憶媒体などの有形媒体に対応するコンピュータ可読記憶媒体、又は、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通信媒体を含み得

50

る。このようにして、コンピュータ可読媒体は、一般に、(1) 非一時的である有形コンピュータ可読記憶媒体、又は、(2) 信号若しく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、本開示で説明される技法を実装するための命令、コード及び/又はデータ構造を取り出すために、1つ若しくは複数のコンピュータ、又は1つ若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。

【0236】

[0242] 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 又は他の光ディスク記憶装置、磁気ディスクストレージ、又は他の磁気記憶装置、フラッシュメモリ、若しくは、命令又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得る。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)、又は赤外線、無線、及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、又は他のリモート発信源から送信される場合には、同軸ケーブル、光ファイバーケーブル、ツイストペア、DSL、又は赤外線、無線、及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及びデータ記憶媒体は、接続、搬送波、信号、又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。本明細書で使用されるディスク(disk)及びディスク(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)及びBlu-ray(登録商標)ディスク(disc)を含み、一方ディスク(disk)は、通常、データを磁氣的に再生し、ディスク(disc)は、データをレーザー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。

【0237】

[0243] 命令は、1つ又は複数のデジタル信号プロセッサ(DSP)のような1つ又は複数のプロセッサ、汎用マイクロプロセッサ、特定用途向け集積回路(ASIC)、フィールドプログラマブル論理アレイ(FPGA)、又は他の等価な集積回路若しくはディスクリット論理回路によって実行され得る。従って、本明細書で使用される「プロセッサ」という用語は、前述の構造、又は本明細書で説明された技法の実装に適した任意の他の構造のいずれかを指し得る。加えて、幾つかの態様では、本明細書に記載された機能は、符号化及び復号のために構成された専用のハードウェアモジュール及び/若しくはソフトウェアモジュール内に設けられる場合があるか、又は複合コーデックに組み込まれる場合がある。また、本技法は、1つ又は複数の回路又は論理要素において完全に実装され得る。

【0238】

[0244] 本開示の技法は、ワイヤレスハンドセット、集積回路(IC)、若しくはICのセット(例えば、チップセット)を含む、多種多様な機器又は装置において実装され得る。本開示では、開示される技法を実行するように構成された機器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、又はユニットが説明されたが、それらの構成要素、モジュール、又はユニットは、必ずしも異なるハードウェアユニットによって実現する必要があるとは限らない。むしろ、前述のように、適切なソフトウェア及び/又はファームウェアとともに、様々なユニットがコーデックハードウェアユニットにおいて組み合わせられ得るか、又は前述のような1つ若しくは複数のプロセッサを含む、相互動作可能なハードウェアユニットの集合体によって設けられ得る。

【0239】

[0245] 様々な例について説明した。これら及び他の例は、以下の特許請求の範囲内である。

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。

[C1]

10

20

30

40

50

ビデオデータをコード化する方法であって、
テクスチャ優先コード化を使用して、ビデオデータをコード化することと、
複数の隣接ブロックを使用して、前記ビデオデータのブロックのための、隣接ブロック
ベース視差ベクトル (N B D V) 導出プロセスを実行することと、ここにおいて、前記 N
B D V 導出プロセスは視差ベクトルを導出する、を備え、ここにおいて、前記 N B D V 導
出プロセスを実行することは、

1 つ以上の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、
ブロックベースビュー合成予測 (B V S P) モードでコード化された前記複数の隣接ブ
ロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、

前記 1 つ以上の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとし
て、ビュー間予測モードを使用してコード化された前記複数の隣接ブロックのうちの隣接
ブロックに関連付けられた動きベクトルを指定することと
を備える、方法。

[C 2]

前記 N B D V 導出プロセスを実行することが、
前記 1 つ以上の利用可能な視差動きベクトルから、前記視差ベクトルを導出すること
を更に備える、C 1 に記載の方法。

[C 3]

前記 N B D V 導出プロセスを実行することが、
ある順序で前記複数の隣接ブロックをチェックすることと、
特定の隣接ブロックが B V S P モードでコード化された場合、又は、特定の隣接ブロッ
クがビュー間予測を使用してコード化された場合、前記視差ベクトルを導出することと
を更に備える、C 1 に記載の方法。

[C 4]

前記 N B D V 導出プロセスを実行することが、
前記複数の隣接ブロックのいずれかがビュー間予測を使用してコード化されるかどうか
を決定するために、前記複数の隣接ブロックをチェックすることと、

前記チェックすることに基づいて、前記複数の隣接ブロックのうちの 1 つがビュー間予
測を使用してコード化される場合、ビュー間予測を使用してコード化された前記隣接ブ
ロックから、前記視差ベクトルを導出することと、

ビュー間予測を使用してコード化された中の前記複数の隣接ブロックのいずれもない場
合、前記複数の隣接ブロックのいずれかが B V S P モードを使用してコード化されるかど
うかを決定するために、前記複数の隣接ブロックをチェックすることと、

前記チェックすることに基づいて、前記複数の隣接ブロックのうちの 1 つが B V S P モ
ードを使用してコード化され、前記複数の隣接ブロックのいずれもビュー間予測を使用し
てコード化されない場合、B V S P モードを使用してコード化された前記隣接ブロックか
ら、前記視差ベクトルを導出することと
を更に備える、C 1 に記載の方法。

[C 5]

前記 N B D V 導出プロセスを実行することが、
前記複数の隣接ブロックのいずれかが B V S P モードを使用してコード化されるかどう
かを決定するために、前記複数の隣接ブロックをチェックすることと、

前記チェックすることに基づいて、前記複数の隣接ブロックのうちの 1 つが B V S P モ
ードを使用してコード化される場合、B V S P モードを使用してコード化された前記隣接
ブロックから、前記視差ベクトルを導出することと、

B V S P モードを使用してコード化された中の前記複数の隣接ブロックのいずれもない
場合、前記複数の隣接ブロックのいずれかがビュー間予測を使用してコード化されるかど
うかを決定するために、前記複数の隣接ブロックをチェックすることと、

前記チェックすることに基づいて、前記複数の隣接ブロックのうちの 1 つがビュー間予
測を使用してコード化され、前記複数の隣接ブロックのいずれも B V S P モードを使用し

10

20

30

40

50

てコード化されない場合、ビュー間予測を使用してコード化された前記隣接ブロックから、前記視差ベクトルを導出することと
を更に備える、C 1 に記載の方法。

[C 6]

前記 N B D V 導出プロセスを実行することが、
深度参照ビュー中の深度ブロックから 1 つの深度値を選択し、前記深度値を更新された視差ベクトルに変換することと、前記更新された視差ベクトルを前記ビデオデータの
ブロックに適用することと
を更に備える、C 2 に記載の方法。

[C 7]

前記ビデオデータの前記ブロックがコード化された後、前記ビデオデータのブロックの
ための動きベクトルとして、前記更新された視差ベクトルを記憶すること
を更に備える、C 6 に記載の方法。

[C 8]

前記導出された視差ベクトルを記憶するために、追加のメモリを割り振ること
を更に備える、C 2 に記載の方法。

[C 9]

前記ビデオデータの前記ブロックが、マクロブロックである、C 1 に記載の方法。

[C 1 0]

前記ビデオデータの前記ブロックが、マクロブロックのサブ区分又は区分である、C 1
に記載の方法。

[C 1 1]

前記ビデオデータの前記ブロックが、コード化単位又は予測単位である、C 1 に記載の
方法。

[C 1 2]

B V S P モードと前記導出された視差ベクトルとを使用して、前記ビデオデータの
ブロックをコード化すること
を更に備える、C 2 に記載の方法。

[C 1 3]

深度ベース動きベクトル予測 (D - M V P) と前記導出された視差ベクトルとを使用し
て、前記ビデオデータの前記ブロックをコード化すること
を更に備える、C 2 に記載の方法。

[C 1 4]

ビデオデータをコード化するように構成された装置であって、前記装置は、
前記ビデオデータのブロックを記憶するように構成されたメモリと、
テクスチャ優先コード化を使用して、ビデオデータをコード化することと、
複数の隣接ブロックを使用して、前記ビデオデータの前記ブロックに対して、隣接ブ
ロックベース視差ベクトル (N B D V) 導出プロセスを実行することと、ここにおいて、
前記 N B D V 導出プロセスは視差ベクトルを導出する

を行うように構成されたビデオコードとを備え、ここにおいて、前記 N B D V 導出プロ
セスを実行することは、

1 つ以上の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして
、ブロックベースビュー合成予測 (B V S P) モードでコード化された前記複数の隣接ブ
ロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、

前記 1 つ以上の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルと
して、ビュー間予測モードを使用してコード化された前記複数の隣接ブロックのうちの隣
接ブロックに関連付けられた動きベクトルを指定することと
を備える、装置。

[C 1 5]

前記ビデオコードが、

10

20

30

40

50

前記 1 つ以上の利用可能な視差動きベクトルから、前記視差ベクトルを導出することを行うように更に構成される、C 1 4 に記載の装置。

[C 1 6]

前記ビデオコードが、
ある順序で前記複数の隣接ブロックをチェックすることと、
特定の隣接ブロックが B V S P モードでコード化された場合、又は、特定の隣接ブロックがビュー間予測を使用してコード化された場合、前記視差ベクトルを導出することと
を行うように更に構成される、C 1 4 に記載の装置。

[C 1 7]

前記ビデオコードが、
前記複数の隣接ブロックのいずれかがビュー間予測を使用してコード化されるかどうかを決定するために、前記複数の隣接ブロックをチェックすることと、
前記チェックすることに基づいて、前記複数の隣接ブロックのうちの 1 つがビュー間予測を使用してコード化される場合、ビュー間予測を使用してコード化された前記隣接ブロックから、前記視差ベクトルを導出することと、

ビュー間予測を使用してコード化された中の前記複数の隣接ブロックのいずれもない場合、前記複数の隣接ブロックのいずれかが B V S P モードを使用してコード化されるかどうかを決定するために、前記複数の隣接ブロックをチェックすることと、

前記チェックすることに基づいて、前記複数の隣接ブロックのうちの 1 つが B V S P モードを使用してコード化され、前記複数の隣接ブロックのいずれもビュー間予測を使用してコード化されない場合、B V S P モードを使用してコード化された前記隣接ブロックから、前記視差ベクトルを導出することと
を行うように更に構成される、C 1 4 に記載の装置。

[C 1 8]

前記ビデオコードが、
前記複数の隣接ブロックのいずれかが B V S P モードを使用してコード化されるかどうかを決定するために、前記複数の隣接ブロックをチェックすることと、

前記チェックすることに基づいて、前記複数の隣接ブロックのうちの 1 つが B V S P モードを使用してコード化される場合、B V S P モードを使用してコード化された前記隣接ブロックから、前記視差ベクトルを導出することと、

B V S P モードを使用してコード化された中の前記複数の隣接ブロックのいずれもない場合、前記複数の隣接ブロックのいずれかがビュー間予測を使用してコード化されるかどうかを決定するために、前記複数の隣接ブロックをチェックすることと、

前記チェックすることに基づいて、前記複数の隣接ブロックのうちの 1 つがビュー間予測を使用してコード化され、前記複数の隣接ブロックのいずれも B V S P モードを使用してコード化されない場合、ビュー間予測を使用してコード化された前記隣接ブロックから、前記視差ベクトルを導出することと
を行うように更に構成される、C 1 4 に記載の装置。

[C 1 9]

前記ビデオコードが、
深度参照ビュー中の深度ブロックから 1 つの深度値を選択し、前記深度値を更新された視差ベクトルに変換することと、前記更新された視差ベクトルを前記ビデオデータの
前記ブロックに適用することと
を行うように更に構成される、C 1 5 に記載の装置。

[C 2 0]

前記ビデオコードが、
前記ビデオデータの
前記ブロックがコード化された後、前記ビデオデータのブロックのための動きベクトルとして、前記更新された視差ベクトルを、前記メモリに記憶すること
を行うように更に構成される、C 1 9 に記載の装置。

[C 2 1]

10

20

30

40

50

- 前記ビデオコーダが、
前記導出された視差ベクトルを記憶するために、追加のメモリを割り振ること
を行うように更に構成される、C 1 5 に記載の装置。
- [C 2 2]
前記ビデオデータの前記ブロックが、マクロブロックである、C 1 4 に記載の装置。
- [C 2 3]
前記ビデオデータの前記ブロックが、マクロブロックのサブ区分又は区分である、C 1
4 に記載の装置。
- [C 2 4]
前記ビデオデータの前記ブロックが、コード化単位又は予測単位である、C 1 4 に記載
の装置。 10
- [C 2 5]
前記ビデオコーダが、
B V S P モードと前記導出された視差ベクトルとを使用して、前記ビデオデータの前記
ブロックをコード化すること
を行うように更に構成される、C 1 5 に記載の装置。
- [C 2 6]
前記ビデオコーダが、
深度ベース動きベクトル予測 (D - M V P) と前記導出された視差ベクトルとを使用し
て、前記ビデオデータの前記ブロックをコード化すること
を行うように更に構成される、C 1 5 に記載の装置。 20
- [C 2 7]
ビデオデータをコード化するように構成された装置であって、前記装置は、
テクスチャ優先コード化を使用して、ビデオデータをコード化するための手段と、
複数の隣接ブロックを使用して、前記ビデオデータのブロックに対して、隣接ブロック
ベース視差ベクトル (N B D V) 導出プロセスを実行すること、ここにおいて、前記 N B
D V 導出プロセスは視差ベクトルを導出する、を行うための手段とを備え、ここにおいて
、前記 N B D V 導出プロセスを実行するための前記手段は、
1 つ以上の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、
ブロックベースビュー合成予測 (B V S P) モードでコード化された前記複数の隣接プロ
ックのうちの隣接ブロックに関連付けられた動きベクトルを指定するための手段と、
前記 1 つ以上の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとし
て、ビュー間予測モードを使用してコード化された前記複数の隣接ブロックのうちの隣接
ブロックに関連付けられた動きベクトルを指定するための手段と
を備える、装置。 30
- [C 2 8]
前記 N B D V 導出プロセスを実行するための前記手段が、
前記 1 つ以上の利用可能な視差動きベクトルから、前記視差ベクトルを導出するための
手段
を更に備える、C 2 7 に記載の装置。 40
- [C 2 9]
実行されたとき、ビデオデータをコード化するように構成された 1 つ以上のプロセッサ
に、
テクスチャ優先コード化を使用して、ビデオデータをコード化することと、
複数の隣接ブロックを使用して、前記ビデオデータの前記ブロックに対して、隣接プロ
ックベース視差ベクトル (N B D V) 導出プロセスを実行することと、ここにおいて、前
記 N B D V 導出プロセスは視差ベクトルを導出する、を行わせる命令を記憶する、コンピ
ュータ可読記憶媒体であって、ここにおいて、前記 N B D V 導出プロセスを実行すること
は、
1 つ以上の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、 50

ブロックベースビュー合成予測（BVSP）モードでコード化された前記複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することと、

前記１つ以上の利用可能な視差動きベクトルのうちの利用可能な視差動きベクトルとして、ビュー間予測モードを使用してコード化された前記複数の隣接ブロックのうちの隣接ブロックに関連付けられた動きベクトルを指定することとを備える、コンピュータ可読記憶媒体。

[C 3 0]

前記命令が、前記１つ以上のプロセッサに、

前記１つ以上の利用可能な視差動きベクトルから、前記視差ベクトルを導出することとを更に行わせる、C 2 9 に記載のコンピュータ可読記憶媒体。

10

【図 1】

図 1

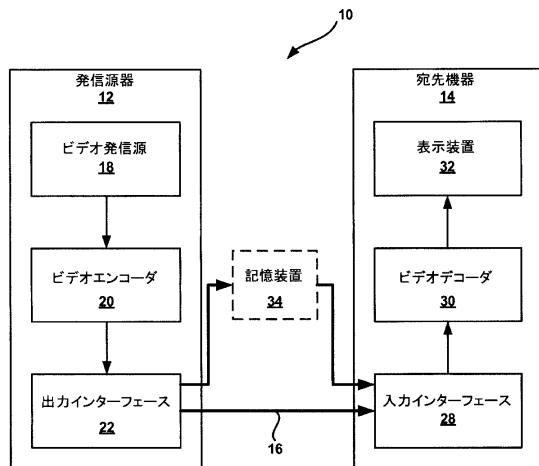


FIG. 1

【図 2】

図 2

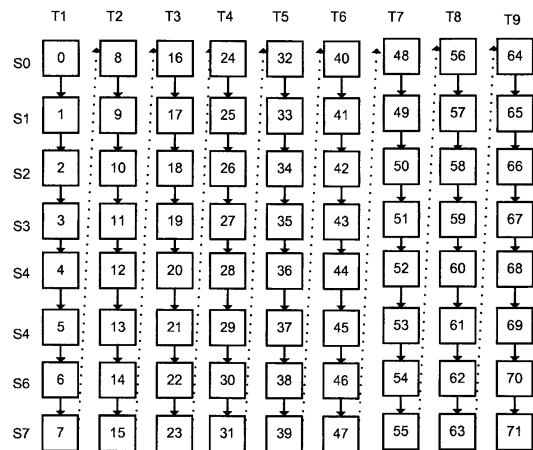
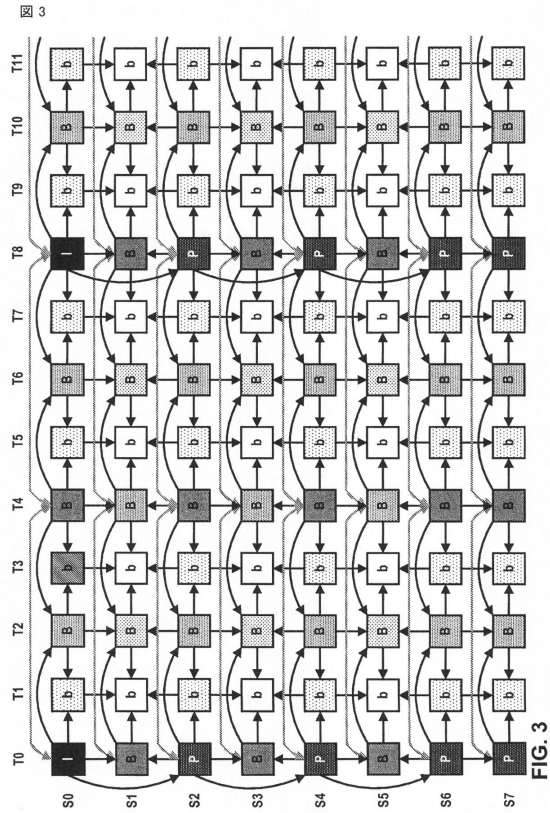
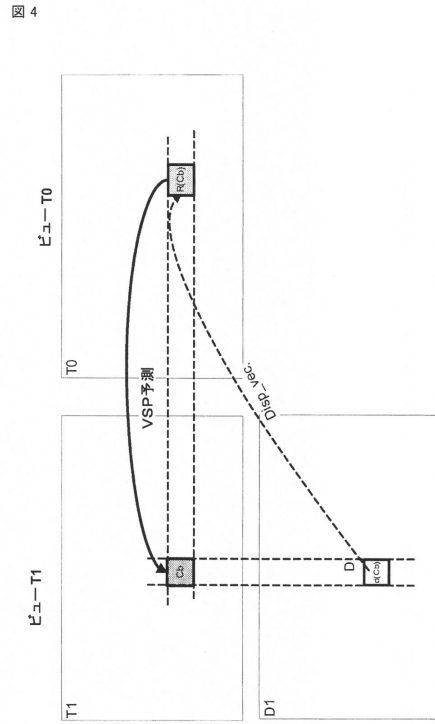


FIG. 2

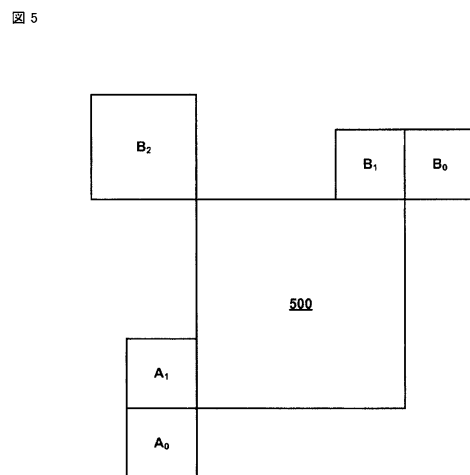
【図 3】



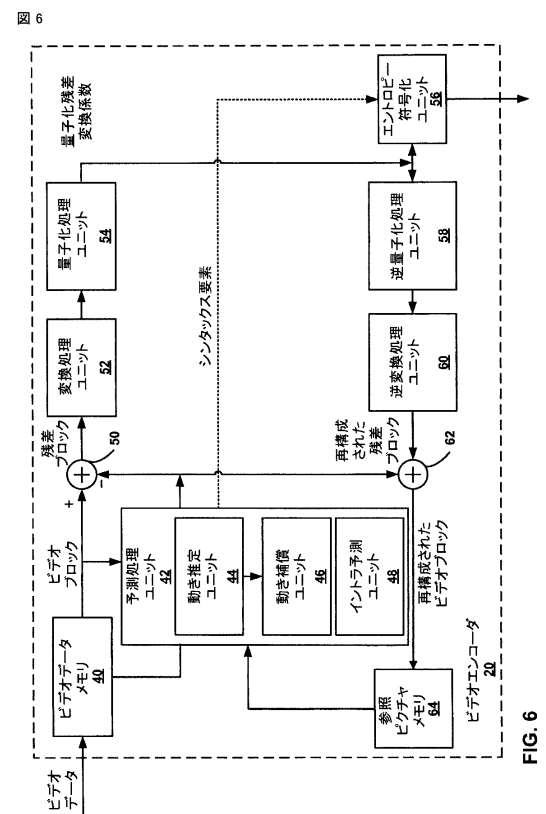
【図 4】



【図 5】



【図 6】



【図 7】

図 7

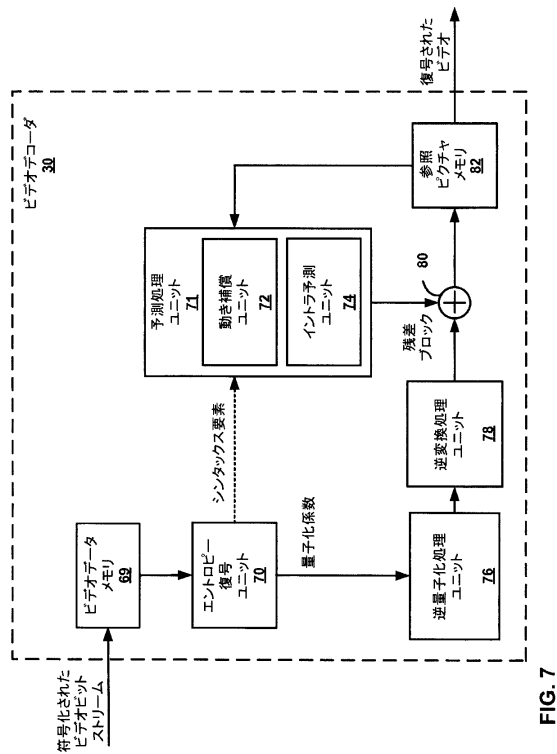


FIG. 7

【図 8】

図 8

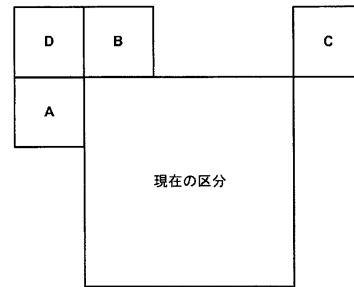


FIG. 8

【図 9】

図 9

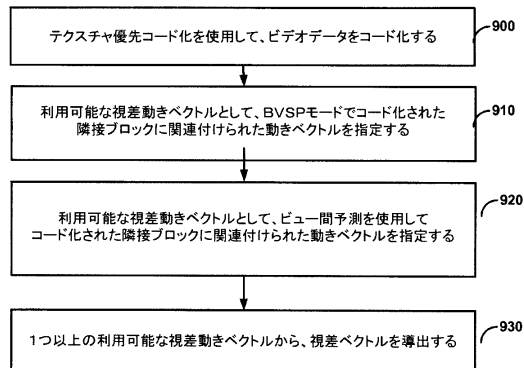


FIG. 9

【図 10】

図 10

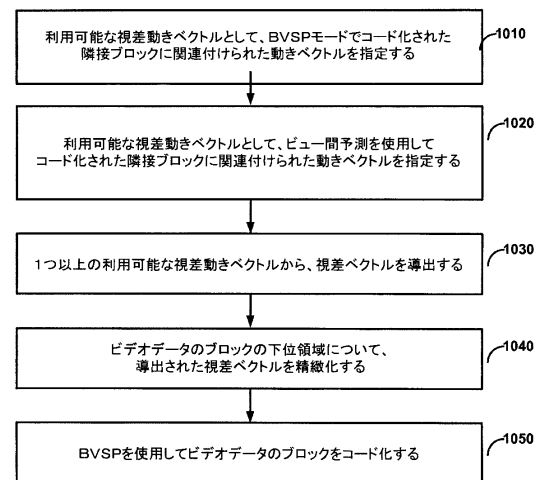


FIG. 10

フロントページの続き

- (31)優先権主張番号 61/770,268
 (32)優先日 平成25年2月27日(2013.2.27)
 (33)優先権主張国 米国(US)
 (31)優先権主張番号 61/772,321
 (32)優先日 平成25年3月4日(2013.3.4)
 (33)優先権主張国 米国(US)
 (31)優先権主張番号 61/803,384
 (32)優先日 平成25年3月19日(2013.3.19)
 (33)優先権主張国 米国(US)
 (31)優先権主張番号 61/815,656
 (32)優先日 平成25年4月24日(2013.4.24)
 (33)優先権主張国 米国(US)
 (31)優先権主張番号 14/189,648
 (32)優先日 平成26年2月25日(2014.2.25)
 (33)優先権主張国 米国(US)

- (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 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 カン、ジェウォン
 大韓民国、1 3 5 - 0 1 0、ソウル、カンナム - グ、ノンヒュン - ドン 2 6 3 - 1 2

審査官 赤穂 州一郎

- (56)参考文献 国際公開第 2 0 1 2 / 1 7 6 6 8 4 (W O , A 1)
 特表 2 0 1 0 - 5 2 2 4 9 4 (J P , A)
 Yu-Lin Chang, Chi-Ling Wu, Yu-Pao Tsai, and Shawmin Lei, 3D-CE5.h related: Depth-oriented Neighboring Block Disparity Vector (DoNBDV) with virtual depth, JCT3V-B0090, Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 2nd Meeting: Shanghai, 2 0 1 2 年 1 0 月 1 1 日, p.1-4.
 Dong Tian, Feng Zou, Anthony Vetro, Title: CE1.h: Backward View Synthesis Prediction using Neighbouring Blocks, JCT3V-C0152, Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 3rd Meeting: Geneva, 2 0 1 3 年 1 月 2 1 日, p.1-5.
 Yu-Lin Chang, Chi-Ling Wu, Yu-Pao Tsai, and Shawmin Lei, CE1.h: Depth-oriented Neighboring Block Disparity Vector (DoNBDV) with virtual depth retrieval, JCT3V-C0131, Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 3rd Meeting: Geneva, 2 0 1 3 年 1 月 1 7 日, p.1-5.
 Jian-Liang Lin et al., 3D-CE5.a related: Simplification on the disparity vector derivation for AVC-based 3D video coding, Joint Collaborative Team on 3D Video Coding Extension Development 1st Meeting: Stockholm, 2 0 1 2 年 7 月 2 0 日, JCT2-A0046

(58)調査した分野(Int.Cl., D B 名)

H 0 4 N 1 9 / 0 0 - 1 9 / 9 8
 H 0 4 N 1 3 / 1 0

