

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7614089号
(P7614089)

(45)発行日 令和7年1月15日(2025.1.15)

(24)登録日 令和7年1月6日(2025.1.6)

(51)国際特許分類

F I

H O 4 N 19/105(2014.01) H O 4 N 19/105

H O 4 N 19/176(2014.01) H O 4 N 19/176

H O 4 N 19/52 (2014.01) H O 4 N 19/52

H O 4 N 19/70 (2014.01) H O 4 N 19/70

請求項の数 19 (全85頁)

(21)出願番号	特願2021-523359(P2021-523359)	(73)特許権者	520476341
(86)(22)出願日	令和1年11月6日(2019.11.6)		北京字節跳動網絡技術有限公司
(65)公表番号	特表2022-506162(P2022-506162 A)		Beijing Bytedance Network Technology Co., Ltd.
(43)公表日	令和4年1月17日(2022.1.17)		中華人民共和國100041北京市石景山区実興大街30号院3号楼2層B-0035房間
(86)国際出願番号	PCT/CN2019/116078		Room B-0035, 2/F, No. 30 Building, No. 30, Shixing Road, Shijingshan District Beijing 100041 China
(87)国際公開番号	WO2020/094078	(73)特許権者	520477474
(87)国際公開日	令和2年5月14日(2020.5.14)		バイトダンス インコーポレイテッド
審査請求日	令和3年4月28日(2021.4.28)		最終頁に続く
審判番号	不服2023-12923(P2023-12923/J 1)		
審判請求日	令和5年8月2日(2023.8.2)		
(31)優先権主張番号	PCT/CN2018/114057		
(32)優先日	平成30年11月6日(2018.11.6)		
(33)優先権主張国・地域又は機関			

(54)【発明の名称】 動き情報の位置依存記憶装置

(57)【特許請求の範囲】

【請求項1】

映像の現在のブロックと前記映像のビットストリームとの間の変換において、前記現在のブロックがジオメトリ分割モードでコーディングされることを判定することと、前記変換は、前記ビットストリームから前記現在のブロックに復号すること、又は、前記現在のブロックを前記ビットストリームに符号化することを含み、

第1の動き情報であるM V I n f o 1および第2の動き情報であるM V I n f o 2を判定することと、

前記M V I n f o 1および前記M V I n f o 2の両方が第1の参照ピクチャリストL Xからのものである場合、第2の参照ピクチャリストL (1-X)における前記M V I n f o 1の参照ピクチャの有無、または、第2の参照ピクチャリストL (1-X)における前記M V I n f o 2の参照ピクチャの有無に関わらず、前記現在のブロックの重み付け領域内にあるm×nサイズのサブブロックに対して第1の単一の動き情報のセットを記憶することと、を含み、前記第1の単一の動き情報のセットは単一予測動き情報を含み、X=0またはX=1である、映像データを処理する方法。

【請求項2】

前記単一予測動き情報を含む前記第1の単一の動き情報のセットは、M V I n f o 1またはM V I n f o 2に基づく、請求項1に記載の方法。

【請求項3】

前記単一予測動き情報を含む前記第1の単一の動き情報のセットは、M V I n f o 1ま

たは $MVInfo2$ に等しい、請求項 1 または 2 に記載の方法。

【請求項 4】

前記記憶された第 1 の単一の動き情報のセットは、時間的動き予測、空間的動き予測、またはフィルタリング処理のうちの少なくとも 1 つに使用される、請求項 1 ～ 3 のいずれか 1 項に記載の方法。

【請求項 5】

前記記憶された第 1 の単一の動き情報のセットは、異なるピクチャにおける他のブロックの処理に用いられる、請求項 1 ～ 3 のいずれか 1 項に記載の方法。

【請求項 6】

$MVInfo1$ が前記第 1 の参照ピクチャリスト LX からのものであり、 $MVInfo2$ が前記第 2 の参照ピクチャリスト $L(1-X)$ からのものであることに応答して、前記現在のブロックの前記重み付け領域内の $m \times n$ サイズのサブブロックに対する第 2 の単一の動き情報のセットを記憶することをさらに含み、前記第 2 の単一の動き情報のセットは双予測動き情報を含み、 $X = 0$ または $X = 1$ である、請求項 1 ～ 5 のいずれか 1 項に記載の方法。

10

【請求項 7】

前記双予測動き情報は、前記 $MVInfo1$ と前記 $MVInfo2$ とを組み合わせることに基づく、請求項 6 に記載の方法。

【請求項 8】

前記双予測動き情報は、前記 $MVInfo1$ および前記 $MVInfo2$ の動きベクトルおよび参照ピクチャインデックスを組み合わせることによって導出される、請求項 7 に記載の方法。

20

【請求項 9】

非重み付け領域内の $m \times n$ サイズの第 3 の単一の動き情報のセットを記憶することをさらに含み、前記第 3 の単一の動き情報のセットは前記 $MVInfo1$ または前記 $MVInfo2$ に基づく単一予測動き情報を含む、請求項 1 ～ 8 のいずれか 1 項に記載の方法。

【請求項 10】

$m = 4$ または 8 であってかつ $n = 4$ または 8 である、請求項 1 ～ 9 のいずれか 1 項に記載の方法。

【請求項 11】

前記ジオメトリ分割モードは複数の分割スキームを含み、少なくとも 1 つの分割スキームは前記現在のブロックを 1 つの分割境界で 2 つの分割に分け、そのうちの少なくとも 1 つが非正方形および非長方形となり、各分割において、前記分割境界の周囲の複数の画素が前記重み付け領域の一部である、請求項 1 ～ 10 のいずれか 1 項に記載の方法。

30

【請求項 12】

右上隅のサンプルと左下隅のサンプルとが 2 つの異なる分割にある場合、 $MVInfo1$ は前記右上隅のサンプルを覆う分割に関連付けられた動き情報のセットを含み、 $MVInfo2$ は前記左下隅のサンプルを覆う分割に関連付けられた動き情報のセットを含む、請求項 11 に記載の方法。

【請求項 13】

前記現在のブロックの前記分割境界の方向は左上隅から右下隅である、請求項 12 に記載の方法。

40

【請求項 14】

左上隅のサンプルと右下隅のサンプルとが 2 つの異なる分割にある場合、 $MVInfo1$ は前記左上隅のサンプルを覆う分割に関連付けられた動き情報のセットを含み、 $MVInfo2$ は前記右下隅のサンプルを覆う分割に関連付けられた動き情報のセットを含む、請求項 11 に記載の方法。

【請求項 15】

前記現在のブロックの前記分割境界の方向は右上隅から左下隅である、請求項 14 に記載の方法。

50

【請求項 16】

MVInfo1およびMVInfo2に基づいて導出された予測サンプルの重み付け和として、前記重み付け領域内の $m \times n$ サイズのサブブロックに対して最終予測を生成するために適用される重み付け処理において、前記重み付け領域内の前記 $m \times n$ サイズのサブブロックの予測サンプルの重み付けはいずれも0に等しくない、請求項1～15のいずれか1項に記載の方法。

【請求項 17】

処理装置と命令を記憶する非一時的メモリを備える映像データを処理する装置であって、前記処理装置によって実行される前記命令は、前記処理装置に、

映像の現在のブロックと前記映像のビットストリームとの間の変換において、前記現在のブロックがジオメトリ分割モードでコーディングされることを判定することと、前記変換は、前記ビットストリームから前記現在のブロックに復号すること、又は、前記現在のブロックを前記ビットストリームに符号化することを含み、

第1の動き情報であるMVInfo1および第2の動き情報であるMVInfo2を判定することと、

前記MVInfo1および前記MVInfo2の両方が第1の参照ピクチャリストLXからのものである場合、第2の参照ピクチャリストL(1-X)における前記MVInfo1の参照ピクチャの有無、または、第2の参照ピクチャリストL(1-X)における前記MVInfo2の参照ピクチャの有無に関わらず、前記現在のブロックの重み付け領域内にある $m \times n$ サイズのサブブロックに対して第1の単一の動き情報のセットを記憶すること、を実行させ、前記第1の単一の動き情報のセットは単一予測動き情報を含み、 $X = 0$ または $X = 1$ である、映像データを処理する装置。

【請求項 18】

処理装置に次のことを実行さ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、前記命令は処理装置に、

映像の現在のブロックと前記映像のビットストリームとの間の変換において、前記現在のブロックがジオメトリ分割モードでコーディングされることを判定することと、前記変換は、前記ビットストリームから前記現在のブロックに復号すること、又は、前記現在のブロックを前記ビットストリームに符号化することを含み、

第1の動き情報であるMVInfo1および第2の動き情報であるMVInfo2を判定することと、

前記MVInfo1および前記MVInfo2の両方が第1の参照ピクチャリストLXからのものである場合、第2の参照ピクチャリストL(1-X)における前記MVInfo1の参照ピクチャの有無、または、第2の参照ピクチャリストL(1-X)における前記MVInfo2の参照ピクチャの有無に関わらず、前記現在のブロックの重み付け領域内にある $m \times n$ サイズのサブブロックに対して第1の単一の動き情報のセットを記憶すること、を実行させ、前記第1の単一の動き情報のセットは単一予測動き情報を含み、 $X = 0$ または $X = 1$ である、非一時的コンピュータ可読記憶媒体。

【請求項 19】

映像の現在のブロックから前記映像のビットストリームへの符号化において、前記現在のブロックがジオメトリ分割モードでコーディングされることを判定することと、

第1の動き情報であるMVInfo1および第2の動き情報であるMVInfo2を判定することと、

前記判定することに基づいてビットストリームを生成することと、

前記MVInfo1および前記MVInfo2の両方が第1の参照ピクチャリストLXからのものである場合、第2の参照ピクチャリストL(1-X)における前記MVInfo1の参照ピクチャの有無、または、第2の参照ピクチャリストL(1-X)における前記MVInfo2の参照ピクチャの有無に関わらず、前記現在のブロックの重み付け領域内にある $m \times n$ サイズのサブブロックに対して第1の単一の動き情報のセットを記憶することと、ここで、前記第1の単一の動き情報のセットは単一予測動き情報を含み、 $X = 0$

または $X = 1$ であり、

非一時的コンピュータ可読記録媒体にビットストリームを記憶すること、を含む、映像のビットストリームを記憶する方法。

【発明の詳細な説明】

【技術分野】

【0001】

関連出願の相互参照

パリ条約に基づく適用可能な特許法および／または規則に基づいて、本願は、2018年11月6日出願の国際特許出願PCT/CN2018/114057号、2019年5月17日出願の国際特許出願PCT/CN2019/087372号、2019年6月5日出願の国際特許出願PCT/CN2019/090127、および2019年6月20日出願の国際特許出願PCT/CN2019/092151号の優先権および利益を適時に主張することを目的とする。米国法に基づくすべての目的のために、上記出願の開示全体は、本明細書の開示の一部として参照により援用される。

【0002】

本明細書は、映像符号化技術、システムおよびデバイスに関する。

【背景技術】

【0003】

映像圧縮の進歩にもかかわらず、デジタル映像は、依然として、インターネットおよび他のデジタル通信ネットワークにおいて最大の帯域幅の使用量を占めている。映像の受信および表示が可能な接続されたユーザ機器の数が増加するにつれ、デジタル映像の使用に対する帯域幅需要は増大し続けることが予測される。

【発明の概要】

【0004】

ジオメトリ分割を含む、デジタル映像符号化に関連するデバイス、システム、および方法が記載される。記載された方法は、既存の映像符号化規格（例えば、高効率映像符号化（HEVC）および／または汎用映像符号化（VVC））および将来の映像符号化規格またはビデオコーデックの両方に適用され得る。

【0005】

1つの代表的な態様において、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、映像の現在のブロックと映像のビットストリーム表現との間で変換を行うために、動き候補を動き候補リストに挿入する順序を、優先規則に基づいて決定することであって、ジオメトリ分割モードを使用して符号化される、決定することと、この判定および動き候補リストに基づいて、この変換を行うことを含む。

【0006】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、現在の映像ブロックと映像のビットストリーム表現の間で変換を行うための動き候補リストに、1つの正規の動き候補から導出された、List0とList1との1つ以上の平均化された単一予測動き候補を挿入することであって、現在のブロックを、ジオメトリ分割モードを使用して符号化する、挿入することと、動き候補リストに基づいて、この変換を行うことを含む。

【0007】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、現在の映像ブロックと映像のビットストリーム表現との間で変換を行うための動き候補リストに、単一予測を伴う正規の動き候補から導出された、単一予測を伴う1つ以上の平均化動き候補を挿入することであって、現在のブロックを、ジオメトリ分割モードを使用して符号化する、挿入することと、この動き候補リストに基づいて、この変換を行うことを含む。

【0008】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用して

もよい。この方法は、現在の映像ブロックと映像のビットストリーム表現との間で変換を行うための動き候補リストに、利用可能な動き候補から導出された1つ以上の仮想動き候補を挿入することであって、現在のブロックは、ジオメトリ分割モードを使用して符号化されている、挿入することと、動き候補リストに基づいて、この変換を行うことと、を含む。

【0009】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、ジオメトリ分割モードを使用して符号化された映像の現在のブロックと映像のビットストリーム表現との間で変換するための動き候補リストへの動き候補の挿入する順序について、優先規則に基づいて決定することであって、優先規則が、動き候補の動き候補が導出された位置、または動き候補に関連付けられた1つ以上の参照ピクチャリストに基づくものである、決定することと、決定および動き候補リストに基づいて、変換を行うことを含む。

10

【0010】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は現在の映像ブロックと映像のビットストリーム表現との間で変換するために、動き候補を動き候補リストに挿入する順序を優先規則に基づいて決定することであって、現在のブロックは、ジオメトリ分割モードを使用して符号化され、順番または挿入は、シーケンス間、ピクチャ間、スライス間、タイルグループ間、または現在のブロックから映像の次のブロックに亘って変更可能である、決定することと、決定と動き候補リストに基づいて変換を行うことと、を含む。

20

【0011】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、ジオメトリ分割モードを使用して符号化された現在の映像ブロックを複数の分割に分割することと、複数の分割の各々に対応する複数の動き候補リストを構築することと、複数の動き候補リストに基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む。

【0012】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、ジオメトリ分割モードを使用して符号化された現在の映像ブロックを複数の分割に分割することと、複数の動き候補リストを構築することと、複数の分割の各々が、複数の動き候補リストのうち同じ動き候補リストを選択するように構成されることと、選択された同じ動き候補リストに基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む。

30

【0013】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、ビデオユニットにおける現在の映像ブロックとこの映像のビットストリーム表現との間で変換を行うことを含み、このビットストリーム表現は、この映像領域において有効にされるジオメトリ分割モードのための動き候補リストにおける許容される動き候補の最大数を示すフィールドを含む。

40

【0014】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、ジオメトリ分割モードを使用して複数の分割に分割された現在の映像ブロックに対して、現在のブロックの少なくとも1つの幅または高さに基づいて選択された、複数の重み係数群から1つの重み係数群を選択することと、現在のブロックと映像のビットストリーム表現との間での変換の一部として、複数の分割のうちの少なくとも2つの共通の境界に沿ったサンプルにこの重み係数群を適用することとを含む。

【0015】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、ジオメトリ分割モードを使用して、複数の分割に分割された現在の

50

映像ブロック内のサンプルに対して、複数の分割のうちの少なくとも2つの共通の境界の角度に基づく少なくとも1つの重み係数を判定することと、少なくとも1つの重み係数に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む。

【0016】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、ジオメトリ分割モードを使用して複数の分割に分割された現在の映像ブロックに対して、複数の分割のうちの少なくとも2つの共通の境界に沿った、サンプルサイズが 4×4 のサンプルサイズとは異なるサンプルに対する動き補償処理を行うことと、動き補償処理に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うこととを含む。

10

【0017】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、ジオメトリ分割モードを使用して、現在の映像ブロックを複数の分割に分割することと、複数の分割に関連付けられた動き情報から導出された、現在のブロックにおける $K \times L$ 領域のための1つの動き情報のセットを記憶することと、複数の分割のうちの少なくとも1つに関連付けられた少なくとも1つの動き情報のセットを使用して、現在のブロックと、映像のビットストリーム表現の間で変換を行うことと、を含む。

【0018】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、現在の映像ブロックを複数の分割に分割することと、ジオメトリ分割モードを使用して符号化された現在のブロックと映像のビットストリーム表現との間で変換を行うこととを含み、現在のブロックの各 $M \times N$ ブロックユニットごとに1つの動き情報のセットが記憶され、この1つの動き情報のセットは、複数の分割の各々に関連付けられた動き情報から導出されたものであり、 M または N のいずれか一方が4に等しくなく、 $M \times N$ ブロックユニットにおける各サンプルが同じ動き情報を共有する。

20

【0019】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、現在の映像ブロックを複数の分割に分割することと、現在のブロックと映像のビットストリーム表現との間での変換の一部として、記憶されるべき第1のサブ領域に関連付けられた第2の動き情報とは異なる第1の動き情報に基づいて、現在のブロックの第1のサブ領域に対して動き補償処理を行うこととを含む。

30

【0020】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、ジオメトリ分割モードを使用して、現在の映像ブロックを複数の分割に分割することと、複数の分割に関連付けられた分割方向または復号化されたマージインデックスまたはマージ候補リストに基づいて、第1のサンプルセットおよび第2のサンプルセットの動き情報を記憶することとあって、第1のサンプルセットは、複数の分割のうちの少なくとも2つの共通の境界上にあり、第2のサンプルセットは、複数の分割のうちの少なくとも2つのうちの1つの内部にある、記憶することと、記憶された動き情報に基づいて、現在のブロックと映像のビットストリーム表現との間で変換、または映像の後続のブロックとビットストリーム表現との間で変換を行うこととを含む。

40

【0021】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用してもよい。この方法は、現在の映像ブロックを複数の分割に分割することと、複数の分割のうちの少なくとも2つの分割の動き情報に基づく仮想双予測動き情報を記憶することと、仮想双予測動き情報に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む。

【0022】

別の代表的な態様では、開示される技術は、映像処理の方法を提供するために使用して

50

もよい。この方法は、現在の映像ブロックを複数の分割に分割することと、低遅延チェックフラグに基づいて、複数の分割の第1の分割（M v I n f o 1）の動き情報および複数の分割の第2の分割（M v I n f o 2）の動き情報を記憶することであって、低遅延チェックフラグは、現在のブロックを構成する現在のピクチャのPOC値以下のピクチャオーダカウンタ（POC）値を有するすべての参照ピクチャを示す、記憶することと、記憶した動き情報に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む。

【0023】

さらに別の代表的な態様において、上記方法は、処理装置が実行可能なコードの形式で実施され、コンピュータ可読プログラム媒体に記憶される。

【0024】

さらに別の代表的な態様では、上述した方法を行うように構成された、または行うように動作可能なデバイスが開示される。このデバイスは、この方法を実装するようにプログラムされた処理装置を含んでもよい。

【0025】

さらに別の代表的な態様では、ビデオデコーダ装置は、本明細書で説明されるような方法を実装し得る。

【0026】

開示される技術の上記および他の態様および特徴は、図面、説明および特許請求の範囲でより詳細に説明される。

【図面の簡単な説明】

【0027】

【図1】 マージ候補リスト構築の導出処理の一例である。

【図2】 空間的マージ候補の位置の例を示す。

【図3】 空間的マージ候補の冗長性チェックに考慮される候補対の例を示す。

【図4A】 $N \times 2N$ および $2N \times N$ 個の分割の第2の予測ユニット（PU）のための例示的な位置を示す。

【図4B】 $N \times 2N$ および $2N \times N$ 個の分割の第2の予測ユニット（PU）のための例示的な位置を示す。

【図5】 時間的マージ候補のための動きベクトルのスケーリングの例である。

【図6】 時間的マージ候補の候補位置の例、C0およびC1を示す。

【図7】 結合双予測マージ候補の例を示す。

【図8】 動きベクトル予測候補の導出処理例を示す。

【図9】 空間的動きベクトル候補のための動きベクトルのスケーリングの例である。

【図10】 CUのための代替的な時間的動きベクトル予測（ATMVP）動き予測の例を示す。

【図11】 4つのサブブロック（A-D）およびその近傍のブロック（a-d）を有する1つのCUの例を示す。

【図12】 異なるMV精度で符号化する例のフローチャートである。

【図13A】 135度分割タイプ（左上隅から右下隅への分割）、および45度分割パターンを示す。1つのCUを2つの三角形予測ユニット（2つの分割パターン）に分割する図である。

【図13B】 135度分割タイプ（左上隅から右下隅への分割）、および45度分割パターンを示す。1つのCUを2つの三角形予測ユニット（2つの分割パターン）に分割する図である。

【図14】 近傍のブロックの位置の例を示す。

【図15A】 動きベクトル記憶装置の例を示す。

【図15B】 動きベクトル記憶装置の例を示す。

【図16A】 三角形予測モード（TPM）の信号通知の例を示す。

【図16B】 三角形予測モード（TPM）の信号通知の例を示す。

10

20

30

40

50

【図 1 7】 T P M フラグ符号化におけるコンテキスト選択に用いられる近傍のブロック（A および L）の例を示す。

【図 1 8 A】 重複ブロック動き補償 O B M C が適用されるサブブロックの例を示す。

【図 1 8 B】 重複ブロック動き補償 O B M C が適用されるサブブロックの例を示す。

【図 1 8 C】 三角形予測モードのための単一予測 M V 選択の例を示す。

【図 1 9】 I C パラメータを導出するために使用される近傍のサンプルの例を示す。

【図 2 0 A】 簡略化したアフィン動きモデル 4 - パラメータアフィンおよび 6 - パラメータアフィンモデルを示す。

【図 2 0 B】 簡略化したアフィン動きモデル 4 - パラメータアフィンおよび 6 - パラメータアフィンモデルを示す。

【図 2 1】 サブブロックごとのアフィン動きベクトルフィールド（M V F）の例を示す。

【図 2 2 A】 4 パラメータアフィンモデルおよび 6 パラメータアフィンモデルの例を示す。

【図 2 2 B】 4 パラメータアフィンモデルおよび 6 パラメータアフィンモデルの例を示す。

【図 2 3】 A F _ I N T E R のための動きベクトル予測子 M V P の一例を示す。

【図 2 4 A】 A F _ M E R G E の候補の例を示す。

【図 2 4 B】 A F _ M E R G E の候補の例を示す。

【図 2 5】 アフィンマージモードの候補位置の例を示す。

【図 2 6】 オプティカルフローの軌跡の一例を示す。

【図 2 7 A】 a) ブロックの外側のアクセス位置、b) 余分なメモリアクセスおよび計算を回避するためにパディングを用いる、B I O w / o ブロック拡張の例を示す。

【図 2 7 B】 a) ブロックの外側のアクセス位置、b) 余分なメモリアクセスおよび計算を回避するためにパディングを用いる、B I O w / o ブロック拡張の例を示す。

【図 2 8】 バイラテラルテンプレートマッチングに基づくデコーダ側動きベクトル改良（D M V R）の例を示す。

【図 2 9】 異なる分割の動き候補リスト構築処理に用いられるブロックの例を示す。

【図 3 0 A】 t r i a n g l e D i r = 0 の場合の M V 記憶領域の 3 つの例を示す。

【図 3 0 B】 t r i a n g l e D i r = 0 の場合の M V 記憶領域の 3 つの例を示す。

【図 3 0 C】 t r i a n g l e D i r = 0 の場合の M V 記憶領域の 3 つの例を示す。

【図 3 1】 映像処理方法の例を示すフローチャートである。

【図 3 2】 映像処理方法の例を示すフローチャートである。

【図 3 3】 映像処理方法の例を示すフローチャートである。

【図 3 4】 映像処理方法の例を示すフローチャートである。

【図 3 5】 映像処理方法の例を示すフローチャートである。

【図 3 6】 映像処理方法の例を示すフローチャートである。

【図 3 7】 映像処理方法の例を示すフローチャートである。

【図 3 8】 映像処理方法の例を示すフローチャートである。

【図 3 9】 映像処理方法の例を示すフローチャートである。

【図 4 0】 映像処理方法の例を示すフローチャートである。

【図 4 1】 映像処理方法の例を示すフローチャートである。

【図 4 2】 映像処理方法の例を示すフローチャートである。

【図 4 3】 映像処理方法の例を示すフローチャートである。

【図 4 4】 映像処理方法の例を示すフローチャートである。

【図 4 5】 映像処理方法の例を示すフローチャートである。

【図 4 6】 映像処理方法の例を示すフローチャートである。

【図 4 7】 映像処理方法の例を示すフローチャートである。

【図 4 8】 映像処理方法の例を示すフローチャートである。

【図 4 9】 本明細書に記載されるビジュアルメディアの復号化またはビジュアルメディアの符号化技術を実装するためのハードウェアプラットフォームの一例を示すブロック図である。

【図 5 0】 開示される技術が実装され得る例示的な映像処理システムのブロック図である。

10

20

30

40

50

【発明を実施するための形態】

【0028】

本明細書は、伸張または復号化されたデジタル映像の品質を向上させるために、ビデオビットストリームのデコーダで使用する様々な技術を提供する。さらに、ビデオエンコーダは、さらなる符号化に使用される復号化されたフレームを再構成するために、符号化の処理中にこれらの技術を実装してもよい。

【0029】

本明細書では、理解を容易にするために章見出しを使用しているが、実施形態および技術を対応する章に限定するものではない。このように、一つの章からの実施形態は、他の章からの実施例と組み合わせることができる。

【0030】

1. 概要

【0031】

本特許明細書は、映像符号化技術に関する。具体的には、本発明は、映像符号化のジオメトリ分割における動きベクトル符号化および信号通知に関する。本発明は、HEVCのような既存の映像符号化規格またはファイナライズされるべき規格（例えば、汎用映像符号化：VVC）に適用され得る。本発明は、将来の映像符号化規格またはビデオコーデックにも適用可能である。

【0032】

2. 導入コメント

【0033】

映像符号化規格は、主に周知のITU-TおよびISO/IEC規格の開発によって発展してきた。ITU-TはH.261とH.263を、ISO/IECはMPEG-1とMPEG-4 Visualを、両団体はH.262/MPEG-2 VideoとH.264/MPEG-4 高度映像符号化（AVC）とH.265/HEVC規格を共同で作成した。映像符号化規格、H.262は、時間予測プラス変換符号化が利用されるハイブリッド映像符号化構造に基づく。HEVCを超えた将来の映像符号化技術を探索するため、2015年には、VCEGとMPEGが共同で共同映像探索チーム（JVET）を設立した。それ以来、多くの新しい方法がJVETによって採用され、共同探索モデル（JEM）と呼ばれる参照ソフトウェアに組み込まれてきた。2018年4月、VCEG（Q6/16）とISO/IEC JTC1 SC29/WG11（MPEG）の間に共同映像探索チーム（JVET）が作られ、HEVCと比較してビットレートを50%低減することを目標とした汎用映像符号化（VVC）規格に取り組むことになった。

【0034】

VVCドラフトの最新バージョン、即ち、汎用映像符号化（ドラフト2）は、以下を参照することができる。

【0035】

http://phenix.it-sudparis.eu/jvet/doc_end_user/documents/11_Ljubljana/wg11/JVET-K1001-v7.zip

【0036】

VTMと呼ばれるVVCの最新の参照ソフトウェアは、以下のものを見出すことができる。

【0037】

https://vcgit.hhi.fraunhofer.de/jvet/VVCSOFTWARE_VTM/tags/VTM-2.1

【0038】

2.1 HEVC/H.265におけるインター予測

【0039】

各インター予測PUは、1つまたは2つの参照ピクチャリストのための動きパラメータ

10

20

30

40

50

を有する。動きパラメータは、動きベクトルおよび参照ピクチャインデックスを含む。2つの参照ピクチャリストのうちの1つの参照ピクチャリストの使用は、`inter__pred__idc`を使用して信号通知されてもよい。動きベクトルは、予測子に対してデルタとして明確に符号化されてもよい。

【0040】

1つのCUがスキップモードで符号化される場合、1つのPUがこのCUに関連付けられ、有意な残差係数がなく、符号化動きベクトルデルタも参照ピクチャインデックスもない。マージモードを指定し、これにより、現在のPUのための動きパラメータを、空間的および時間的候補を含む近傍のPUから取得する。マージモードは、スキップモードのためだけでなく、任意のインター予測されたPUに適用することができる。マージモードの代替は、動きパラメータを明確に送信することであり、動きベクトル（より正確には、動きベクトル予測子と比較した動きベクトルの差）、各参照ピクチャリストに対応する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使用量が、PUごとに明確に信号通知される。このようなモードを、本開示では高度動きベクトル予測（AMVP）と呼ぶ。

【0041】

2つの参照ピクチャリストのうちの1つを使用することを信号通知が示す場合、1つのサンプルのブロックからPUを生成する。これを「単一予測」と呼ぶ。PスライスおよびBスライスの両方に対して単一予測が利用可能である。

【0042】

両方の参照ピクチャリストを使用することを信号通知が示す場合、2つのサンプルのブロックからPUを生成する。これを「双予測」と呼ぶ。Bスライスのみに双予測が利用可能である。

【0043】

以下、HEVCに規定されるインター予測モードについて詳細に説明する。まず、マージモードについて説明する。

【0044】

2.1.1 参照ピクチャリスト

【0045】

HEVCにおいて、インター予測という用語は、現在の復号化されたピクチャ以外の参照ピクチャのデータ要素（例えば、サンプル値または動きベクトル）から導出された予測を示すために用いられる。H.264/AVCと同様に、複数の参照ピクチャから1つのピクチャを予測することができる。インター予測に用いられる参照ピクチャは、1つ以上の参照ピクチャリストにまとめられる。参照インデックスは、リストにおけるいずれの参照ピクチャを使用して予測信号を生成するかを識別する。

【0046】

1つの参照ピクチャリスト`List0`はPスライスに用いられ、2つの参照ピクチャリスト`List0`および`List1`はBスライスに用いられる。なお、`List0/1`に含まれる参照ピクチャは、撮影／表示順にしても、過去および将来のピクチャからのものであってもよい。

【0047】

2.1.2 マージモード

【0048】

2.1.2.1 マージモードの候補の導出

【0049】

マージモードを使用してPUを予測する場合、ビットストリームからマージ候補リストにおけるエントリを指すインデックスを構文解析し、これを使用して動き情報を検索する。このリストの構成は、HEVC規格で規定されており、以下のステップのシーケンスに基づいてまとめることができる。

- ・ステップ1：初期候補導出

- ステップ1.1：空間的候補導出

10

20

30

40

50

- ステップ 1. 2：空間的候補の冗長性チェック
 - ステップ 1. 3：時間的候補導出
 - ・ステップ 2：追加候補挿入
 - ステップ 2. 1：双予測候補の作成
 - ステップ 2. 2：動きゼロ候補の挿入
- 【0050】

これらのステップは図 1 にも概略的に示されている。空間的マージ候補導出のために、5つの異なる位置にある候補の中から最大 4つのマージ候補を選択する。時間的マージ候補導出のために、2つの候補の中から最大 1つのマージ候補を選択する。デコーダ側では PUごとに一定数の候補を想定しているので、ステップ 1で得られた候補の数が、スライスヘッダにおいて信号通知されるマージ候補の最大数 (Max Num Merge Cand) に達しない場合、追加の候補を生成する。候補の数は一定であるので、最良マージ候補のインデックスを、短縮された単項 2 値化 (TU) を使用して符号化する。CUのサイズが 8 に等しい場合、現在の CUのすべての PUは、 $2N \times 2N$ 予測ユニットのマージ候補リストと同じ 1つのマージ候補リストを共有する。

【0051】

以下、上述したステップに関連付けられた動作を詳しく説明する。

【0052】

2. 1. 2. 2 空間的候補導出

【0053】

空間的マージ候補の導出において、図 2 に示す位置にある候補の中から、最大 4つのマージ候補を選択する。導出の順序は A_1 、 B_1 、 B_0 、 A_0 、 B_2 である。位置 A_1 、 B_1 、 B_0 、 A_0 のいずれかの PUが利用可能でない場合 (例えば、別のスライスまたはタイルに属しているため)、またはイントラ符号化された場合にのみ、位置 B_2 が考慮される。位置 A_1 の候補を加えた後、残りの候補を加えると、冗長性チェックを受け、それにより、同じ動き情報を有する候補を確実にリストから排除でき、符号化効率を向上させることができる。計算の複雑性を低減するために、前述の冗長性チェックにおいて、考えられる候補対のすべてを考慮することはしない。その代わりに、図 3 の矢印で結ばれたペアのみを考慮し、冗長性チェックに使用された対応する候補が同じ動き情報を持っていない場合にのみ、候補をリストに加える。重複した動き情報の別のソースは、 $2N \times 2N$ とは異なる分割に関連付けられた「第 2 の PU」である。一例として、4A および図 4B は、それぞれ、 $N \times 2N$ および $2N \times N$ の場合の第 2 の PUを描いている。現在の PUを $N \times 2N$ に分割する場合、リスト構築に位置 A_1 の候補は考慮されない。実際、この候補を加えることにより、双予測ユニットが同じ動き情報を有するようになり、1つの符号化ユニットに 1つの PUのみを有することは冗長である。同様に、現在の PUを $2N \times N$ に分割する場合、位置 B_1 は考慮されない。

【0054】

2. 1. 2. 3 時間的候補導出

【0055】

このステップにおいて、1つの候補のみがリストに追加される。具体的には、この時間的マージ候補の導出において、所与の参照ピクチャリストにおける現在のピクチャとの間に最小の POC 差を有するピクチャに属する同一位置 PUに基づいて、スケーリングされた動きベクトルを導出する。スライスヘッダにおいて、同一位置 PUの導出に用いられる参照ピクチャリストが明確に信号通知される。図 5 に点線で示すように、時間的マージ候補のスケーリングされた動きベクトルが得られる。これは、POC 距離 t_b および t_d を利用して、同一位置 PUの動きベクトルからスケーリングしたものである。 t_b は、現在のピクチャの参照ピクチャと現在のピクチャの POC 差として規定し、 t_d は、同一位置 PUの参照ピクチャと同一位置ピクチャの POC 差として規定する。時間的マージ候補の参照ピクチャインデックスをゼロに等しく設定する。このスケーリング処理の実際的な実現については、HEVC 仕様に記載されている。B スライスの場合、2つの動きベクトル

、即ち、1つは参照ピクチャリスト0のためのもの、もう1つは参照ピクチャリスト1のためのものを取得し、これらを組み合わせることによって、双予測マージ候補を形成する。

【0056】

参照フレームに属する同一位置PU(Y)において、図6に示すように、候補C₀と候補C₁との間で時間的候補の位置を選択する。位置C₀のPUが利用可能でない場合、イントラ符号化されている場合、または現在のCTUの外側にある場合、位置C₁が使用される。そうでない場合、位置C₀が時間的マージ候補の導出に使用される。

【0057】

2.1.2.4 追加候補挿入

【0058】

時空間的マージ候補の他に、2つの追加のタイプのマージ候補、すなわち、結合双予測マージ候補およびゼロマージ候補がある。空間的および時間的マージ候補を利用することで、結合双予測マージ候補を生成する。結合双予測マージ候補は、Bスライスのみで使用される。最初の候補の第1の参照ピクチャリスト動きパラメータと別の候補の第2の参照ピクチャリスト動きパラメータとを組み合わせることで、結合双予測候補を生成する。これら2つのタプルが異なる動き仮説を提供する場合、これらのタプルは、新しい双予測候補を形成する。一例として、図7は、オリジナルリスト(左側)における、mvL₀、refIdxL₀またはmvL₁、refIdxL₁を有する2つの候補を使用して、最終リスト(右側)に加えられる結合双予測マージ候補を生成する場合を示す。これらの追加のマージ候補を生成するために考慮される組み合わせについては、様々な規則が存在する。

【0059】

動きゼロ候補を挿入し、マージ候補リストにおける残りのエントリを埋めることにより、MaxNumMergeCand容量にヒットする。これらの候補は、空間的変位がゼロであり、新しいゼロ動き候補をリストに加える度にゼロから始まり増加する参照ピクチャインデックスを有する。これらの候補が使用する参照フレームの数は、それぞれ、一方方向予測の場合は1つ、双予測の場合は2つである。最終的には、これらの候補に対して冗長性チェックは行われない。

【0060】

2.1.2.5 並列処理のための動き推定領域

【0061】

符号化処理を高速化するために、動き推定を並列に行うことができ、それによって、所与の領域内のすべての予測ユニットの動きベクトルを同時に導出する。1つの予測ユニットは、その関連する動き推定が完了するまで、隣接するPUから動きパラメータを導出することができないので、空間的近傍からのマージ候補の導出は、並列処理に干渉する可能性がある。符号化効率と処理待ち時間との間のトレードオフを緩和するために、HEVCは、動き推定領域(MER)を規定し、そのサイズは、「log2_parallel_merge_level_minus2」構文要素を使用してピクチャパラメータセットにおいて信号通知される。1つのMERを規定するとき、同じ領域にあるマージ候補は使用不可としてマークされ、それゆえにリスト構築においては考慮されない。

【0062】

2.1.3 AMVP

【0063】

AMVPは、動きパラメータの明確な伝送に使用される、動きベクトルの近傍のPUとの時空間的相関を利用する。各参照ピクチャリストにおいて、まず、左側の時間的に近傍のPU位置の可用性をチェックし、冗長な候補を取り除き、ゼロベクトルを加えることで、候補リストの長さを一定にすることで、動きベクトル候補リストを構成する。次いで、エンコーダは、候補リストから最良の予測子を選択し、選択された候補を示す対応するインデックスを送信することができる。マージインデックスの信号通知と同様に、最良の動きベクトル候補のインデックスは、短縮された単項を使用して符号化される。この場合に符号化対象の最大値は2である(図8参照)。以下の章では、動きベクトル予測候補の導

10

20

30

40

50

出処理の詳細を説明する。

【0064】

2. 1. 3. 1 AMVP候補の導出

【0065】

図8に、動きベクトル予測候補の導出処理をまとめる。

【0066】

動きベクトル予測において、空間的動きベクトル候補と時間的動きベクトル候補という2つのタイプの動きベクトル候補が考えられる。空間的動きベクトル候補を導出するために、図2に示したように、5つの異なる位置にある各PUの動きベクトルに基づいて、最終的には2つの動きベクトル候補を導出する。

【0067】

時間的動きベクトル候補を導出するために、2つの異なる同じ場所に配置された位置に基づいて導出された2つの候補から1つの動きベクトル候補を選択する。第1の時空間的候補リストを作成した後、リストにおける重複した動きベクトル候補を除去する。候補の数が2よりも多い場合、関連づけられた参照ピクチャリストにおける参照ピクチャインデックスが1よりも大きい動きベクトル候補をリストから削除する。時空間的動きベクトル候補の数が2未満である場合は、追加のゼロ動きベクトル候補をリストに加える。

【0068】

2. 1. 3. 2 空間的動きベクトル候補

【0069】

空間的動きベクトル候補の導出において、図2に示すような位置にあるPUから導出された5つの候補のうち、最大2つの候補を考慮するそれらの位置は動きマージの位置と同じである。現在のPUの左側のための導出の順序は、 A_0 、 A_1 、スケーリングされた A_0 、スケーリングされた A_1 として規定される。現在のPUの上側のための導出の順序は、 B_0 、 B_1 、 B_2 、スケーリングされた B_0 、スケーリングされた B_1 、スケーリングされた B_2 として規定される。そのため、辺ごとに、動きベクトル候補として使用できる場合は4つ、すなわち空間的スケーリングを使用する必要がない2つの場合と、空間的スケーリングを使用する2つの場合とがある。4つの異なる場合をまとめると、以下のようになる。

・空間的スケーリングなし

- － (1) 同じ参照ピクチャリスト、および同じ参照ピクチャインデックス (同じPOC)
- － (2) 異なる参照ピクチャリストであるが、同じ参照ピクチャ (同じPOC)

・空間的スケーリング

- － (3) 同じ参照ピクチャリストであるが、異なる参照ピクチャ (異なるPOC)
- － (4) 異なる参照ピクチャリスト、および異なる参照ピクチャ (異なるPOC)

【0070】

最初に非空間的スケーリングの場合をチェックし、次に空間的スケーリングを行う。参照ピクチャリストにかかわらず、POCが近傍のPUの参照ピクチャと現在のPUの参照ピクチャとで異なる場合、空間的スケーリングを考慮する。左側候補のすべてのPUが利用可能でないか、またはイントラ符号化されている場合、上側の動きベクトルのスケーリングは、左側および上側MV候補の並列導出に役立つ。そうでない場合、上側の動きベクトルに対して空間的スケーリングは許可されない。

【0071】

空間的スケーリング処理において、図9に示すように、時間的スケーリングと同様にし、近傍のPUの動きベクトルをスケーリングする。主な違いは、現在のPUの参照ピクチャリストおよびインデックスを入力として与え、実際のスケーリング処理は時間的スケーリングと同じであることである。

【0072】

2. 1. 3. 3 時間的動きベクトル候補

【0073】

10

20

30

40

50

参照ピクチャインデックスを導出する以外は、時間的マージ候補を導出するための処理は、すべて、空間的動きベクトル候補を導出するための処理と同じである（図6参照）。参照ピクチャインデックスはデコーダに信号通知される。

【0074】

2.2 JEMにおける新しいインター予測方法

【0075】

2.2.1 サブCUに基づく動きベクトル予測

【0076】

QTBTを有するJEMにおいて、各CUは、各予測方向に対して最大1つの動きパラメータセットを有することができる。エンコーダにおいて、ラージCUをサブCUに分割し、ラージCUのすべてのサブCUの動き情報を導出することにより、2つのサブCUレベルの動きベクトル予測方法を考慮する。代替的な時間的動きベクトル予測（ATMVP）方法により、各CUが、配列された参照ピクチャにおける現在のCUよりも小さい複数のブロックから複数の動き情報のセットをフェッチすることが可能となる。時空間的動きベクトル予測（STMVP）法において、時間的動きベクトル予測子および空間的近傍動きベクトルを使用して、サブCUの動きベクトルを再帰的に導出する。

【0077】

サブCU動き予測のためにより正確な動きフィールドを維持するために、参照フレームの動き圧縮は現在無効にされている。

【0078】

2.2.1.1 代替の時間的動きベクトル予測

【0079】

図10を参照する。代替的な時間的動きベクトル予測（ATMVP）において、動きベクトルの時間的動きベクトル予測（TMVP）法は、現在のCUより小さいブロックから複数の動き情報のセット（動きベクトルおよび参照インデックスを含む）を取り出すことで修正される。一例において、サブCUは、 $N \times N$ 個のブロックの正方形である（ N は、デフォルトで4に設定される）。

【0080】

ATMVPは、CU内のサブCUの動きベクトルを2つのステップで予測する。第1のステップは、参照ピクチャ1における対応するブロックを、いわゆる時間的ベクトルで特定することである。この参照ピクチャをモーションソースピクチャと呼ぶ。第2のステップでは、現在のCUをサブCUに分割し、各サブCUに対応するブロックから、各サブCUの動きベクトルおよび参照インデックスを得る。

【0081】

第1のステップにおいて、現在のCUの空間的に近傍のブロックの動き情報によって、参照ピクチャおよび対応するブロックを判定する。近傍のブロックの繰り返し走査処理を回避するために、現在のCUのマージ候補リストにおける第1のマージ候補を用いる。第1の利用可能な動きベクトルおよびその関連する参照インデックスを、時間的ベクトルおよびモーションソースピクチャのインデックスに設定する。このように、ATMVPでは、TMVPに比べて、対応するブロックをより正確に特定することができ、対応するブロック（配列されたブロックと呼ばれることがある）は、常に現在のCUに対して右下または中心位置にある。

【0082】

第2のステップにおいて、現在のCUの座標に時間ベクトルを加えることで、モーションソースピクチャにおける時間的ベクトルによって、サブCUの対応するブロックを特定する。サブCUごとに、その対応するブロックの動き情報（中心サンプルを覆う最小の動きグリッド）を使用して、サブCUの動き情報を導出する。対応する $N \times N$ ブロックの動き情報を特定した後、HEVCのTMVPと同様に、現在のサブCUの動きベクトルおよび参照インデックスに変換され、動きスケーリングや他の手順が適用される。例えば、デコーダは、低遅延条件（現在のピクチャのすべての参照ピクチャのPOCが現在のピクチャ

10

20

30

40

50

ャのPOCよりも小さい)が満たされているかどうかをチェックし、場合によっては、動きベクトル MV_x (参照ピクチャリストXに対応する動きベクトル)を使用して、各サブCUの動きベクトル MV_y (Xが0または1に等しく、Yが $1-X$ に等しい)を予測する。

【0083】

2.2.1.2 時空間的動きベクトル予測 (STMVP)

【0084】

この方法において、サブCUの動きベクトルは、ラストスキャンの順に沿って再帰的に導出される。図11はこの概念を説明する。4つの 4×4 サブCU、A、B、C、およびDを含む 8×8 CUを考える。現在のフレームの近傍の 4×4 ブロックには、a、b、c、dというラベルが付けられている。

10

【0085】

サブCU Aの動きの導出は、その2つの空間的近傍を特定することで始まる。第1の近傍は、サブCU Aの上の $N \times N$ ブロックである(ブロックc)。このブロックcが利用可能でないか、またはイントラ符号化されている場合、サブCU Aより上の他の $N \times N$ 個のブロックをチェックする(ブロックcから始まり、左から右へ)。第2の近傍は、サブCU Aの左側のブロックである(ブロックb)。ブロックbが利用可能でないか、またはイントラ符号化されている場合、サブCU Aの左側の他のブロックをチェックする(ブロックbを中心に、上から下へ)。各リストの近傍のブロックから得られた動き情報を、所与のリストの第1の参照フレームにスケーリングする。次に、HEVCに規定されているTMVP導出と同じ手順に従って、サブブロックAの時間的動きベクトル予測子(TMVP)を導出する。ロケーションDにおける配列されたブロックの動き情報をフェッチし、それに応じてスケーリングする。最後に、動き情報を検索し、スケーリングした後、参照リストごとにすべての利用可能な動きベクトル(3まで)を別々に平均する。この平均化された動きベクトルを現在のサブCUの動きベクトルとする。

20

【0086】

2.2.1.3 サブCU動き予測モード信号通知

【0087】

サブCUモードは追加のマージ候補として有効とされ、モードを信号通知するために追加の構文要素は必要とされない。ATMVPモードおよびSTMVPモードを表すように、各CUのマージ候補リストに2つの追加のマージ候補を加える。シーケンスパラメータセットがATMVPおよびSTMVPが有効であることを示す場合、7個までのマージ候補を使用する。追加のマージ候補の符号化ロジックは、HMにおけるマージ候補の場合と同じであり、つまり、PまたはBスライスにおける各CUについて、2つの追加のマージ候補に対して2回以上のRDチェックが必要となる。

30

【0088】

JEMにおいて、マージインデックスのすべてのピンは、CABACによってコンテキスト符号化される。一方、HEVCにおいては、第1の2値のみがコンテキスト符号化され、残りの2値はコンテキストバイパス符号化される。

【0089】

2.2.2 適応型動きベクトル差解像度

【0090】

HEVCにおいて、`use_integer_mv_flag`がスライスヘッダにおいて0であるとき、 $1/4$ 輝度サンプルの単位で動きベクトルの差(MVD)(動きベクトルとPUの予測動きベクトルとの差)が信号通知される。JEMにおいて、ローカル適応型動きベクトル解像度(LAMVR)が導入される。JEMにおいて、MVDは、 $1/4$ 輝度サンプル、整数輝度サンプルまたは4つの輝度サンプルの単位で符号化できる。MVD分解能は符号化ユニット(CU)レベルで制御され、MVD解像度フラグは、少なくとも1つのノンゼロMVDモジュールを有する各CUに対して条件付きで信号通知される。

40

【0091】

少なくとも1つのノンゼロMVDモジュールを有するCUの場合、 $1/4$ 輝度サンプル

50

MV精度がCUにおいて使用されるかどうかを示すために、第1のフラグが信号通知される。第1のフラグ(1に等しい)が、1/4輝度サンプルMV精度が使用されていないことを示す場合、整数輝度サンプルMV精度が使用されるかまたは4輝度サンプルMV精度が使用されるかを示すために、別のフラグが信号通知される。

【0092】

CUの第1のMVD解像度フラグがゼロであるか、またはCUに対して符号化されていない(つまり、CUにおけるすべてのMVDがゼロである)場合、CUに対して1/4輝度サンプルMV解像度が使用される。CUが整数輝度サンプルMV精度または4輝度サンプルMV精度を使用する場合、CUのAMVP候補リストにおけるMVPを対応する精度に丸める。

【0093】

エンコーダにおいて、CUレベルのRDチェックは、どのMVD解像度をCUに用いるかを判定するために用いられる。すなわち、1つのMVD解像度ごとに3回、CUレベルのRDチェックを行う。エンコーダの速度を速めるために、JEMにおいては、以下の符号化方式が適用される。

- ・通常の1/4輝度サンプルMVD解像度を有するCUのRDチェック中、現在のCUの動き情報(整数輝度サンプル精度)が記憶される。整数輝度サンプルおよび4輝度サンプルのMVD解像度を有する同じCUのRDチェック中に、記憶された動き情報(丸められた後)は、更なる小範囲動きベクトル改良の開始点として使用されるので、時間がかかる動き推定処理が3回重複しない。

- ・4輝度サンプルMVD解像度を有するCUのRDチェックを条件付きで呼び出す。CUの場合、整数輝度サンプルMVD解像度のRDコストが1/4輝度サンプルMVD解像度のそれよりもはるかに大きい場合、CUのための4輝度サンプルMVD解像度のRDチェックはスキップされる。

【0094】

符号化処理を図12に示す。まず、1/4画素MVをテストし、RDコストを計算し、 $RDCost_0$ と表し、次に、整数MVをテストし、RDコストを $RDCost_1$ と表す。 $RDCost_1 < th * RDCost_0$ (ただし、 th は正の値である)である場合、4画素MVをテストし、そうでない場合、4画素MVをスキップする。基本的に、整数または4画素MVをチェックするときには、1/4画素MVに対して動き情報およびRDコスト等が既知であり、これを再利用して整数または4画素MVの符号化処理を高速化することができる。

【0095】

2.2.3 三角形予測モード

【0096】

三角形予測モード(TPM)の概念は、動き補償予測のために新しい三角形分割を導入することである。図13A、13Bに示すように、CUを対角線または逆対角線方向に2つの三角形予測ユニットに分割する。CUにおける各三角形予測ユニットは、1つの単一予測候補リストから導出された独自の単一予測動きベクトルおよび参照フレームインデックスを使用して、インター予測される。三角形予測ユニットを予測した後、対角エッジに対して適応重み付け処理を行う。そして、CU全体に対して変換および量子化処理を行う。なお、このモードは、マージモード(スキップモードを含む。スキップモードは、特別なマージモードとして扱われる)にのみ適用される。

【0097】

本願において、三角形予測モード(TPM)を、三角形分割モード、三角形予測モード、三角形分割モード、ジオメトリ分割モード、またはジオメトリ分割モードともいう。

【0098】

いくつかの実施形態において、非TPM(非ジオメトリとも称する)マージモードの場合、マージリストへのマージインデックスを信号通知し、複数の正規マージ候補を含むマージリストを、(最大5個の空間的近傍から導出された)空間的マージ候補、時間的マ

10

20

30

40

50

ジ候補 (TMVP)、HMVP、対の平均化候補、デフォルトマージ候補 (いくつかの割り当てられた参照ピクチャを有するゼロMV) として構築する。いくつかの実施形態において、通常のマージ候補は、非ジオメトリ (非TPM) 分割モードを使用して符号化された映像のブロックに対して導出された動き候補である。

【0099】

対照的に、TPM符号化ブロックの場合、マージリストへの2つのマージインデックス (予測符号化を伴う) が信号通知される。

【0100】

ある場合では、マージリストは、上述した非TPM分割モードの場合と同様に構築されてもよく、その結果、TPM動き候補が生成される。

【0101】

別の場合において、2つの分割の単一予測の動き情報を選択するために、パリティが使用してもよい。2つの選択されたマージインデックスは、 $idx0$ および $idx1$ と表されてもよい。各マージ候補について、その参照ピクチャリスト $X = (idx0 \& idx1)$ が真である場合、分割の動き情報をマージ候補リスト X 情報に設定する。そうでない場合、分割の動き情報をマージ候補リスト Y ($Y = 1 - X$) 情報に設定する。その結果、単一予測をTPM候補とする最終的な動き情報が得られ、第1の場合の候補は、通常のマージ候補 (非TPMモード符号化ブロックについて説明したマージリスト構築処理を使用して生成された) となる。

【0102】

2.2.3.1 TPMの単一予測候補リスト

【0103】

TPM動き候補リストと呼ばれる単一予測候補リストは、5つの単一予測動きベクトル候補からなる。これは、図14に示すように、5つの空間的に近傍のブロック (1~5) および2つの時間的に同じ位置にあるブロック (6~7) を含むCUの7つの近傍のブロックから導出される。単一予測動きベクトル、双予測動きベクトルのL0動きベクトル、双予測動きベクトルのL1動きベクトル、および双予測動きベクトルのL0、L1動きベクトルの平均動きベクトルの順に、7つの近傍のブロックの動きベクトルを収集し、単一予測候補リストに入れる。候補の数が5未満である場合、動きベクトルゼロをリストに加える。このTPMリストに追加された動き候補をTPM候補と呼び、空間的/時間的ブロックから導出された動き情報を正規の動き候補と呼ぶ。

【0104】

具体的には、以下のステップが含まれる。

1) A_1 、 B_1 、 B_0 、 A_0 、 B_2 、 $Col1$ 、 $Col2$ から正規の動き候補を、ブルーニング操作を実行せずに得る (図14のブロック1~7に対応)。

2) 変数 $numCurrMergeCand = 0$ を設定する

3) A_1 、 B_1 、 B_0 、 A_0 、 B_2 、 $Col1$ 、 $Col2$ 、から導出された各正規の動き候補で、 $numCurrMergeCand$ が5未満のものに対し、正規の動き候補が単一予測である ($List0$ または $List1$ のいずれかから) 場合、であり、 $numCurrMergeCand$ を1だけ増加させたTPM候補としてマージリストに直接加えられる。このようなTPM候補を、「もともと単一予測されている候補」と命名する。フルブルーニングを適用する。

4) A_1 、 B_1 、 B_0 、 B_0 、 A_0 、 B_2 、 $Col1$ 、 $Col2$ から導出され、 $numCurrMergeCand$ が5未満である各動き候補について、正規の動き候補が双予測である場合には、リスト0からの動き情報を新たなTPM候補としてTPMマージリストに追加し (つまり、リスト0からの単一予測となるように修正し)、 $numCurrMergeCand$ を1だけ増加させる。このようなTPM候補を、「短縮 $List0$ 予測候補」と呼ぶ。

フルブルーニングを適用する。

5) A_1 、 B_1 、 B_0 、 B_0 、 A_0 、 B_2 、 $Col1$ 、 $Col2$ から導出され、 $numCu$

10

20

30

40

50

`rrMergeCand`が5未満である各動き候補について、正規の動き候補が双予測である場合には、リスト1からの動き情報をTPMマージリストに追加し（つまり、リスト1からの単一予測となるように修正し）、`numCurrMergeCand`を1だけ増加させる。このようなTPM候補を「短縮List1-予測候補」と呼ぶ。
フルブルーニングを適用する。

6) `A1`、`B1`、`B0`、`A0`、`B2`、`Col1`、`Col2`、から導出され、`numCurrMergeCand`が5未満である各動き候補について、正規の動き候補が双予測である場合、

— `List0`参照ピクチャスライスQPが`List1`参照ピクチャスライスQPよりも小さい場合、`List1`の動き情報をまず`List0`参照ピクチャにスケーリングし、2つのMVの平均（一方はオリジナル`List0`からのもので、他方は`List1`からのスケーリングされたMV）をTPMマージリストに加える。このような候補を、`List0`動き候補からの平均単一予測と呼び、`numCurrMergeCand`を1だけ増加させる。

— そうでない場合、`List0`の動き情報をまず`List1`参照ピクチャにスケーリングし、2つのMVの平均（一方はオリジナル`List1`からのものであり、他方は`List0`からのスケーリングされたMV）をTPMマージリストに加える。このようなTPM候補を、`List1`動き候補からの平均単一予測と呼ばれ、`numCurrMergeCand`を1だけ増加させる。

フルブルーニングを適用する。

7) `numCurrMergeCand`が5未満である場合、ゼロ動きベクトル候補を加える。

【0105】

2.2.3.2 適応重み付け処理

【0106】

各三角形予測ユニットを予測した後、2つの三角形予測ユニット間の対角エッジに適応重み付け処理を施し、CU全体の最終予測を導出する。2つの重み係数群を以下のように定義する。

・第1の重み係数群は、 $\{7/8, 6/8, 4/8, 2/8, 1/8\}$ および $\{7/8, 4/8, 1/8\}$ をそれぞれ輝度およびクロミナンスサンプルに用いる。

・第2の重み係数群は、 $\{7/8, 6/8, 5/8, 4/8, 3/8, 2/8, 1/8\}$ および $\{6/8, 4/8, 2/8\}$ をそれぞれ輝度および色差サンプルに用いる。

【0107】

2つの三角形予測ユニットの動きベクトルの比較に基づいて、重み係数群を選択する。第2の重み係数群は、2つの三角形予測ユニットの参照ピクチャが異なる場合、またはその動きベクトルの差が16画素よりも大きい場合に用いられる。そうでない場合、第1の重み係数群を用いる。例を図15A～15Bに示す。

【0108】

2.2.3.3 動きベクトル記憶装置

【0109】

三角形予測ユニットの動きベクトル（図16Aおよび16B中、`Mv1`、`Mv2`）は、 4×4 個のグリッドに記憶される。各 4×4 グリッドに対して、CUにおける 4×4 グリッドの位置に基づいて、単一予測または双予測動きベクトルを記憶する。図16A～16Bに示すように、重み付けされていない領域に位置する（すなわち、対角エッジに位置しない） 4×4 グリッドに対して、`Mv1`または`Mv2`のいずれか一方の単一予測動きベクトルを記憶する。一方、重み付け領域に位置する 4×4 グリッドについては、双予測動きベクトルを記憶する。以下の規則に基づいて、`Mv1`および`Mv2`から双予測動きベクトルを導出する。

1) `Mv1`および`Mv2`が異なる方向（`L0`または`L1`）の動きベクトルを有する場合、`Mv1`および`Mv2`を単に組み合わせることで、双予測動きベクトルが形成される。

- 2) Mv 1とMv 2の両方が同じL 0 (またはL 1) 方向から来ている場合、
- －Mv 2の参照ピクチャがL 1 (またはL 0) 参照ピクチャリストにおけるピクチャと同じである場合、Mv 2はそのピクチャにスケーリングされる。Mv 1とスケーリングされたMv 2とを組み合わせ、双予測動きベクトルを形成する。
 - －Mv 1の参照ピクチャがL 1 (またはL 0) 参照ピクチャリストにおけるピクチャと同じである場合、Mv 1はそのピクチャにスケーリングされる。スケーリングされたMv 1およびMv 2を組み合わせ、双予測動きベクトルを形成する。
 - －そうでない場合、重み付け領域のためにMv 1のみが記憶される。

【0 1 1 0】

2. 2. 3. 4 三角形予測モード (TPM) の信号通知

10

【0 1 1 1】

TPMが使用されているかどうかを示すための1つのビットフラグが、まず信号通知されてもよい。その後、(図13 A、13 Bに示すような) 2つの分割パターン、および2つの分割の各々に対して選択されたマージインデックスをさらに信号通知する。

【0 1 1 2】

2. 2. 3. 4. 1 TPMフラグの信号通知

【0 1 1 3】

1つの輝度ブロックの幅および高さを、それぞれWおよびHで表す。 $W * H < 64$ の場合、三角形予測モードは無効になる。

【0 1 1 4】

20

1つのブロックをアフィンモードで符号化する場合、三角形予測モードも無効にされる。

【0 1 1 5】

1つのブロックがマージモードで符号化される時、1つのビットフラグを信号通知して、このブロックに対して三角形予測モードが有効とされるか無効とされるかを示すことができる。

【0 1 1 6】

このフラグは、次式(図17参照)に基づいて、3つのコンテキストで符号化される。

【0 1 1 7】

$$Ctx_index = ((left_block_L_available \ \& \ L_is_coded_with_TPM?) \ 1 : 0) \\ + ((Above_block_A_available \ \& \ A_is_coded_with_TPM?) \ 1 : 0);$$

30

【0 1 1 8】

2. 2. 3. 4. 2 (図13 A～13 Bに示すような) 2つの分割パターンの表示の信号通知、および2つの分割の各々に対する選択されたマージインデックス

【0 1 1 9】

なお、分割パターンと、2つの分割のマージインデックスとは、互いに符号化される。一例において、2つの分割が同じ参照インデックスを使用できないように制限される。そのため、2つの(分割パターン) * N (マージ候補の最大数) * (N - 1) 個の可能性がある、Nが5に設定される。1つの表示はコード化され、分割パターン、2つのマージインデックス、コード化された指示の間のマッピングは、以下に定義された配列から導出される。

40

```
const uint8_t g_TriangleCombination[TRIANGLE_MAX_NUM_CANDS][3] = {
    {0, 1, 0}, {1, 0, 1}, {1, 0, 2}, {0, 0, 1}, {0, 2, 0},
    {1, 0, 3}, {1, 0, 4}, {1, 1, 0}, {0, 3, 0}, {0, 4, 0},
    {0, 0, 2}, {0, 1, 2}, {1, 1, 2}, {0, 0, 4}, {0, 0, 3},
    {0, 1, 3}, {0, 1, 4}, {1, 1, 4}, {1, 1, 3}, {1, 2, 1},
    {1, 2, 0}, {0, 2, 1}, {0, 4, 3}, {1, 3, 0}, {1, 3, 2},
    {1, 3, 4}, {1, 4, 0}, {1, 3, 1}, {1, 2, 3}, {1, 4, 1},
```

50

{0, 4, 1}, {0, 2, 3}, {1, 4, 2}, {0, 3, 2}, {1, 4, 3},
 {0, 3, 1}, {0, 2, 4}, {1, 2, 4}, {0, 4, 2}, {0, 3, 4}} ;
 [0120]

分割パターン(45°または135°) = g_TriangleCombination[signaled_indication][0];
 [0121]

候補Aのマージインデックス = g_TriangleCombination[signaled_indication];
 [0122]

候補Bのマージインデックス = g_TriangleCombination[signaled_indication];
 [0123]

2つの動き候補A、Bを導出すると、AまたはBのいずれか一方から2つの分割の(PU1、PU2)動き情報を設定することができ、PU1がマージ候補AまたはBの動き情報を使用するかどうかは、2つの動き候補の予測方向に依存する。表1は、2つの分割を有する、2つの導出された動き候補AおよびBの間の関係を示す。

[0124]

【表1】

表1：2つのマージ候補(A, B)から導出された分割の動き情報の導出

Aの予測方向	Bの予測方向	PU1の動き情報	PU2の動き情報
L0	L0	A (L0)	B (L0)
L1	L1	B (L1)	A (L1)
L0	L1	A (L0)	B (L1)
L1	L0	B (L0)	A (L1)

[0125]

2.2.3.4.3 (merge_triangle_idxで示す)表示のエントロピー符号化

[0126]

merge_triangle_idxは、[0, 39] (それぞれを含む)の範囲内にある。K-th order Exponential Golomb (EG) コードは、merge_triangle_idxの2値化に使用される(Kは1に設定される)。

[0127]

K-th order EG

[0128]

(より多くのビットを使用してより小さな数を符号化することを犠牲にして)より少ないビットでより大きな数を符号化するため、これは、非負の整数パラメータkを使用して一般化され得る。非負の整数xを次数kのexp-Golombコードで符号化するには、次のようにする。

1. 前述のorder-0 exp-Golombコードを使用して $\lfloor x / 2^k \rfloor$ を符号化する。次に、

2. $x \bmod 2^k$ をバイナリでエンコードする。

[0129]

10

20

30

40

50

【表 2】

表 2 : E x p - G o l o m b - k 符号化の例

x	$k=0$	$k=1$	$k=2$	x	$k=0$	$k=1$	$k=2$
0	1	10	100	10	0001011	001100	01110
1	010	11	101	11	0001100	001101	01111
2	011	0100	110	12	0001101	001110	0010000
3	00100	0101	111	13	0001110	001111	0010001
4	00101	0110	01000	14	0001111	00010000	0010010
5	00110	0111	01001	15	000010000	00010001	0010011
6	00111	001000	01010	16	000010001	00010010	0010100
7	0001000	001001	01011	17	000010010	00010011	0010101
8	0001001	001010	01100	18	000010011	00010100	0010110
9	0001010	001011	01101	19	000010100	00010101	0010111

【 0 1 3 0 】

2 . 2 . 3 . 5 T P M の最近の進展

通常のマージ候補リストは、動きベクトルを余分にプルーニングすることなく、三角形分割マージ予測に再利用される。通常のマージ候補リストにおける各マージ候補について、その L 0 または L 1 動きベクトルのうちの 1 つのみを三角形予測に用いる。また、L 0 対 L 1 の動きベクトルを選択する順番は、そのマージインデックスパリティに基づく。この方式によれば、通常のマージリストをそのまま使用してもよい。

【 0 1 3 1 】

一度正規マージリストを構築すると、正規マージ候補リストにおける各マージ候補について、その L 0 または L 1 動きベクトルのうちの 1 つのみを三角形予測に用いる。また、L 0 対 L 1 の動きベクトルを選択する順番は、そのマージインデックスパリティに基づく。具体的には、図 1 8 C に示すように、偶数値インデックスを有する候補の場合、まずその L 0 動きベクトルを選択し、三角形予測に使用する。それが利用可能でない場合、その L 1 動きベクトルが代わりに使用される。奇数値インデックスを有する候補の場合、まず、その L 1 動きベクトルを選択し、これを三角形予測に使用する。それが利用可能でない場合、その L 0 動きベクトルが代わりに使用される。図 1 8 C において、各マージインデックスに対応し、まず「x」でマークされた動きベクトルを三角形予測のために選択する。

【 0 1 3 2 】

7 . 3 . 7 . 7 データ構文のマージ

【 0 1 3 3 】

10

20

30

40

50

【表 3】

merge_data(x0, y0, cbWidth, cbHeight) {	Descriptor
if (CuPredMode[x0][y0] == MODE_IBC) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	ae(v)
} else {	
if(sps_mmvd_enabled_flag cbWidth * cbHeight != 32)	
regular_merge_flag[x0][y0]	ae(v)
if (regular_merge_flag[x0][y0] == 1) {	
if(MaxNumMergeCand > 1)	
merge_idx[x0][y0]	ae(v)
} else {	
if(sps_mmvd_enabled_flag && cbWidth * cbHeight != 32)	
mmvd_merge_flag[x0][y0]	ae(v)
if(mmvd_merge_flag[x0][y0] == 1) {	
if(MaxNumMergeCand > 1)	
mmvd_cand_flag[x0][y0]	ae(v)
mmvd_distance_idx[x0][y0]	ae(v)
mmvd_direction_idx[x0][y0]	ae(v)
} else {	
if(MaxNumSubblockMergeCand > 0 && cbWidth >= 8 && cbHeight >= 8)	
merge_subblock_flag[x0][y0]	ae(v)
if(merge_subblock_flag[x0][y0] == 1) {	
if(MaxNumSubblockMergeCand > 1)	
merge_subblock_idx[x0][y0]	ae(v)
} else {	
if(sps_ciip_enabled_flag && cu_skip_flag[x0][y0] == 0 && (cbWidth * cbHeight) >= 64 && cbWidth < 128 && cbHeight < 128) {	
ciip_flag[x0][y0]	ae(v)
if(ciip_flag[x0][y0] && MaxNumMergeCand > 1)	
merge_idx[x0][y0]	ae(v)
}	
if(MergeTriangleFlag[x0][y0]) {	
merge_triangle_split_dir[x0][y0]	ae(v)
merge_triangle_idx0[x0][y0]	ae(v)
merge_triangle_idx1[x0][y0]	ae(v)
}	
}	
}	
}	
}	
}	

【 0 1 3 4 】

2. 2. 4 重複ブロック動き補償

【 0 1 3 5 】

H. 263では、以前から重複ブロック動き補償（OBMC）が使用されている。JEMにおいて、H. 263とは異なり、OBMCは、CUレベルの構文を使用してオン／オフを切り替えることができる。JEMにおいてOBMCを使用する場合、OBMCは、CUの右下の境界を除くすべての動き補償（MC）ブロック境界に対して行われる。また、輝度およびクロマ成分の両方に適用される。JEMにおいて、1つのMCブロックは1つの符号化ブロックに対応する。CUがサブCUモードで符号化された（サブCUマージ、

10

20

30

40

50

アフィン、およびFRUCモードを含む) 場合、CUの各サブブロックはMCブロックである。均一にCU境界を処理するために、OBMCは、すべてのMCブロック境界に対してサブブロックレベルで実行され、サブブロックサイズは、図18A、18Bに示すように、 4×4 に等しく設定される。

【0136】

OBMCが現在のサブブロックに適用される場合、現在の動きベクトルの他に、4つの接続された近傍のサブブロックの動きベクトルも、利用可能であり、現在の動きベクトルと同一でない場合には、現在のサブブロックのための予測ブロックを導出するために使用される。複数の動きベクトルに基づくこれらの複数の予測ブロックを組み合わせ、現在のサブブロックの最終予測信号を生成する。

【0137】

近傍のサブブロックの動きベクトルに基づく予測ブロックを P_N (N は、近傍の上、下、左、右のサブブロックのインデックス) とし、現在のサブブロックの動きベクトルに基づく予測ブロックを P_C とする。 P_N が現在のサブブロックと同じ動き情報を含む近傍のサブブロックの動き情報に基づく場合、OBMCは P_N から行われない。そうでない場合、 P_N のすべてのサンプルを P_C 内の同じサンプルに加える。すなわち、 P_N の4つの行/列を P_C に加える。 P_N には重み係数 $\{1/4, 1/8, 1/16, 1/32\}$ を用い、 P_C には重み係数 $\{3/4, 7/8, 15/16, 31/32\}$ を用いる。例外は、小さなMCブロック (すなわち、符号化ブロックの高さまたは幅が4に等しいか、または1つのCUがサブCUモードで符号化された場合) であり、その場合、2つの行/列の P_N のみが P_C に追加される。この場合、 P_N に対して重み係数 $\{1/4, 1/8\}$ が使用され、 P_C に対して重み係数 $\{3/4, 7/8\}$ が使用される。垂直 (水平) 方向に近傍のサブブロックの動きベクトルに基づいて生成された P_N に対して、 P_N の同じ行 (列) におけるサンプルを、同じ重み係数で P_C に加算する。

【0138】

JEMにおいて、サイズが256輝度サンプル以下のCUの場合、CUレベルフラグを信号通知して現在のCUに対してOBMCが適用されているかどうかを示す。サイズが256輝度サンプルよりも大きい、またはAMVPモードで符号化されていないCUの場合、OBMCがデフォルトで適用される。エンコーダにおいて、OBMCがCUに適用される場合、その影響は動き推定ステージ中に考慮される。上側近傍のブロックおよび左側近傍のブロックの動き情報を使用してOBMCにより形成された予測信号は、現在のCUの元の信号の上側および左側の境界を補償するために用いられ、その後、通常の動き推定処理が適用される。

【0139】

2.2.5 局所照明補償

【0140】

局所照明補償 (LIC) は、倍率 a およびオフセット b を使用して、照明変化の線形モデルに基づく。そして、各インターモード符号化ユニット (CU) に対して適応的に有効または無効とされる。

【0141】

LICがCUに適用される場合、最小二乗誤差法が使用され、現在のCUの近傍のサンプルおよびそれらに対応する基準サンプルを使用することによって、パラメータ a および b を導出する。具体的には、図19に示すように、CUのサブサンプリング (2:1のサブサンプリング) された近傍サンプルと、参照ピクチャ内の対応するサンプル (現在のCUまたはサブCUの動き情報によって特定される) とが使用される。ICパラメータは、各予測方向に対して別々に導出され、適用される。

【0142】

CUがマージモードで符号化される場合、LICフラグは、マージモードにおける動き情報のコピーと同様に、近傍のブロックからコピーされ、そうでない場合、ICフラグがCUに信号通知され、LICが適用されるかどうかを示す。

【0143】

1つのピクチャに対してLICが有効化されるとき、1つのCUに対してLICが適用されるかどうかを判定するために、追加のCUレベルRDチェックが必要である。CUのためにLICが有効化される場合、整数画素動き探索および小数画素動き探索のために、SADおよびSATDの代わりに、それぞれ、絶対差の平均除去和（MR-SAD）および絶対アダマール変換差の平均除去和（MR-SATD）が使用される。

【0144】

符号化の複雑性を低減するために、JEMにおいては、以下の符号化方式が適用される。
・現在のピクチャとその参照ピクチャとの間に明瞭な照度変化がない場合、LICはピクチャ全体に対して無効にされる。この状況を識別するために、エンコーダにおいて、現在のピクチャおよび現在のピクチャのすべての参照ピクチャのヒストグラムを計算する。現在のピクチャと現在のピクチャのすべての参照ピクチャとの間のヒストグラム差が所与の閾値よりも小さい場合、現在のピクチャに対してLICを無効化し、そうでない場合、現在のピクチャに対してLICを有効化する。

【0145】

2.2.6 アフィン動き補償予測

【0146】

HEVCにおいて、動き補償予測（MCP）のために並進運動モデルのみが適用される。実際の世界ではあるが、動きには様々な種類があり、例えば、ズームイン／ズームアウト、回転、透視運動、および他の不規則な動きがある。JEMにおいて、簡易アフィン変換動き補償予測が適用される。図20A-20Bに示すように、ブロックのアフィン動きフィールドは、2つの制御点動きベクトルで表される。

【0147】

ブロックの動きベクトルフィールド（MVF）は、以下の式で表される。

【0148】

【数1】

$$\begin{cases} v_x = \frac{(v_{1x} - v_{0x})}{w} x - \frac{(v_{1y} - v_{0y})}{w} y + v_{0x} \\ v_y = \frac{(v_{1y} - v_{0y})}{w} x + \frac{(v_{1x} - v_{0x})}{w} y + v_{0y} \end{cases}$$

(1-a)

【0149】

6パラメータアフィンの場合、

【0150】

【数2】

$$\begin{cases} v_x = \frac{v_{1x} - v_{0x}}{w} x + \frac{v_{2x} - v_{0x}}{h} y + v_{0x} \\ v_y = \frac{v_{1y} - v_{0y}}{w} x + \frac{v_{2y} - v_{0y}}{h} y + v_{0y} \end{cases}$$

(1-b)

【0151】

(v_{0x} , v_{0y}) は、左上隅の制御点の動きベクトルであり、(v_{1x} , v_{1y}) は、右上隅の制御点の動きベクトルであり、(v_{2x} , v_{2y}) は、左下隅の制御点の動きベクトルであり、(x , y) は、現在のブロックにおける左上サンプルに対する代表点の座標を表す。VTMにおいて、代表点をサブブロックの中心位置とする。例えば、現在のブロックにおけるサブブロックの左上の角の左上のサンプルの座標が(x_s , y_s)である場合、代表点の座標を($x_s + 2$, $y_s + 2$)とする。

【0152】

動き補償予測をさらに簡単にするために、サブブロックに基づくアフィン変換予測が適用される。サブブロックのサイズ $M \times N$ は、式2のように導出され、 $MvPre$ は、動きベクトルの分数精度(JEMでは、 $1/16$)であり、(v_{2x} , v_{2y})は、式(1)に従って算出された、左下制御点の動きベクトルである。

【0153】

【数3】

$$\begin{cases} M = clip3(4, w, \frac{w \times MvPre}{\max(abs(v_{1x} - v_{0x}), abs(v_{1y} - v_{0y}))}) \\ N = clip3(4, h, \frac{h \times MvPre}{\max(abs(v_{2x} - v_{0x}), abs(v_{2y} - v_{0y}))}) \end{cases}$$

(2)

【0154】

式(2)によって導出された後、 M および N は、それぞれ w および h の除数とするために、必要に応じて下方向に調整されるべきである。

【0155】

各 $M \times N$ 個のサブブロックの動きベクトルを導出するために、図20A~20Bに示すように、式(1)に基づいて、各サブブロックの中心サンプルの動きベクトルを計算し、 $1/16$ の分数精度に丸める。次に、2.2.3章に記載の動き補償補間フィルタを適用し、導出された動きベクトルを使用して各サブブロックの予測を生成する。

【0156】

図21は、サブブロックごとのアフィン動きベクトルフィールド(MVF)の例を示す。

【0157】

MCPの後、各サブブロックの高精度動きベクトルを丸め、通常の動きベクトルと同じ精度で保存する。

【0158】

2.2.6.1 AF__INTERモード

【0159】

JEMにおいて、2つのアフィン動きモード、AF__INTERモードおよびAF__MERGEモードがある。幅と高さの両方が8より大きいCUの場合、AF__INTERモードを適用することができる。AF__INTERモードが使用されるかどうかを示すために、ビットストリームにおいてCUレベルのアフィンフラグが信号通知される。このモードでは、近傍のブロックを使用して、動きベクトル対 $\{(v_0, v_1) \mid v_0 = \{v_A, v_B, v_C\}, v_1 = \{v_D, v_E\}\}$ を有する候補リストを構築する。図22A~22Bに示すように、ブロックA、BまたはCの動きベクトルから v_0 を選択し、近傍のブロックからの動きベクトルを、参照リストおよび近傍のブロックへの参照のPOCと、現在のC

Uへの参照のPOCと、現在のCUのPOCとの関係に基づいてスケーリングする。そして、近傍のブロックDおよびEから v_1 を選択する方法は類似している。候補リストの数が2未満である場合、AMVP候補の各々を複製した動きベクトル対でリストを埋める。候補リストが2よりも大きい場合、まず、近傍の動きベクトルの整合性（対候補における2つの動きベクトルの類似性）に基づいて候補をソートし、最初の2つの候補のみを保持する。RDコストチェックを使用して、どの動きベクトル対候補を現在のCUの制御点動きベクトル予測（CPMVP）として選択するかを判定する。そして、候補リストにおけるCPMVPの位置を示すインデックスをビットストリームにおいて信号通知する。現在のアフィンCUのCPMVPを判定した後、アフィン動き推定を適用し、制御点動きベクトル（CPMV）を求める。次に、CPMVとCPMVPとの差をビットストリームにおいて信号通知する。

10

【0160】

図23は、AF__INTERのための動きベクトル予測子MVPの一例を示す。

【0161】

AF__INTERモードにおいて、4／6パラメータアフィンモードが使用される場合、2／3個の制御点が必要であり、従って、図22A、22Bに示すように、これらの制御点のために2／3個のMVDを符号化することが必要である。既存の実装形態において、MVを以下のように導出することが提案され、即ち、 mv_{d1} および mv_{d2} は mv_{d0} から予測される。

【0162】

20

【数4】

$$mv_0 = \overline{mv}_0 + mv_{d0}$$

$$mv_1 = \overline{mv}_1 + mv_{d1} + mv_{d0}$$

$$mv_2 = \overline{mv}_2 + mv_{d2} + mv_{d0}$$

【0163】

ここで $\overline{mv}_i$ 、 mv_{di} 、 mv_1 は、それぞれ、図22Bに示すように、左上の画素（ $i=0$ ）、右上の画素（ $i=1$ ）、左下の画素（ $i=2$ ）の予測動きベクトル、動きベクトルの差分、動きベクトルである。なお、2つの動きベクトル（例えば、 $mv_A(x_A, y_A)$ および $mv_B(x_B, y_B)$ ）の加算は、2つの成分を別々に合計したものに等しく、即ち、 $newMV = mv_A + mv_B$ であり、 $newMV$ の2つのコンポーネントをそれぞれ（ $x_A + x_B$ ）および（ $y_A + y_B$ ）に設定する。

30

【0164】

2.2.6.2 AF__MERGEモード

【0165】

AF__MERGEモードでCUを適用する場合、CUは、有効な近傍の再構築ブロックから、アフィンモードで符号化された第1のブロックを得る。そして、候補ブロックの選択順は、図24Aに示すように、左、上、右上、左下から左上へとなる。図24Bに示すように、左下近傍のブロックAをアフィンモードで符号化すると、ブロックAを含むCUの左上隅、右上隅、左下隅の動きベクトル v_2 、 v_3 、 v_4 が導出される。そして、 v_2 、 v_3 、 v_4 に基づいて、現在のCUにおける左上隅の動きベクトル v_0 を算出する。次に、現在のCUの右上の動きベクトル v_1 を算出する。

40

【0166】

現在のCU v_0 、 v_1 のCPMVを導出した後、簡易アフィン動きモデル式1に基づいて、現在のCUのMVFを生成する。現在のCUがAF__MERGEモードで符号化されているかどうかを識別するために、アフィンモードで符号化された近傍のブロックが少なくとも1つある場合、ビットストリーム内にアフィンフラグを信号通知する。

50

【0167】

いくつかの実装形態において、アフィンマージ候補リストは次のステップを含む。

【0168】

継承されたアフィン候補を挿入する

【0169】

継承されたアフィン候補は、その有効な近傍のアフィン符号化ブロックのアフィン動きモデルからその候補から導出されたことを意味する。共通のベースでは、図24A～24Bに示すように、候補位置のスキャン順序は、A1、B1、B0、A0、B2のようになる。

【0170】

候補を導出した後、フルブルーニング処理を行い、同じ候補がリストに挿入されているかを確認する。同じ候補が存在する場合、導出された候補を廃棄する。

【0171】

構築されたアフィン候補を挿入する

【0172】

アフィンマージ候補リストにおける候補の数がMaxNumAffineCand未満である（この寄与において5に設定される）場合、構築されたアフィン候補を候補リストに挿入する。構築されたアフィン候補は、各制御点の近傍の動き情報を組み合わせることで候補を構築することを意味する。

【0173】

まず、図24A～24Bに示される特定された空間的近傍および時間的近傍から、制御点の動き情報を導出する。CPk（k=1, 2, 3, 4）は、k番目の制御点を表す。A0、A1、A2、B0、B1、B2、B3は、CPk（k=1, 2, 3）を予測するための空間的位置であり、Tは、CP4を予測するための時間的位置である。

【0174】

CP1、CP2、CP3、CP4の座標は、それぞれ、（0、0）、（W、0）、（H、0）、（W、H）であり、W、Hは、現在のブロックの幅および高さである。

【0175】

図25は、アフィンマージモードの候補位置の例を示す。

【0176】

各制御点の動き情報は、以下の優先順位に従って取得される。

【0177】

CP1の場合、チェックの優先順位はB2→B3→A2である。利用可能であれば、B2を使用する。そうではなく、B2が利用可能である場合、B3が使用される。B2とB3の両方が使用できない場合、A2が使用される。3つの候補のすべてが利用可能でない場合、CP1の動き情報を取得することができない。

【0178】

CP2の場合、チェックの優先順位はB1→B0である。

【0179】

CP3の場合、チェックの優先順位はA1→A0である。

【0180】

CP4にはTを用いる。

【0181】

次に、これらの制御点の組み合わせを使用して、アフィンマージ候補を構築する。

【0182】

6パラメータアフィン候補を構築するためには、3つの制御点の動き情報が必要である。3つの制御点は、以下の4つの組み合わせ（{CP1, CP2, CP4}, {CP1, CP2, CP3}, {CP2, CP3, CP4}, {CP1, CP3, CP4}）のうち1つを選択することができる。{CP1, CP2, CP3}, {CP2, CP3, CP4}, {CP1, CP3, CP4}の組み合わせを、左上、右上、左下の制御点で表される

10

20

30

40

50

6 パラメータ動きモデルに変換する。

【0183】

4 パラメータアフィン候補を構築するためには、2つの制御点の動き情報が必要である。2つの制御点は、以下の6つの組み合わせ（{CP1, CP4}、{CP2, CP3}、{CP1, CP2}、{CP2, CP4}、{CP1, CP3}、{CP3, CP4}）のうち1つを選択することができる。{CP1, CP4}、{CP2, CP3}、{CP2, CP4}、{CP1, CP3}、{CP3, CP4}の組み合わせを、左上および右上の制御点で表される4パラメータ動きモデルに変換する。

【0184】

構築されたアフィン候補の組み合わせを以下の順に候補リストに挿入する。

10

【0185】

{CP1, CP2, CP3}、{CP1, CP2, CP4}、{CP1, CP3, CP4}、{CP2, CP3, CP4}、{CP1, CP2}、{CP1, CP3}、{CP2, CP3}、{CP1, CP4}、{CP2, CP4}、{CP3, CP4}

【0186】

ある組み合わせの参照リストX（Xは0または1）に対して、制御点における使用率が最も高い参照インデックスをリストXの参照インデックスとして選択し、差分参照ピクチャに対する動きベクトルのポイントをスケーリングする。

【0187】

候補を導出した後、フルブルーニング処理を行い、同じ候補がリストに挿入されているかを確認する。同じ候補が存在する場合、導出された候補を廃棄する。

20

【0188】

動きベクトルがゼロのパディング

【0189】

アフィンマージ候補リストにおける候補の数が5未満である場合、リストが一杯になるまで、参照インデックスがゼロのゼロ動きベクトルを候補リストに挿入する。

【0190】

2. 2. 7 双方向オプティカルフロー

【0191】

双方向オプティカルフロー（BIO）は、双予測のためにブロック単位の動き補償の上で行われるサンプル単位の動きの改良である。サンプルレベルの動きの改良は、信号通知を使用しない。

30

【0192】

図26は、オプティカルフローの軌跡の一例を示す。

【0193】

ブロックの動き補償の後、 $I^{(k)}$ を基準 k （ $k=0, 1$ ）からの輝度値、そして $\partial I^{(k)} / \partial x$ と $\partial I^{(k)} / \partial y$ を、それぞれ、 $I^{(k)}$ 勾配の水平および垂直方向成分とする。オプティカルフローが有効であると仮定すると、動きベクトルフィールド（ v_x, v_y ）は、次式で与えられる。

【0194】

40

【数5】

$$\partial I^{(k)} / \partial t + v_x \partial I^{(k)} / \partial x + v_y \partial I^{(k)} / \partial y = 0.$$

(3)

【0195】

このオプティカルフロー方程式と各試料の動き軌跡のエルミート補間を組み合わせると

50

、関数値 $I^{(k)}$ と導関数 $\partial I^{(k)} / \partial x$ と $\partial I^{(k)} / \partial y$ の両方に一致するユニークな 3 次多項式が得られる。 $t = 0$ におけるこの多項式の値が B I O 予測である。

【 0 1 9 6 】

【数 6】

$$pred_{BIO} = 1/2 \cdot (I^{(0)} + I^{(1)} + v_x/2 \cdot (\tau_1 \partial I^{(1)} / \partial x - \tau_0 \partial I^{(0)} / \partial x) + v_y/2 \cdot (\tau_1 \partial I^{(1)} / \partial y - \tau_0 \partial I^{(0)} / \partial y)) \quad (4)$$

10

【 0 1 9 7 】

τ_0 および τ_1 は、図 2 8 に示すように、参照フレームまでの距離を示す。距離 τ_0 および τ_1 は、Ref 0 および Ref 1 の $\tau_0 = POC(current) - POC(Ref 0)$, $\tau_1 = POC(Ref 1) - POC(current)$ の POC に基づいて、算出される。両方の予測が同じ時系列から来たものである場合（両方とも過去から来たものであるか、または両方とも将来から来たものである）、符号が異なっている（例えば $\tau_0 \cdot \tau_1 < 0$ ）B I O は、予測が同じ時間的モーメントからのものではない（即ち、 $\tau_0 \neq \tau_1$ ）場合にのみ適用され、両方の参照領域は、非ゼロの動き（ MV_{x0} , MV_{y0} , MV_{x1} , $MV_{y1} \neq 0$ ）を有し、ブロックの動きベクトルは、時間的間隔 $MV_{x0} / MV_{x1} = MV_{y0} / MV_{y1} = -\tau_0 / \tau_1$ に比例する。

20

【 0 1 9 8 】

点 A と点 B との間の値の差 Δ （図 2 6 における動き軌跡と基準フレーム平面との交差点）を最小にすることで、動きベクトルフィールド（ v_x , v_y ）を判定する。モデルは、 Δ に対してローカルテイラー展開の第 1 の線形項のみを用いる。

【 0 1 9 9 】

【数 7】

$$\Delta = (I^{(0)} - I^{(1)}_0 + v_x(\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x) + v_y(\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y)) \quad (5)$$

30

【 0 2 0 0 】

式（5）におけるすべての値は、サンプル位置（ i' , j' ）に依存し、これまでの表記から省略した。動きがローカル周辺エリアにおいて一貫していると仮定すると、現在の予測点（ i , j ）を中心とする $(2M+1) \times (2M+1)$ 個の正方形窓の内側で、 Δ を最小限に抑える。M は 2 に等しい。

【 0 2 0 1 】

【数 8】

$$(v_x, v_y) = \arg \min_{v_x, v_y} \sum_{[i', j'] \in \Omega} \Delta^2[i', j'] \quad (6)$$

40

【 0 2 0 2 】

この最適化問題に対して、J E M は、まず垂直方向に最小化し、次に水平方向に最小化する簡単なアプローチを使用する。その結果

【 0 2 0 3 】

50

【数 9】

$$v_x = (s_1 + r) > m? clip3 \left(-thBIO, thBIO, -\frac{s_3}{(s_1 + r)} \right) : 0 \quad (7)$$

$$v_y = (s_5 + r) > m? clip3 \left(-thBIO, thBIO, -\frac{s_6 - v_x s_2 / 2}{(s_5 + r)} \right) : 0 \quad (8)$$

10

【0204】

【数10】

$$\begin{aligned} s_1 &= \sum_{[i',j] \in \Omega} (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x)^2; s_3 = \sum_{[i',j] \in \Omega} (I^{(1)} - I^{(0)}) (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x); \\ s_2 &= \sum_{[i',j] \in \Omega} (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x) (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y); \\ s_5 &= \sum_{[i',j] \in \Omega} (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y)^2; s_6 = \sum_{[i',j] \in \Omega} (I^{(1)} - I^{(0)}) (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y) \end{aligned}$$

20

(9)

【0205】

ゼロで割るかまたは非常に小さな値になることを回避するために、式(7)および式(8)に正規化パラメータ r および m を導入する。

30

【0206】

【数11】

$$r = 500 \cdot 4^{d-8} \quad (10)$$

$$m = 700 \cdot 4^{d-8} \quad (11)$$

40

【0207】

d は映像サンプルのビット深さである。

【0208】

BIOのメモリアクセスを通常の二重予測運動き補償と同じにするために、すべての予測値と勾配値である $I^{(k)}$ 、 $\partial I^{(k)} / \partial x$ 、 $\partial I^{(k)} / \partial y$ は、現在のブロック内の位置について計算される。式9において、予測ブロックの境界上の現在の予測点を中心とする $(2M+1) \times (2M+1)$ 個の正方形窓は、ブロックの外側の位置にアクセスする必要がある(図27Aに示す)。JEMでは、ブロック外の $I^{(k)}$ 、 $\partial I^{(k)} / \partial x$ 、 $\partial I^{(k)} / \partial y$ の値は、ブロック内で最も近い利用可能な値に等しくなるように設定されている。例えば、これは、図27Bに示すように、パディングとして実装されてもよい。

50

【0209】

B I Oを用いることで、サンプルごとに動きフィールドを改良することができる。計算の複雑性を低減するために、J E Mではブロックに基づくB I Oの設計が用いられている。4×4ブロックに基づいて動きの改良を計算する。ブロックに基づくB I Oにおいて、4×4ブロックにおけるすべてのサンプルの、式(9)における s_n の値を統合し、次いで、 s_n の統合した値を使用して、4×4ブロックのためのB I O動きベクトルオフセットを導出する。具体的には、ブロックに基づくB I O導出には、以下の式が用いられる。

【0210】

【数12】

$$\begin{aligned} s_{1,b_k} &= \sum_{(x,y) \in b_k} \sum_{[i',j'] \in \Omega(x,y)} (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x)^2; s_{3,b_k} = \sum_{(x,y) \in b_k} \sum_{[i',j'] \in \Omega} (I^{(1)} - I^{(0)}) (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x), \\ s_{2,b_k} &= \sum_{(x,y) \in b_k} \sum_{[i',j'] \in \Omega} (\tau_1 \partial I^{(1)} / \partial x + \tau_0 \partial I^{(0)} / \partial x) (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y), \\ s_{5,b_k} &= \sum_{(x,y) \in b_k} \sum_{[i',j'] \in \Omega} (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y)^2; s_{6,b_k} = \sum_{(x,y) \in b_k} \sum_{[i',j'] \in \Omega} (I^{(1)} - I^{(0)}) (\tau_1 \partial I^{(1)} / \partial y + \tau_0 \partial I^{(0)} / \partial y) \end{aligned}$$

(12)

【0211】

b_k は、予測ブロックのk番目の4×4のブロックに属するサンプルのセットを示す。式(7)および式(8)における s_n を((s_n, b_k) > 4)に置き換え、関連する動きベクトルオフセットを導出する。

【0212】

場合によっては、B I OのM V管理は、雑音または不規則な動きのために信頼できない場合がある。従って、B I Oにおいて、M Vレジメンの大きさは閾値 t_h B I Oにクリップされる。閾値は、現在のピクチャの参照ピクチャがすべて一方向からのものであるかどうかに基づいて判定される。例えば、現在のピクチャの参照ピクチャが全て一方向からのものであれば、閾値は $12 \times 2^{14-d}$ に設定され、そうでなければ $12 \times 2^{13-d}$ に設定される。

【0213】

H E V C動き補償処理(2D分離可能F I R)に準拠した演算を使用して、動き補償補間と同時にB I Oの勾配を算出する。この2D分離可能F I Rの入力は、ブロック動きベクトルの分数部分に基づいて、動き補償処理および端数位置($frac{X}$, $frac{Y}$)の場合と同じ参照フレームサンプルである。水平勾配 $\partial I / \partial x$ 信号を、まずデスケーリングシフト $d-8$ で端数位置 $frac{Y}$ に対応するB I O $f i l t e r S$ を使用して垂直に補間するとき、次に、勾配フィルタB I O $f i l t e r G$ を、分数位置 $frac{X}$ に対応する水平方向に $18-d$ だけデスケーリングシフトして適用する。垂直方向勾配 $\partial I / \partial y$ のとき、デスケーリングシフト $d-8$ で分数位置 $frac{Y}$ に対応するB I O $f i l t e r G$ を使用して垂直に第1の勾配フィルタを適用し、次に、B I O $f i l t e r S$ を使用して、端数位置 $frac{X}$ に対応する水平方向に $18-d$ だけデスケーリングシフトさせて信号を移動させる。妥当な複雑性を維持するために、勾配計算B I O $f i l t e r G$ および信号変位B I O $f i l t e r F$ のための補間フィルタの長さはより短い(6タップ)。表4は、B I Oにおけるブロック動きベクトルの異なる分数位置のための勾配計算に使用されるフィルタを示す。表5は、B I Oにおける予測信号の生成に用いられる補間フィルタを示す。

【0214】

10

20

30

40

50

【表 4】

表 4：B I Oにおけるグラジエント算出用フィルタ

端数画素位置	勾配用補間フィルタ (B I O F i l t e r G)
0	{ 8, -39, -3, 46, -17, 5 }
1/16	{ 8, -32, -13, 50, -18, 5 }
1/8	{ 7, -27, -20, 54, -19, 5 }
3/16	{ 6, -21, -29, 57, -18, 5 }
1/4	{ 4, -17, -36, 60, -15, 4 }
5/16	{ 3, -9, -44, 61, -15, 4 }
3/8	{ 1, -4, -48, 61, -13, 3 }
7/16	{ 0, 1, -54, 60, -9, 2 }
1/2	{ -1, 4, -57, 57, -4, 1 }

10

【 0 2 1 5 】

【表 5】

表 5：B I Oにおける予測信号生成のための補間フィルタ

端数画素位置	予測信号用補間フィルタ (B I O F i l t e r S)
0	{ 0, 0, 64, 0, 0, 0 }
1/16	{ 1, -3, 64, 4, -2, 0 }
1/8	{ 1, -6, 62, 9, -3, 1 }
3/16	{ 2, -8, 60, 14, -5, 1 }
1/4	{ 2, -9, 57, 19, -7, 2 }
5/16	{ 3, -10, 53, 24, -8, 2 }
3/8	{ 3, -11, 50, 29, -9, 2 }
7/16	{ 3, -11, 44, 35, -10, 3 }
1/2	{ 3, -10, 35, 44, -11, 3 }

20

30

【 0 2 1 6 】

J E Mにおいて、2つの予測が異なる参照ピクチャからのものである場合、B I Oをすべての双予測ブロックに適用することができる。C Uに対してL I Cが有効になっている場合、B I Oは無効になる。

【 0 2 1 7 】

J E Mにおいて、O B M Cは、通常のM C処理の後のブロックに適用される。計算の複雑性を低減するために、O B M C処理中にB I Oを適用しない。つまり、B I Oは、それ自身のM Vを使用する場合、1つのブロックのM C処理において適用され、O B M C処理において近傍のブロックのM Vを使用する場合、M C処理においては適用されない。

40

【 0 2 1 8 】

2. 2. 8 デコーダ側動きベクトル改良

【 0 2 1 9 】

バイプレディクション動作では、1つのブロック領域の予測のために、リスト0の動きベクトル (M V) とリスト1の動きベクトル (M V) を使用して形成された双予測ブロックをそれぞれ組み合わせて1つの予測信号を形成する。デコーダ側動きベクトル改良 (D

50

MV R) 方法において、バイラテラルテンプレートマッチング処理によって、双予測の2つの動きベクトルをさらに改良する。追加の動き情報を送信することなく改良されたMVを得るために、デコーダにおいてバイラテラルテンプレートマッチングを適用し、バイラテラルテンプレートと参照ピクチャにおける再構成サンプルとの間でひずみに基づく検索を行う。

【0220】

DMVRにおいて、図28に示すように、list0の最初のMV0とlist1のMV1とから、それぞれ双予測ブロックの重み付け結合（すなわち、平均）としてバイラテラルテンプレートを生成する。テンプレートマッチング操作は、生成されたテンプレートと参照ピクチャにおけるサンプル領域（最初の予測ブロックの付近）との間のコスト尺度を計算することからなる。2つの参照ピクチャの各々について、テンプレートコストが最小となるMVを、そのリストの更新されたMVと見なし、元のMVに置き換える。JEMにおいて、各リストに対して9つのMV候補を検索する。9つのMV候補は、元のMVと、水平または垂直方向のいずれかまたは両方向に元のMVに対してオフセットしている1つの輝度サンプルを有する8つの周囲のMVを含む。最後に、2つの新しいMV、即ち、MV0'およびMV1'は、図28に示すように、最終的な双予測結果を生成するために使用される。絶対差の合計(SAD)をコスト尺度として使用する。なお、1つの周囲のMVによって生成された予測ブロックのコストを計算する場合、実際のMVの代わりに、丸められたMV（整数画素）を使用して予測ブロックを得る。

【0221】

DMVRは、追加の構文要素を送信することなく、過去の参照ピクチャからの1つのMVと、将来の参照ピクチャからの1つのMVとの間の双予測のマージモードに適用される。JEMにおいて、CUに対してLIC、アフィンの動き、FRUCまたはサブCUマージ候補が有効である場合、DMVRは適用されない。

【0222】

2.2.9 サブブロックマージ候補リスト

【0223】

なお、非サブブロックマージ候補の通常マージリストに加え、すべてのサブブロック関連の動き候補を別個のマージリストに入れることが推薦される。

【0224】

サブブロック関連動き候補を別個のマージリストに入れ、「sub-blockマージ候補リスト」とする。

【0225】

一例において、サブブロックマージ候補リストは、アフィンマージ候補、ATMVP候補、および／またはサブブロックに基づくSTMVP候補を含む。

【0226】

2.2.9.1 アフィンマージリスト構築の他の例

【0227】

この寄与において、通常マージリストにおけるATMVPマージ候補をアフィンマージリストの第1の位置に移動させる。新しいリストにおけるすべてのマージ候補（すなわち、サブブロックに基づくマージ候補リスト）は、サブブロック符号化ツールに基づく。

【0228】

3. 実施形態によって克服される問題の例

【0229】

三角形分割の設計において、1つのブロックを2つの分割に分割することができる。動き補償に起因するメモリ帯域幅を節約するために、2つの分割は単一予測されるべきであることが必要である。動き補償処理中、対角エッジに対して双予測が使用され、他のすべての部分に対して単一予測が使用される。各分割に対して双予測が許可される場合、対角エッジに位置するサンプルは、4つのMVを有し、そのうちの2つは1つの分割からのものであり、2つは別の分割からのものである。このような設計には、以下のような問題が

10

20

30

40

50

ある。

1) 単一予測マージリストの設計は、参照ピクチャリストに依存しており、すなわち、参照ピクチャリスト 0 から参照ピクチャに関連付けられた M V の優先度が高くなるように設計されている。すなわち、すべての単一予測の正規の動き候補から導出された T P M 候補を、すべての双予測の正規の動き候補から導出された T P M 候補の前に挿入する。しかしながら、一般的に、空間的に近傍のブロックの位置は、m v 情報の類似性の点でより強い相関を有する。

2) 2つの分割は異なる T P M 候補インデックスを選択することができるが、2つの分割は現在のブロックの近傍のブロックとの間に異なる相関を有する可能性があるため、両方の分割に対して同じマージ候補リストを利用することは最適ではない。

3) T P M 候補の最大数は 5 に固定され、少なくともソフトウェア設計には使い勝手が悪い。

4) 重み係数群は、あらゆる種類のブロックに対して予め規定される。

5) 4 × 4 レベルに動き情報を記憶し、メモリ容量を増加させる。それらをどのように圧縮するかは研究すべき問題である。

6) T P M 動き情報は、2つの復号化されたマージインデックスのパリティに依存する。偶数値インデックスを有する候補の場合、まずその L 0 動きベクトルを選択し、三角形予測に使用する。それが利用可能でない場合、その L 1 動きベクトルが代わりに使用される。奇数値インデックスを有する候補の場合、まず、その L 1 動きベクトルを選択し、これを三角形予測に使用する。それが利用可能でない場合、その L 0 動きベクトルが代わりに使用される。従って、2つの候補が同じ参照ピクチャリストからのものであることが可能である。

【0 2 3 0】

4. 例示の実施形態

【0 2 3 1】

提案された方法は、任意の非正方形／非長方形の分割、例えば幾何学的分割に適用されてもよい。以下の説明において、非正方形／非長方形分割モード（T P M）の一例として、「三角形分割モード」を使用し、T P M 候補リストに挿入される動きベクトル予測候補を「T P M 候補」と呼ぶ。そして、以前に符号化ブロックに関連付けられた動き情報を、「正規の動き候補」とする。なお、他の種類の分割であってもよい。

【0 2 3 2】

以下の技術の詳細なリストは、一般的な概念を説明するための例と見なされるべきである。これらの技術は狭い意味で解釈されるべきではない。さらに、これらの技術は、任意の方法で組み合わせることができる。

【0 2 3 3】

注文の異なる T P M 候補リスト（A M V P／マージ／その他）構築処理

1. すべての双予測正規の動き候補から導出された T P M 候補の前に、すべての単方向予測の正規の動き候補から導出された T P M 候補を常に挿入するのではなく、予測方向を優先基準として、まずどの T P M 候補を追加するかを判定することが提案される。

a. 一例において、リスト X からの単一予測を伴うすべての正規の動き候補を、リスト Y からの単一予測を伴うものの前に T P M 候補として候補リストに挿入する。

b. 一例において、双予測の正規の動き候補から導出されたすべての T P M 候補（例えば、短縮 L i s t 0 - 予測候補、短縮 L i s t 1 - 予測候補）にはより高い優先順位が与えられ、すなわち、単一予測正規の動き候補から導出されたものの前に T P M 候補を挿入することができる。

c. 一例において、この順序は、双予測の正規の動き候補の L i s t X から導出されたすべての T P M 候補（例えば、短縮 L i s t X - 予測候補）、双予測の正規の動き候補の L i s t Y から導出されたすべての T P M 候補（例えば、短縮 L i s t Y - 予測候補）、単一予測の正規の動き候補から導出されたすべての T P M 候補（例えば、元々は単一予測候補）として定義される。

d. 一例において、1つの双予測の正規の動き候補から導出されたTPM候補を、別の双予測の正規の動き候補から導出されたTPM候補の前にリストに加えてもよい。

e. 一例において、1つの双予測の正規の動き候補から導出されたTPM候補を、別の双予測の正規の動き候補から導出されたTPM候補とインターリーブ方式で加算してもよい。

i. 正規の動き候補 C_A 、 C_B が2つであり、TPM候補が4つである場合、それぞれ C_A 、 C_B から導出された短縮 $List0/1$ -予測候補を含む。リストへの追加順序は、以下のように定義されてもよい。 C_A から導出された短縮 $List0$ -予測候補、 C_B から導出された短縮 $List1$ -予測候補、 C_A から導出された短縮 $List1$ -予測候補、 C_B から導出された短縮 $List0$ -予測候補。

10

ii. 代替的に、2つの正規の動き候補 C_A 、 C_B 、および4つのTPM候補がある場合、それぞれ C_A 、 C_B から導出された短縮 $List0/1$ -予測候補を含む。リストへの追加順序は、以下のように定義されてもよい。 C_A から導出された短縮 $List1$ -予測候補、 C_B から導出された短縮 $List0$ -予測候補、 C_A から導出された短縮 $List0$ -予測候補、 C_B から導出された短縮 $List1$ -予測候補。

2. すべての単方向予測正規の動き候補から導出されたTPM候補を、すべての双予測正規の動き候補から導出されたTPM候補の前に常に挿入するのではなく、正規の動き候補に関連付けられた符号化モード情報を優先基準として、まずどのTPM候補を追加するかを判定することが提案される。

a. 符号化された情報は、AMVPまたはマージモードを含んでもよい。

20

i. 一例において、1つの正規の動き候補 C_A がブロックAから導出され、別の1つの C_B がブロックBから導出され、ブロックAがAMVPモードで符号化され、Bがマージモードで符号化される場合、 C_A から導出されたTPM候補を、 C_B から導出されたものの前にリストに加えてもよい。代替的に、 C_B から導出されたTPM候補を、 C_A から導出されたTPM候補の前にリストに加えてもよい。

b. 符号化された情報は、参照インデックスおよび/またはPOC差を含んでもよい。

i. 一例において、1つのTPM候補 C'_A が別のTPM候補 C'_B に比べて小さい参照インデックスに関連付けられている場合、 C'_A を C'_B の前にリストに追加してもよい。

ii. 一例において、1つのTPM候補 C'_A が、別のTPM候補 C'_B と比較して、参照ピクチャと現在のピクチャとの間のより小さいPOC距離に関連付けられている場合、 C'_A は、 C'_B の前にリストに追加してもよい。

30

c. 符号化された情報は、参照ピクチャのピクチャ/スライス/タイルグループ量子化パラメータ(QP)および/または参照ピクチャの時間層インデックスを含んでもよい。

i. 一例において、1つのTPM候補 C'_A が、別のTPM候補 C'_B と比較して小さいQPを有する参照ピクチャに関連付けられている場合、 C'_A は、 C'_B の前にリストに追加してもよい。

3. すべての単方向予測正規の動き候補から導出されたTPM候補を、すべての双予測正規の動き候補から導出されたTPM候補の前に常に挿入するのではなく、動き候補が導出された位置を優先基準として、まずどのTPM候補を追加するかを判定することが提案されている。

40

a. 一例において、1つの正規の動き候補 C_A がブロックAから導出され、Aの後に別の C_B がブロックBから導出され、Bがチェックされた場合、 C_A から導出されたTPM動き候補を、 C_B から導出されたものの前にリストに加えてもよい（例えば、 C_A が単一予測である場合、直接継承するか、またはそれぞれが C_A に関連付けられた単一予測方向からコピーされた2つのTPM候補）。

b. 代替的に、位置および参照ピクチャリスト（または予測方向）の両方を優先基準として使用してもよい。例えば、1つの正規の動き候補 C_A がブロックAから導出され、別の C_B がブロックBから導出され、Aの後にBがチェックされた場合、リストXの C_A から導出されたTPM動き候補をリストに加えた後、リストXの C_B から導出されたTPM動き候補をリストに加えてもよい。リストYの C_A から導出されたTPM動き候補をリ

50

スト Y の C_B から導出されたものに加えてもよく、X が 0 または 1 に等しく、Y が $(1 - X)$ に等しい。

4. 短縮 $List 0$ - 予測候補の前に、すべての短縮 $List 1$ - 予測候補を加えることが提案される。

a. 代替的に、例えば、利用可能な TPM 動き候補に基づいて、短縮 $List 0$ - 予測候補および短縮 $List 1$ - 予測候補の順番を適応的に変更してもよい。

i. 一例において、第 1 の TPM 動き候補が、 $List 1$ からの単一予測を伴う、本来 1 つの予測された候補である場合、短縮 $List 0$ - 予測候補の前に、トランケートされた $List 1$ - 予測候補を追加してもよい。

ii. 代替的に、元々未予測の候補が $List 1$ よりも多い場合、短縮 $List 0$ - 予測候補の前に短縮 $List 1$ - 予測候補を追加してもよい。

b. 代替的に、1 つの双予測動き候補から導出されたすべての TPM 候補（例えば、1 つの短縮 $List 0$ - 予測候補および 1 つの短縮 $List 0$ - 予測候補）を、別の双予測動き候補から導出されたすべての TPM 候補の前にリストに追加してもよい。

i. 代替的に、1 つの動き候補から導出された TPM 候補について、それらを加える順番は、リストにおける利用可能な TPM 動き候補の情報にさらに依存してもよく、例えば、4. a である。

【0234】

より多くの TPM 候補を有する TMP 候補リスト（AMVP / マージ / その他）構築処理

5. 1 つの正規の動き候補から導出された $List 0$ および $List 1$ の動き候補からの複数の平均単一予測を、両方とも TPM 動き候補リストに加えてもよい。

a. 代替的に、それらのうちの 1 つのみを追加してもよく、どの 1 つを追加するかは、リストにおける利用可能な TPM 候補の情報、例えば、4. a に依存する。

b. 代替的に、2 つの予測リストの参照ピクチャインデックスに基づいて、いずれか一方のみを加えるようにしてもよい。より小さい参照ピクチャインデックスに関連付けられた動きベクトルは保持され、最終予測方向がこのような参照ピクチャに関連付けられる。

c. 代替的に、これらのうちの 1 つのみを追加してもよく、いずれを追加するかは、2 つの予測リストにおける参照ピクチャと現在のピクチャとの間の POC 距離に依存する。より小さな POC 距離に関連付けられた動きベクトルが保持され、最終的な予測方向がこのような参照ピクチャに関連付けられる。

d. 一例において、 $List 0$ の動き候補からのすべての平均化された単一予測を、 $List 1$ の動き候補からのすべての平均化された単一予測の前に挿入してもよい。代替的に、 $List 1$ の動き候補からのすべての平均化された単一予測を、 $List 0$ の動き候補からのすべての平均化された単一予測の前に挿入してもよい。

6. 単一予測の正規の動き候補を使用して平均単一予測 TPM 候補を導出できるようにすることが提案される。

a. リスト L X からの単一予測正規の動き候補を使用して、例えば、双予測正規の動き候補またはリスト X からの他の単一予測正規の動き候補を使用して、リスト L X の平均化単一予測候補を生成してもよい。

b. リスト L X からの単一予測正規の動き候補をまず LY ($Y = 1 - X$) にスケールリングし、このスケールリングした動きベクトルを使用して、リスト L Y の平均化単一予測候補を生成してもよく、例えば、双予測正規の動き候補またはリスト X からの他の単一予測正規の動き候補を用いる。

7. また、利用可能な TPM 動き候補から導出された仮想 TPM 動き候補を TPM 候補リストに加えてもよい。

a. 一例において、特定のステップの後（例えば、短縮 $List 1$ 予測候補を生成した後）、マージリストに追加された仮想候補から仮想候補から導出されてもよい。

b. 一例において、仮想候補は、元の単一予測候補のみから導出されてもよい。

c. 一例において、仮想候補は、元の単一予測候補および短縮 $List 1$ 予測候補のみから導出されてもよい。

d. 一例において、仮想候補は、ある参照ピクチャインデックス（即ち、0 に等しい）を有する候補、および／または参照ピクチャと現在のピクチャとの間にある範囲の POC 距離を有する候補、または 1 つの参照ピクチャリストから導出されてもよい。

e. 一例において、仮想候補は、第 1 の K 個の TPM 動き候補のみから導出されてもよく、例えば、K を 1 に設定する。

i. 例えば、2 つの TPM 動き候補の MV を平均化し、仮想候補を得ることができる。

f. 動きベクトルが C_{mv} であり、参照インデックスが C_{refidx} である TPM 動き候補に対して、以下の方法を使用して仮想ベクトルを導出してよい。

i. C_{mv} の水平または垂直成分にオフセットを加え、同じ C_{refidx} を用いる。

ii. C_{mv} の水平成分および垂直成分の両方にオフセットを加え、同じ C_{refidx} を使用する。

iii. C_{refidx} が指していない他の参照ピクチャに C_{mv} をスケーリングする。スケーリングされた動きベクトルと C_{mv} をスケーリングした参照ピクチャのインデックスとを使用して、新しい TPM 動き候補とする。

g. 仮想 TPM 候補を追加するとき、プルーニングは適用されないことがある。

i. 代替的に、仮想 TPM 候補を生成したものを除いた候補に対してプルーニングを適用してもよい。

【0235】

適応挿入順序を有する TPM 候補リスト（AMVP / マージ / その他）構築処理

8. TPM 候補の挿入順序（優先順位基準）は、シーケンスからシーケンスへ、ピクチャからピクチャへ、スライスからスライスへ、タイルグループからタイルグループへ、ブロックからブロックへ等々と変更されてもよい。

a. 一例において、それは、ブロックサイズ / ブロック形状 / 分割パターンに依存し得る。

b. 代替的に、この順序は、VPS / SPS / PPS / スライスヘッダ / タイルグループヘッダ / タイル / CTU / CU において、エンコーダからデコーダに信号通知されてもよい。

【0236】

TPM 符号化ブロックにおける分割のための複数の TPM リスト

9. ジオメトリ分割ごとに、すべての分割に 1 つのリストを使用する代わりに、1 つの別個の動き候補リストを構築してもよい。

a. 一例において、1 つの分割の場合、その関連付けられた候補リストは、リスト X から予測された動き候補のみを含み、他の分割の場合、その関連付けられた候補リストは、リスト Y から予測された動き候補のみを含み、Y は X に等しくない。

b. 一例において、異なる空間的ブロックおよび／または時間的ブロックにアクセスして、マージリストに追加されるべき動き候補を導出してよい。

c. 一例において、1 つのジオメトリ分割における空間的ブロックおよび／または時間的ブロックの位置は、分割の位置に依存してもよい。

d. 一例において、1 つのジオメトリ分割における空間的ブロックおよび／または時間的ブロックの位置は、分割方式（左上から右下へ（45 度）または右上から左下へ（135 度））に依存してもよい。

e. 一例において、135 度の分割パターン（図 13A ~ 13B に示す）においては、PU1 のマージリストを構築するために、より上のブロックをチェックしてもよい。上記ブロック（灰色でマークされている）のいくつかの例を図 29 に示す。

f. 一例において、135 度の分割パターン（図 13A ~ 13B に示す）においては、PU2 のマージリストを構築するために、より多くの左側のブロックをチェックしてもよい。左ブロック（灰色でマークされている）のいくつかの例を図 29 に示す。

g. 一例において、45° の分割パターン（図 13A ~ 13B に示す）においては、PU1 のマージリストを構築するために、より左側のブロックおよびより上側のブロック

10

20

30

40

50

をチェックしてもよい。左上のブロック（灰色でマークされている）のいくつかの例を図 29 に示す。

h. 一例において、 45° の分割パターン（図 13 A ~ 13 B に示す）においては、PU2 のマージリストを構築するために、より多くの時間的ブロックをチェックしてもよい。

i. 一例において、TPM リストのインデックスは、分割ごとに信号通知される。

10. 複数の TPM リストを構築し、すべての分割で共有することができ、各分割はそれらのうちの 1 つを選択することができる（例えば、各分割は、複数の TPM リストから同じ TPM リストを選択することができる）。

a. 一例において、TPM リストのインデックスは、最初に信号通知されてもよい。次いで、TPM 候補のインデックスをさらに信号通知することができる。

b. 別の例において、TPM リストのインデックスと TPM 候補のインデックスとは、結合して符号化されてもよい。

【0237】

最大許容 TPM 候補の信号通知

11. マージ / AMVP のための最大許容 TPM 候補は、SPS / VPS / PPS / ピクチャヘッダ / スライスヘッダ / タイルグループヘッダ / LCU の行 / グループにおいて信号通知される。

a. 代替的に、最大許容 TPM 動き候補は、非 TPM 符号化ブロックに使用されるもの、例えば、非 TPM マージ候補リストサイズ、またはサブブロックマージ候補リストサイズに設定される。

b. 最大数は、信号通知される時に、単項符号化または切り捨て単項符号化によって 2 値化される。

c. 最大数を直接信号通知する代わりに、 $(M - \text{最大数})$ を信号通知してもよく、例えば、 $M = 5$ または 6 である。

d. 1 つのビットフラグを、TPM 候補の最大数が正規の動き候補 / サブブロック動き候補の最大数と同じであるかどうかを示すように信号通知することができる。

【0238】

重み係数群

12. 重み係数群の選択は、現在のブロックの幅および / または高さに依存してもよい。

a. 一例において、幅と高さの比が閾値よりも大きい場合、特定の重み係数群（例えば、群 1）を選択してもよい。

b. 一例において、高さとの比が閾値よりも大きい場合、特定の重み係数群（例えば、群 1）を選択してもよい。

c. 複数の重み係数群のセットを予め規定しておき、ブロックの幅および / または高さが異なる場合、そのうちの 1 つまたは 2 つから 1 つの重み係数群を選択してもよい。

13. サンプルの位置に基づいて、重み係数をオンザフライで変更してもよい。

a. 一例において、この角度は、2 つの分割を区別するエッジの角度に依存してもよい。

b. 一例において、重み係数は、サンプルの位置の関数として定義されてもよい。

c. 一例において、1 つの TPM 符号化ブロックの最終予測ブロックは、 $(f(x, y) * P1(x, y) + (2M - f(x, y)) * P2(x, y) + offset) > M$ と設定される。P1, P2 は、 (x, y) に位置するサンプルの 2 つの予測値であり、M は整数値であり、オフセットは $(1 < (M - 1))$ に設定でき、 $f(x, y)$ は、第 1 の予測値に適用された重みを戻す係数である。

14. 重み付け領域に位置するブロックに対して、メモリ帯域幅を低減するために、 4×4 レベルの代わりに 8×8 レベルで動き補償を行う。

a. 代替的に、重み付け領域に位置するブロックに対して、メモリ帯域幅を低減するために、 4×4 レベルの代わりに 8×4 または 4×8 レベルで動き補償を行う。

b. 代替的に、 8×4 または 4×8 に基づく動き補償が許可される場合、 8×4 また

10

20

30

40

50

は 4×8 の選択は、以下に依存し得る。

i. ブロックの幅および／または高さ

ii. 分割パターン（例えば、4 5 度または 1 3 5 度）。

【0 2 3 9】

動き情報の記憶装置

15. TPM符号化ブロックを復号化した後、1つまたは複数のHMVPテーブルを更新してもよい。

a. 一例において、他の分割の動き情報を除く1つの分割の動き情報に基づいて、1つまたは複数のHMVPテーブルを更新してもよい。

b. 一例において、記憶された動き情報に基づいて、重み付けられていない領域を除いた重み付け領域／または重み付けられていない領域から重み付けられた領域を除いた重み付け領域のために、1つまたは複数のHMVPテーブルを更新してもよい。

c. 一例において、TPM符号化ブロックスフ動き情報の導出に用いられるマージ候補リストにおける1つのマージ候補に基づいて、1つまたは複数のHMVPテーブルを更新してもよい。

i. 一例において、1つのマージ候補は、復号化されたマージインデックスのマージインデックス（例えば、`merge_triangle_idx0`または`merge_triangle_idx1`）に関連付けられる。

ii. 一例において、1つのマージ候補は、 k に等しいマージインデックス（例えば、 $k=0$ ）に関連付けられる。

16. 1つの分割のための三角形動き情報の導出処理は、例えば他の分割の予測方向に基づいて、他の分割からの復号化された動き情報に依存してもよい。

d. 一例において、`merge_triangle_idx0`に等しいインデックスを有する1つの復号化された候補のLXとして表される第1の参照ピクチャリストが（例えば、`merge_triangle_idx0`のパリティおよびLXの可用性に基づいて）選択される。第2の復号化された候補インデックスのパリティにかかわらず、第2の候補のLY（ $Y=1-X$ ）として示される第2の参照ピクチャリストに関連付けられた動き情報が利用可能である場合、それを三角形分割の動き情報として選択する。以下の説明において、参照ピクチャリストLXまたはLYに関連付けられた動き情報を、LX動き情報またはLY動き情報とも呼ぶ。

i. 代替的に、`merge_triangle_idx1`を有する第2の候補のLY（ $Y=1-X$ ）動き情報が利用可能でない場合、第2の候補のLX動き情報を代わりに使用してもよい。

ii. 代替的に、`merge_triangle_idx1`を有する第2の候補のLY（ $Y=1-X$ ）動き情報が利用可能でない場合、第2の候補のLX動き情報を使用してLYの動き情報を導出してもよい。そして、導出されたLY動き情報をTPM符号化に使用してもよい。

1) 一例において、この導出処理は、LXにおける参照ピクチャを参照し、LYにおける参照ピクチャを参照する動きベクトルのスケーリングを含んでもよい。

e. 一例において、`merge_triangle_idx1`に等しいインデックスを有する1つの復号化された候補のLXとして表される第1の参照ピクチャリストが選択される（例えば、`merge_triangle_idx1`のパリティおよびLXの可用性に基づいて）。第2の復号化された候補インデックスのパリティにかかわらず、LY（ $Y=1-X$ ）動き情報が利用可能である場合、それを三角形分割の動き情報として選択する。

i. 代替的に、`merge_triangle_idx0`を有する候補のLY（ $Y=1-X$ ）動き情報が利用可能でない場合、LX動き情報を代わりに使用してもよい。

ii. 代替的に、`merge_triangle_idx0`を有する候補のLY（ $Y=1-X$ ）動き情報が利用できない場合、`merge_triangle_idx0`を有する候補のLX動き情報を使用してLYの動き情報を導出してもよい。そして、導出さ

10

20

30

40

50

れたLY動き情報をTPM符号化に使用してもよい。

1) 一例において、この導出処理は、LXにおける参照ピクチャを参照し、LYにおける参照ピクチャを参照する動きベクトルのスケーリングを含んでもよい。

17. 複数の三角形分割に対して同じ参照ピクチャリストからの動き情報を使用することは許されない。

f. 一例において、1つの適合ビットストリームは、2つの三角形分割が異なる参照ピクチャリストから予測されることを満たすべきである。

g. 代替的に、マージインデックスのパリティから動き情報を導出する処理は、例えば、黒丸15に開示される方法を使用して、2つの異なるリストから2つの分割を予測することを常に出力してもよい。

18. 動き情報をどのように記憶するか、および／または重み付け領域および／または非重み付け領域のために記憶されるべき動き情報をどのように導出するかは、参照ピクチャリストにおける三角形分割および／または参照ピクチャの動き情報を導出するために利用される三角形分割および／またはマージ候補リストの分割方向および／またはデコードされたマージインデックスに依存してもよい（例えば、全ての参照ピクチャは現在のピクチャと比較してPOC値が小さいまたは大きくない）。

a. 一例において、1つの分割のために選択された復号化マージインデックスに関連付けられたマージ候補の動き情報を記憶してもよい。

i. 一例において、選択された復号マージインデックスは、復号マージ候補インデックスの1つ、例えば、merge_triangle_idx0またはmerge_triangle_idx1に設定されてもよい。

ii. 一例において、選択された復号化マージインデックスは、変数kに設定される。

1) 一例において、kは0に設定される。

2) 一例において、k番目のマージ候補のすべての情報を継承して記憶してもよい。

3) 一例において、k番目のマージ候補の部分情報を継承し、残りの情報を記憶する前に修正してもよい。

(a) 一例において、k番目のマージ候補の一般化双予測(GBi)に用いられる重み係数の表示は、継承されなくてもよい。代替的に、TPMが記憶した動き情報に特定のGBi重み付けインデックスを割り当ててもよい（例えば、0）。

(b) 一例において、k番目のマージ候補の動きベクトル精度の表示は、継承されなくてもよい。

iii. 代替的に、選択された復号化マージインデックスは、分割方向に依存してもよい。

iv. 一例において、選択された復号化されたマージインデックスに等しいインデックスを有するマージ候補が双予測である場合、双予測動き情報を記憶してもよい。

v. 一例において、選択された復号化されたマージインデックスに等しいインデックスを有するマージ候補が単一予測である場合、単一予測の動き情報を記憶してもよい。

1) 代替的に、双予測動き情報を記憶してもよく、記憶された動き情報はマージ候補から導出されてもよい。

vi. 代替的に、2つの分割の動き情報に関する関数を記憶してもよい。

1) 例えば、2つの分割の動きベクトルの平均を記憶してもよい。

b. 一例において、MvInfo1およびMvInfo2が2つの分割の動き情報を示すとする、MvInfoAにおけるLXからLY($Y=1-X$) (Aが1または2)に1つの予測方向を修正することで、仮想双予測動き情報を記憶してもよい。

i. 一例において、MvInfoAの動きベクトルおよび参照インデックスは、不変に維持される。

ii. 一例において、MvInfoAの参照インデックスは不変に維持され、MvInfoAの動きベクトルは反対の値に設定される。

10

20

30

40

50

i i i . 双予測動き情報は、修正 $MvInfoA$ および未修正 $MvInfoB$ ($B = 3 - A$ と共に) を含んでもよい。

i v . 一例において、このような修正は、 $MvInfo1$ および $MvInfo2$ の両方が1つの同じ予測方向から来たものである場合にのみ適用されてもよい。

c . 一例において、 $MvInfo1$ および $MvInfo2$ が2つの分割の動き情報を示すとする、動き情報をどのように記憶するかは、現在のピクチャと比較してすべての参照ピクチャの POC 値が小さいか否かを示す低遅延チェックフラグに依存しうる。

i . 一例において、以下が順に適用される。

$MvInfo1$ と $MvInfo2$ が異なる参照ピクチャリストである場合、 $MvInfo0$ と $MvInfo1$ を組み合わせ、双予測動き情報として記憶してもよい。

i i . $MvInfo1$ と $MvInfo2$ が同じ LX ($X = 0$ または 1) 方向にある場合、以下が適用されてもよい。

低遅延チェックフラグが真であれば (例えば、全ての参照ピクチャが現在のピクチャに比べて POC 値が小さいか、または大きくない)、予測方向リストを LX から LY に設定して $MvInfoA$ (例えば、 $A = 2$) を修正してもよく ($Y = 1 \sim X$)、修正された $MvInfoA$ と修正されていない $MvInfoC$ ($C = 3 \sim A$) を組み合わせて仮想的な双予測動き情報を形成して記憶する。

そうでない場合、 $MvInfoB$ (例えば、 $B = 2$) が記憶される。すなわち、単一予測の動き情報が記憶される。

i i i . 一例において、 $MvInfo1$ は、図13Aの $PU1$ および図13Bの $PU1$ と関連付けられ、 $MvInfo2$ は、図13Aの $PU2$ および図13Bの $PU2$ と関連付けられる。

d . 一例において、上記方法は、1つのブロック内の特定のサブブロックの動き情報の記憶に適用されてもよい。

i . 一例において、特定のサブブロックは、重み付け領域におけるサブブロックであってもよい。

i i . 一例において、特定のサブブロックは、ブロックにおける対角線または反対角線を含むサブブロックであってもよい。

i i i . 一例において、特定のサブブロックは、ブロックの右下隅にあるサブブロックであってもよい。

i v . 一例において、特定のサブブロックは、ブロックの右列または下行にあるサブブロックであってもよい。

v . 代替的に、上記方法は、1つのブロック内のすべてのサブブロックの動き情報の記憶に適用されてもよい。

19 . TPM モードの場合、ブロック内のあるサブ領域の動き情報は、このサブ領域の再構成のための動き補償処理で使用されるものとは異なってもよい。

a . 一例において、重み付け領域に位置する $M \times N$ サブ領域 (例えば、三角形予測モードの現在の符号化ユニットにおける 4×4) の場合、動き補償は、双予測として行われてもよいが、双予測の $List0$ または $List1$ からの1つの動き情報のセットのみが記憶されてもよい。

i . 代替的に、重み付け領域に位置する $M \times N$ サブ領域 (例えば、三角形予測モードの現在の符号化ユニットにおける 4×4) の場合、動き補償は、双予測として行われてもよいが、リスト X からの単一予測情報のみを記憶してもよく、記憶された情報は、動き補償処理で利用されるリスト X からの動き情報とは異なる。

i i . 代替的に、重み付け領域に位置する $M \times N$ サブ領域 (例えば、三角形予測モードの現在の符号化ユニットにおける 4×4) の場合、動き補償は、双予測として行われてもよいが、動き補償処理において利用されるものとは異なる (例えば、異なる MV および/または異なる参照ピクチャ) 双予測情報が記憶されてもよい。

b . 一例において、非重み付けエリアに位置する $M \times N$ サブ領域 (例えば、三角形予測モードの現在の符号化ユニットにおける 4×4) の場合、動き補償は、単一予測として

10

20

30

40

50

行われてもよいが、双予測動き情報が記憶されてもよい。

i. 代替的に、非重み付け領域に位置する $M \times N$ 個のサブ領域（例えば、三角形予測モードの現在の符号化ユニットにおいて 4×4 ）の場合、動き補償は、単一予測として行われてもよいが、単一予測の動き情報が記憶されてもよく、動き補償処理において利用されるものとは異なってもよい。

c. 記憶された動き情報は、他のブロックの符号化に利用されてもよい（例えば、近傍のブロックのマージ／AMVPモードにおける空間的動き候補として）。

d. 記憶された動き情報は、異なるピクチャにおける将来のブロックを符号化するために利用されてもよい（例えば、時間的動きベクトル候補を導出するために用いられる）。

e. 記憶された動き情報は、デブロッキングまたはALF等のインループ処理に利用されてもよい。

10

f. 一例において、ブロック内の第1のサブ領域の動き情報は、第1のサブ領域の再構成の動き補償処理に用いられるものと異なってもよく、一方、ブロック内の第2のサブ領域の動き情報は、第2のサブ領域の再構成の動き補償処理に用いられるものと同じであってもよく、第1のサブ領域と第2のサブ領域とは、三角形予測モードにおいて同じ符号化ユニットであってもよい。

20. 重み付け領域に位置するか否かにかかわらず、ブロック全体に対して同じ動き情報のセットを記憶することが提案される。 $MvInfo1$ および $MvInfo2$ が、2つの分割の動き情報をそれぞれ示す（例えば、復号化された／導出された2つのマージ候補インデックスに基づいて）。

20

a. 一例において、単一予測の動き情報（例えば、2つの分割のうちの1つの分割から継承する、または2つの分割の動き情報から導出される）が記憶されてもよい。

i. 一例において、 $MvInfo1$ は、ブロック全体に対して記憶されてもよい。

1. 一例において、 $MvInfo1$ は、ブロック内のすべてのサブ領域に対して記憶されてもよい。

ii. 一例において、 $MvInfo2$ は、ブロック全体に対して記憶されてもよい。

1. 一例において、 $MvInfo1$ は、ブロック内のすべてのサブ領域に対して記憶されてもよい。

iii. 一例において、どの分割の動き情報を記憶するかは、現在のピクチャとの間のPOC距離に依存してもよい。

30

1. 例えば、 $MvInfo1$ が参照する現在のピクチャと参照ピクチャとの間のPOC距離の絶対値が、 $MvInfo2$ が参照する現在のピクチャと参照ピクチャとの間の値よりも小さい場合、 $MvInfo1$ を記憶してもよい。

iv. 一例において、どの分割の動き情報を記憶するかは、参照ピクチャのQPに依存しうる。

1. 例えば、 $MvInfo1$ は、 $MvInfo1$ が参照する参照ピクチャのQPが、 $MvInfo2$ が参照する参照ピクチャの値よりも小さいことを記憶してもよい。

v. 一例において、どの分割の動き情報を記憶するかは、参照ピクチャの参照インデックスに依存してもよい。

1. 例えば、 $MvInfo1$ は、 $MvInfo1$ が参照する参照ピクチャのQPが、 $MvInfo2$ が参照する参照ピクチャの値よりも小さいことを記憶してもよい。

40

2. 例えば、 $MvInfo1$ が参照する参照ピクチャの参照インデックスが、 $MvInfo2$ が参照する参照ピクチャの値よりも小さい場合、 $MvInfo1$ を記憶してもよい。

vi. 一例において、どの分割の動き情報を記憶するかは、1つの分割に関連付けられたマージインデックスに依存してもよい。

1. 例えば、 $MvInfo1$ から導出された関連付けられたマージインデックスが $MvInfo2$ から導出されたものよりも小さい場合、 $MvInfo1$ を記憶してもよい。

2. 例えば、`merge_triangle_idx0`が`merge_triangle`

50

`ngl_e_idx1`より小さい場合、`merge_triangle_idx0`から導出された動き情報を記憶してもよい。

`vii`．一例において、`MvInfo1`および`MvInfo2`から導出された第3の動き情報のセット（`MvInfo3`と表される）が記憶されてもよい。

1．一例において、`MvInfo3`のMVは、`MvInfo1`および`MvInfo2`の2つのMVの平均として導出されてもよく、または`MvInfo1`および`MvInfo2`の一方のMVからのスケーリングされたまたはマッピングされた動きベクトルとして導出され、他方のMVを平均して`MvInfo3`のMVを生成してもよい。

2．一例において、`MvInfo3`の参照ピクチャは、`MvInfo1`および`MvInfo2`の2つの参照ピクチャのうちの1つであってもよい。

`viii`．一例において、`MvInfoX`（ X は0または1）における1つの動きベクトルを、`MvInfoY`（ Y は $1-X$ ）における参照ピクチャに対してスケーリングし、次に、それを使用して記憶されるべき動き情報を導出してもよい。

1．一例において、スケーリングされた`MvInfo1`および`MvInfo2`は、黒丸`vii`に規定されるような、記憶されるべき動き情報を導出するために用いられてもよい。

`ix`．上記方法は、重み付け領域に位置するサブ領域にのみ適用可能である。

1．代替的に、上記方法は、重み付けされていない領域に位置するサブ領域にのみ適用可能であってもよい。

2．代替的に、上記方法は、重み付け領域に位置する特定のサブ領域にのみ適用可能であってもよい。

a．例えば、重み付け領域に位置する右上および／または左下のサブ領域に適用されてもよい。

b．例えば、重み付け領域に位置する左上および／または右下のサブ領域に適用されてもよい。

c．どのサブ領域を適用するかは、分割方向に依存してもよい。

b．一例において、`MvInfo1`および`MvInfo2`から双予測動きベクトルを導出し、記憶する。

i．`MvInfo1`および`MvInfo2`が異なる方向（ $L0$ または $L1$ ）の動きベクトルを有する場合、`MvInfo1`および`MvInfo2`を簡単に組み合わせて双予測動きベクトルを形成する。

ii．`MvInfo1`と`MvInfo2`の両方が同じ LX （ $X=0$ または 1 ）方向から来ている場合、

1． LX のために記憶されたMVは、`MvInfo1`および`MvInfo2`の1つの動きベクトルから、またはそれらの2つから（例えば、平均化によって）導出されてもよい。

2．`MvInfo1`または`MvInfo2`という参照ピクチャが LY （ $Y=1-X$ ）に含まれている場合、`MvInfo1`と`MvInfo2`とを簡単に組み合わせ、双予測動きベクトルを形成し、予測方向の1つを LY に設定する。

3．1つの動きベクトルを LY （ $Y=1-X$ ）における参照ピクチャにスケーリングし、スケーリングされたMVおよび他のMVを組み合わせて双予測動きベクトルを形成することができる。一例において、スケーリングされたMVが参照する LY におけるオブジェクト参照ピクチャは、予め規定されてもよく（例えば、参照ピクチャインデックスが0に等しい）、または信号通知されてもよい。

4．1つの動きベクトルを LY （ $Y=1-X$ ）における参照ピクチャにマッピングし、マッピングされたMVおよび他のMVを組み合わせて双予測動きベクトルを形成することができる。一例において、マッピングされたMVが参照する LY におけるオブジェクト参照ピクチャは、予め規定されてもよく（例えば、参照ピクチャインデックスが0に等しい）、または信号通知されてもよい。一例において、マッピング処理は、スケーリングせずに行われてもよい。例えば、マッピングされたMVは、元のMVと等しいかまたは

10

20

30

40

50

反対であってもよい。

i. 上記方法は、重み付け領域に位置するサブ領域にのみ適用可能である。

1) 代替的に、上記方法は、重み付けられていない領域に位置するサブ領域にのみ適用可能であってもよい。

2) 代替的に、上記方法は、重み付け領域に位置する特定のサブ領域にのみ適用可能であってもよい。

(a) 例えば、重み付け領域に位置する右上および／または左下のサブ領域に適用されてもよい。

(b) 例えば、重み付け領域に位置する左上および／または右下のサブ領域に適用されてもよい。

(c) どのサブ領域を適用するかは、分割方向に依存してもよい。

c. 単一または双予測動き情報を記憶するかまたは双予測動き情報を記憶するかは、2つの分割の復号化された動き情報に依存し得る。

i. 一例において、 $MvInfo1$ および $MvInfo2$ の両方が同じ LX 参照リストに含まれている場合、単一予測の動き情報を記憶してもよい。

ii. 一例において、 $MvInfo1$ および $MvInfo2$ が異なる参照リスト ($L0$ からの1つおよび $L1$ からの他の1つ) からのものである場合、双予測動き情報を記憶してもよい。

iii. 一例において、 $MvInfo1$ および $MvInfo2$ の両方が同じ LX 参照リストに含まれているが、 $MvInfo1$ または $MvInfo2$ が参照する参照ピクチャのうち少なくとも1つが他の参照リスト LY ($Y = 1 - X$) に含まれている場合、双予測動き情報を記憶してもよい。

d. 代替的に、動き情報の記憶は、依然として、1つのサブブロックが重み付け領域に位置するかどうかに基づいて行われる。重み付け領域に位置するものについて、記憶された動き情報は、以下の規則に従って導出される。

i. 一例において、 $MvInfo1$ および $MvInfo2$ が同じ LX ($X = 0$ または 1) 方向からのものである場合、

1. $MvInfo1$ または $MvInfo2$ のいずれか一方の参照ピクチャが LY ($Y = 1 - X$) に含まれている場合、 $MvInfo1$ および $MvInfo2$ を簡単に結合して双予測動きベクトルを形成し、予測方向の1つを LY に設定する。

2. 1つの動きベクトルを LY ($Y = 1 - X$) における参照ピクチャにスケーリングし、スケーリングされた MV および他の MV を組み合わせて双予測動きベクトルを形成することができる。一例において、スケーリングされた MV が参照する LY におけるオブジェクト参照ピクチャは、予め規定されてもよく (例えば、参照ピクチャインデックスが 0 に等しい)、または信号通知されてもよい。

3. 1つの動きベクトルを LY ($Y = 1 - X$) における参照ピクチャにマッピングし、マッピングされた MV および他の MV を組み合わせて双予測動きベクトルを形成することができる。一例において、マッピングされた MV が参照する LY におけるオブジェクト参照ピクチャは、予め規定されてもよく (例えば、参照ピクチャインデックスが 0 に等しい)、または信号通知されてもよい。一例において、マッピング処理は、スケーリングせずに行われてもよい。例えば、マッピングされた MV は、元の MV と等しいかまたは反対であってもよい。

ii. 一例において、 $MvInfo1$ および $MvInfo2$ が同じ LX ($X = 0$ または 1) 方向からのものである場合、 $MvInfo1$ または $MvInfo2$ のいずれかが記憶される。すなわち、単一予測動きベクトルを記憶する。

iii. 一例において、 $MvInfo1$ および $MvInfo2$ が同じ LX ($X = 0$ または 1) 方向からのものである場合、一方の動きベクトルを他方の参照ピクチャに対してスケーリングし、スケーリングされた動きベクトルと他方の動きベクトルとの平均または重み付け平均を記憶してもよい。

iv. 一例において、 $MvInfo1$ および $MvInfo2$ が同じ LX ($X = 0$ ま

10

20

30

40

50

たは1) 方向からのものであり、2つの動きベクトルの平均または重み付け平均を記憶してもよい場合、参照ピクチャは、 $MvIno f 1$ および $MvIn f o 2$ の2つの参照ピクチャのうちの1つであってもよい。

v. 一例において、ブロック全体のために記憶された動き情報は、各サブブロックにおける動き情報から導出されてもよい。例えば、すべてのまたは一部のサブブロックにおけるMVを重み付け加算し、ブロック全体の記憶されたMVを導出する。

vi. 上記の例では、異なるサブ領域は、異なる動き情報を記憶してもよい。

vii. 上記の例では、重み付けエリアにおけるサブ領域は、ユニ予測動き情報または双予測動き情報を記憶してもよい。

e. 代替的に、動き情報の記憶は、依然としてサブブロックの位置に基づく。しかしながら、各サブブロックに対して、単一予測の動き情報のみが記憶されてもよい。

i. 一例において、重み付け領域におけるサブ領域の動き情報は、 $MvIn f o 1$ および／または $MvIn f o 2$ に継承されてもよいし、または $MvIn f o 1$ および／または $MvIn f o 2$ から導出されてもよく、 $MvIn f o 2$ が記憶されてもよい。

f. 一例において、記憶された動き情報は、一部のモジュールにおいてのみ使用される。

i. 一例において、記憶された動き情報は、時間的動き予測に用いられる。

ii. 代替的に、記憶された動き情報は、空間的動き予測に用いられる。

iii. 代替的に、記憶された動き情報は、フィルタリング（例えば、デブロッキング）処理に用いられる。

g. 一例において、記憶された動き情報は、ブロックの動き補償処理において用いられてもよい。

h. 一例において、記憶された動き情報に基づいて、1つまたは複数のHMVPテーブルを更新してもよい。

21. 動き情報の記憶は、 $M \times N$ ブロック単位（ M および N は、同時に4に等しいことができた）に基づくことが提案される。各 $M \times N$ ブロックは、同じ動き情報を共有する。

a. 一例において、 M および N は8に設定される。

b. 一例において、 $M \times N$ ブロックの場合、その一部が重み付け領域に属する場合、その一部は非重み付け領域に属し、このようなブロックは、動きベクトル記憶装置のための重み付け領域の規則に従う。

c. 代替的に、 $M \times N$ ブロックの場合、その第1の部分が重み付け領域に属するが、その第2の部分が非重み付け領域に属する場合、このようなブロックは、非重み付け領域の規則に従い、動き情報を記憶してもよい。

22. 同じ参照ピクチャから2つの分割に対してTPMを予測することを有効にするかまたは無効にするかは、2つの分割の動きベクトルが十分に異なるような条件下にあることができる。

a. 一例において、2つの分割の参照サンプルは重複してはならない。

b. 一例において、 $abs(MV0[0] - MV1[0])$ は TH よりも小さく、 $MV0$ および $MV1$ は、2つの分割の動きベクトルであり、 $MVX[0]$ および $MVX[1]$ は、それぞれ MVX の水平および垂直成分である。関数 $abs(x)$ は x の絶対値を返す。

c. 一例において、 $abs(MV0[1] - MV1[1])$ は、 TH よりも小さい。

d. 一例において、 $abs(MV0[0] - MV1[0]) + abs(MV0[1] - MV1[1])$ は、 TH よりも小さい。

e. 一例において、 $Max(abs(MV0[0] - MV1[0]), abs(MV0[1] - MV1[1]))$ は、 TH よりも小さくなくならず、関数 $Max(x, y)$ は、 x および y のうちの大きい方を返す。

f. 1つの適合ビットストリームは、1つのブロックに対してTPMが有効にされた場合、1つ以上の前述の黒丸が真であることを満たす。

23. 開示される方法において、記憶された動き情報は、現在のピクチャまたは他の復

10

20

30

40

50

号対象のピクチャにおける、後続の復号対象のブロックの動き情報を予測するために用いられてもよい。

a. 記憶された動き情報は、以下を含んでもよい。

v i i . 動きベクトル

v i i i . 参照インデックス

i x . 単一予測または双予測の表示

x . インター予測方向の表示

x i . 一般化双予測 (G B i) の表示

x i i . 動きベクトルの解像度

x i i i . アフィン予測の表示

【0240】

5. 開示される技術の例示的な実施形態

5.1 実施形態#1

三角形マージモードにおける動きベクトル記憶処理

現在の符号化ブロックにおける水平および垂直方向の 4×4 ブロックの数を指定する変数 numSbX および numSbY は、 $\text{numSbX} = \text{cbWidth} \gg 2$ および $\text{numSbY} = \text{cbHeight} \gg 2$ に等しい。

cbWidth および cbHeight は、輝度サンプルにおける現在の符号化ブロックの幅および高さを特定する。

変数 minSb は、 $\min(\text{numSbX}, \text{numSbY}) - 1$ に等しく設定される。

変数 cbRatio は、以下のように導出される。

$\text{cbRatio} = (\text{cbWidth} > \text{cbHeight}) ? (\text{cbWidth} / \text{cbHeight}) : (\text{cbHeight} / \text{cbWidth})$

サブブロックインデックス (xSbIdx, ySbIdx) で、 $xSbIdx = 0 \dots \text{numSbX} - 1$ 、および $ySbIdx = 0 \dots \text{numSbY} - 1$ の各 4×4 サブブロックについて、以下が適用される。

— 変数 xIdx および yIdx は、以下のように導出される。

$xIdx = (\text{cbWidth} > \text{cbHeight}) ? (xSbIdx / \text{cbRatio}) : xSbIdx$

$yIdx = (\text{cbWidth} > \text{cbHeight}) ? ySbIdx : (ySbIdx / \text{cbRatio})$

— 変数 sType は、以下のように導出される。

— triangleDir が 0 に等しい場合、以下が適用される。

$sType = (xIdx == yIdx) ? 2 : ((xIdx > yIdx) ? 0 : 1)$

— そうでない場合 (triangleDir = 1)、以下が適用される。

$sType = (xIdx + yIdx == \text{minSb}) ? 2 : ((xIdx + yIdx < \text{minSb}) ? 0 : 1)$

triangleDir は、分割の方向を指定する。

図30A~30Cに示すように、sType = 0はP1領域に対応し、sType = 1はP2領域に対応し、sType = 2は重み付け領域に対応する。

P1領域の動き情報を (Mv1, refIdx1) とし、P2領域の動き情報を (Mv2, refIdx2) とする。

— sTypeの値によって、以下の割り当てが行われる。

— sTypeが0に等しい場合、 4×4 サブブロックの動き情報は、(Mv1, refIdx1) である。

— そうではなく、sTypeが1に等しいかまたはsTypeが2に等しい場合、且つMv1およびMv2の両方が同じ参照リストからのものであり、且つ現在のブロックが後方参照ピクチャを有するスライス内に位置する (slice.getCheckLDC() が偽である) 場合、 4×4 サブブロックの動き情報は、(Mv2, refIdx2) である。

10

20

30

40

50

－そうでない場合（sTypeが2に等しい）、以下が適用される。

```
refIdxL0 = (predListFlagA == 0) ? refIdx1 : refIdx2
```

```
refIdxL1 = (predListFlagA == 0) ? refIdx2 : refIdx1
```

```
mvL0 = (predListFlagA == 0) ? mv1 : mv2
```

```
mvL1 = (predListFlagA == 0) ? mv2 : mv1
```

predListFlagAは、P1領域の予測リストフラグである。

5.2 実施形態#2

サブブロックインデックス（xSbIdx, ySbIdx）で、xSbIdx=0..numSbX-1、およびySbIdx=0..numSbY-1の各4x4サブブロックについて、以下が適用される。

－sTypeの値によって、以下の割り当てが行われる。

－sTypeが0に等しい場合、4x4サブブロックの動き情報は、（mv1, refIdx1）である。

－そうではなく、sTypeが1に等しい場合、4x4サブブロックの動き情報は、（mv2, refIdx2）である。

－そうでない場合（sTypeが2に等しい）、以下が適用される。

```
refIdxL0 = (predListFlagA == 0) ? refIdx1 : refIdx2
```

```
refIdxL1 = (predListFlagA == 0) ? refIdx2 : refIdx1
```

```
mvL0 = (predListFlagA == 0) ? mv1 : mv2
```

```
mvL1 = (predListFlagA == 0) ? mv2 : mv1
```

predListFlagAは、P1領域の予測リストフラグである。

5.3 実施形態#3

サブブロックインデックス（xSbIdx, ySbIdx）で、xSbIdx=0..numSbX-1、およびySbIdx=0..numSbY-1の各4x4サブブロックについて、以下が適用される。

－sTypeの値によって、以下の割り当てが行われる。

－sTypeが0に等しい場合、4x4サブブロックの動き情報は、（mv1, refIdx1）である。

－そうではなく、sTypeが1に等しいかまたはsTypeが2に等しく、且つmv1およびmv2の両方が同じ参照リストからのものである場合、4x4サブブロックの動き情報は（mv2, refIdx2）である。

－そうでない場合（sTypeが2に等しい）、以下が適用される。

－mv1とmv2が異なる参照リストからのものである場合

```
refIdxL0 = (predListFlagA == 0) ? refIdx1 : refIdx2
```

```
refIdxL1 = (predListFlagA == 0) ? refIdx2 : refIdx1
```

```
mvL0 = (predListFlagA == 0) ? mv1 : mv2
```

```
mvL1 = (predListFlagA == 0) ? mv2 : mv1
```

predListFlagAは、P1領域の予測リストフラグである。

5.4 実施形態#4

サブブロックインデックス（xSbIdx, ySbIdx）で、xSbIdx=0..numSbX-1、およびySbIdx=0..numSbY-1の各4x4サブブロックについて、以下が適用される。

－sTypeの値によって、以下の割り当てが行われる。

－sTypeが0に等しいかまたはsTypeが2に等しい場合、且つmv1およびM

10

20

30

40

50

v 2の両方が同じ参照リストからのものであり、且つ現在のブロックが後方参照ピクチャを有するスライス内に位置する (s l i c e . g e t C h e c k L D C () が偽である) 場合、4 x 4サブブロックの動き情報は、(M v 1 , r e f I d x 1) である。

－そうではなく、s T y p e が1に等しい場合、4 x 4サブブロックの動き情報は、(M v 2 , r e f I d x 2) である。

－そうでない場合 (s T y p e が2に等しい) 、以下が適用される。

r e f I d x L 0 = (p r e d L i s t F l a g A == 0) ? r e f I d x 1 : r e f I d x 2

r e f I d x L 1 = (p r e d L i s t F l a g A == 0) ? r e f I d x 2 : r e f I d x 1

m v L 0 = (p r e d L i s t F l a g A == 0) ? M v 1 : M v 2

m v L 1 = (p r e d L i s t F l a g A == 0) ? M v 2 : M v 1

p r e d L i s t F l a g A は、P 1 領域の予測リストフラグである。

5. 5 実施形態 # 5

サブブロックインデックス (x S b I d x , y S b I d x) で、x S b I d x = 0 . . n u m S b X - 1、および y S b I d x = 0 . . n u m S b Y - 1 の各4 x 4サブブロックについて、以下が適用される。

－s T y p e の値によって、以下の割り当てが行われる。

－s T y p e が0に等しいか、またはs T y p e が2に等しい場合、且つM v 1 およびM v 2 の両方が同じ参照リストからのものである場合、4 x 4サブブロックの動き情報は、(M v 1 , r e f I d x 1) である。

－そうではなく、s T y p e が1に等しい場合、4 x 4サブブロックの動き情報は、(M v 2 , r e f I d x 2) である。

－そうでない場合 (s T y p e が2に等しい) 、以下が適用される。

－M v 1 とM v 2 が異なる参照リストからのものである場合

r e f I d x L 0 = (p r e d L i s t F l a g A == 0) ? r e f I d x 1 : r e f I d x 2

r e f I d x L 1 = (p r e d L i s t F l a g A == 0) ? r e f I d x 2 : r e f I d x 1

m v L 0 = (p r e d L i s t F l a g A == 0) ? M v 1 : M v 2

m v L 1 = (p r e d L i s t F l a g A == 0) ? M v 2 : M v 1

p r e d L i s t F l a g A は、P 1 領域の予測リストフラグである。

5. 6 実施形態 # 6

サブブロックインデックス (x S b I d x , y S b I d x) で、x S b I d x = 0 . . n u m S b X - 1、および y S b I d x = 0 . . n u m S b Y - 1 の各4 x 4サブブロックについて、以下が適用される。

－s T y p e の値によって、以下の割り当てが行われる。

－s T y p e が0に等しい場合、4 x 4サブブロックの動き情報は、(M v 1 , r e f I d x 1) である。

－そうではなく、s T y p e が1に等しい場合、4 x 4サブブロックの動き情報は、(M v 2 , r e f I d x 2) である。

－そうでない場合 (s T y p e が2に等しい) 、以下が適用される。

－M v 1 とM v 2 が異なる参照リストからのものである場合

r e f I d x L 0 = (p r e d L i s t F l a g A == 0) ? r e f I d x 1 : r e f I d x 2

r e f I d x L 1 = (p r e d L i s t F l a g A == 0) ? r e f I d x 2 : r e f I d x 1

m v L 0 = (p r e d L i s t F l a g A == 0) ? M v 1 : M v 2

m v L 1 = (p r e d L i s t F l a g A == 0) ? M v 2 : M v 1

p r e d L i s t F l a g A は、P 1 領域の予測リストフラグである。

－そうでない場合、 4×4 サブブロックの動き情報は、 $((Mv1 + Mv2) / 2, refIdx1)$ である。

5.7 実施形態#7

サブブロックインデックス $(xSbIdx, ySbIdx)$ で、 $xSbIdx = 0 \dots numSbX - 1$ 、および $ySbIdx = 0 \dots numSbY - 1$ の各 4×4 サブブロックについて、以下が適用される。

－ $sType$ の値によって、以下の割り当てが行われる。

－ $sType$ が0に等しい場合、 4×4 サブブロックの動き情報は、 $(Mv1, refIdx1)$ である。

－そうではなく、 $sType$ が1に等しい場合、 4×4 サブブロックの動き情報は、 $(Mv2, refIdx2)$ である。

－そうでない場合 ($sType$ が2に等しい)、 4×4 サブブロックの動き情報は、 $merge_triangle_idx0$ に対応する元のマージ候補の動きである。

5.8 実施形態#8

サブブロックインデックス $(xSbIdx, ySbIdx)$ で、 $xSbIdx = 0 \dots numSbX - 1$ 、および $ySbIdx = 0 \dots numSbY - 1$ の各 4×4 サブブロックについて、以下が適用される。

－ $sType$ の値によって、以下の割り当てが行われる。

－ $sType$ が0に等しいか、または $sType$ が2に等しい場合、および $Mv1$ と $Mv2$ の両方が同じ参照リストからのものであり、且つ $triangleDir$ が0に等しく、 $ySbIdx < numSbY - 1$ 、 4×4 サブブロックの動き情報は、 $(Mv1, refIdx1)$ である。

－そうではなく、 $sType$ が1に等しいかまたは $sType$ が2に等しく、且つ $Mv1$ および $Mv2$ の両方が同じ参照リストからのものであり、且つ ($triangleDir$ が0に等しく、 $ySbIdx = numSbY - 1$ または $triangleDir$ が1に等しい)である場合、 4×4 サブブロックの動き情報は、 $(Mv2, refIdx2)$ である。

－そうでない場合 ($sType$ が2に等しい)、以下が適用される。

－ $Mv1$ と $Mv2$ が異なる参照リストからのものである場合

$refIdxL0 = (predListFlagA == 0) ? refIdx1 : refIdx2$

$refIdxL1 = (predListFlagA == 0) ? refIdx2 : refIdx1$

$mvL0 = (predListFlagA == 0) ? Mv1 : Mv2$

$mvL1 = (predListFlagA == 0) ? Mv2 : Mv1$

$predListFlagA$ は、P1領域の予測リストフラグである。

5.9 実施形態#9

サブブロックインデックス $(xSbIdx, ySbIdx)$ で、 $xSbIdx = 0 \dots numSbX - 1$ 、および $ySbIdx = 0 \dots numSbY - 1$ の各 4×4 サブブロックについて、以下が適用される。

－ $Mv1$ と $Mv2$ の両方が同じ参照リストからのものであり、且つ現在のブロックが後方参照ピクチャを有するスライス内に位置する ($slice.getCheckLDC()$ が偽である) 場合、 4×4 サブブロックの動き情報は、 $(Mv2, refIdx2)$ である。

－そうでない場合、以下が適用される。

$refIdxL0 = (predListFlagA == 0) ? refIdx1 : refIdx2$

$refIdxL1 = (predListFlagA == 0) ? refIdx2 : refIdx1$

$mvL0 = (predListFlagA == 0) ? Mv1 : Mv2$

10

20

30

40

50

`mvL1 = (predListFlagA == 0) ? Mv2 : Mv1`
`predListFlagA`は、P1領域の予測リストフラグである。

5.10 実施形態#10

サブブロックインデックス (`xSbIdx`, `ySbIdx`) で、`xSbIdx = 0..numSbX-1`、および `ySbIdx = 0..numSbY-1` の各 4×4 サブブロックについて、以下が適用される。

`refIdxL0 = (predListFlagA == 0) ? refIdx1 : refIdx2`

`refIdxL1 = (predListFlagA == 0) ? refIdx2 : refIdx1`

`mvL0 = (predListFlagA == 0) ? Mv1 : Mv2`

`mvL1 = (predListFlagA == 0) ? Mv2 : Mv1`

`predListFlagA`は、P1領域の予測リストフラグである。

5.11 実施形態#11

サブブロックインデックス (`xSbIdx`, `ySbIdx`) で、`xSbIdx = 0..numSbX-1`、および `ySbIdx = 0..numSbY-1` の各 4×4 サブブロックについて、以下が適用される。

— `Mv1` と `Mv2` の両方が同じ参照リストからのものである場合、 4×4 サブブロックの動き情報は、(`Mv2`, `refIdx2`) である。

— そうでない場合、以下が適用される。

`refIdxL0 = (predListFlagA == 0) ? refIdx1 : refIdx2`

`refIdxL1 = (predListFlagA == 0) ? refIdx2 : refIdx1`

`mvL0 = (predListFlagA == 0) ? Mv1 : Mv2`

`mvL1 = (predListFlagA == 0) ? Mv2 : Mv1`

`predListFlagA`は、P1領域の予測リストフラグである。

5.12 実施形態#12

サブブロックインデックス (`xSbIdx`, `ySbIdx`) で、`xSbIdx = 0..numSbX-1`、および `ySbIdx = 0..numSbY-1` の各 4×4 サブブロックについて、以下が適用される。

— `Mv1` と `Mv2` の両方が同じ参照リストからのものであり、且つ現在のブロックが後方参照ピクチャを有するスライス内に位置する (`slice.getCheckLDC()` が偽である) 場合、 4×4 サブブロックの動き情報は、(`Mv1`, `refIdx1`) である。

— そうでない場合、以下が適用される。

`refIdxL0 = (predListFlagA == 0) ? refIdx1 : refIdx2`

`refIdxL1 = (predListFlagA == 0) ? refIdx2 : refIdx1`

`mvL0 = (predListFlagA == 0) ? Mv1 : Mv2`

`mvL1 = (predListFlagA == 0) ? Mv2 : Mv1`

`predListFlagA`は、P1領域の予測リストフラグである。

5.13 実施形態#13

サブブロックインデックス (`xSbIdx`, `ySbIdx`) で、`xSbIdx = 0..numSbX-1`、および `ySbIdx = 0..numSbY-1` の各 4×4 サブブロックについて、以下が適用される。

— `Mv1` と `Mv2` の両方が同じ参照リストからのものである場合、 4×4 サブブロックの動き情報は、(`Mv1`, `refIdx1`) である。

— そうでない場合、以下が適用される。

10

20

30

40

50

－Mv1とMv2が異なる参照リストからのものである場合

```
refIdxL0 = (predListFlagA == 0) ? refIdx1 : refIdx2
```

```
refIdxL1 = (predListFlagA == 0) ? refIdx2 : refIdx1
```

```
mvL0 = (predListFlagA == 0) ? Mv1 : Mv2
```

```
mvL1 = (predListFlagA == 0) ? Mv2 : Mv1
```

predListFlagAは、P1領域の予測リストフラグである。

5.14 実施形態#14

サブブロックインデックス (xSbIdx, ySbIdx) で、xSbIdx = 0 . . numSbX - 1、および ySbIdx = 0 . . numSbY - 1 の各 4 x 4 サブブロックについて、以下が適用される。

－Mv1とMv2が異なる参照リストからのものである場合

```
refIdxL0 = (predListFlagA == 0) ? refIdx1 : refIdx2
```

```
refIdxL1 = (predListFlagA == 0) ? refIdx2 : refIdx1
```

```
mvL0 = (predListFlagA == 0) ? Mv1 : Mv2
```

```
mvL1 = (predListFlagA == 0) ? Mv2 : Mv1
```

predListFlagAは、P1領域の予測リストフラグである。

－そうでない場合、4 x 4 サブブロックの動き情報は、((Mv1 + Mv2) / 2, refIdx1) である。

5.15 実施形態#15

xSbIdx = 0 . . numSbX - 1、および ySbIdx = 0 . . numSbY - 1 であるサブブロックインデックス (xSbIdx, ySbIdx) の各 4 x 4 サブブロックごとに、4 x 4 サブブロックの動き情報は、merge_triangle_idx0 に対応する元のマージ候補の動きである。

5.16 実施形態#16

サブブロックインデックス (xSbIdx, ySbIdx) で、xSbIdx = 0 . . numSbX - 1、および ySbIdx = 0 . . numSbY - 1 の各 4 x 4 サブブロックについて、以下が適用される。

－Mv1とMv2の両方が同じ参照リストからのものであり、且つ triangleDir が 0 に等しく、ySbIdx < numSbY - 1 の場合、4 x 4 サブブロックの動き情報は、(Mv1, refIdx1) である。

－そうではなく、Mv1およびMv2の両方が同じ参照リストからのものである (triangleDir が 0 に等しく、ySbIdx = numSbY - 1 または triangleDir が 1 に等しい) 場合、4 x 4 サブブロックの動き情報は (Mv2, refIdx2) である。

－そうでない場合、以下が適用される。

```
refIdxL0 = (predListFlagA == 0) ? refIdx1 : refIdx2
```

```
refIdxL1 = (predListFlagA == 0) ? refIdx2 : refIdx1
```

```
mvL0 = (predListFlagA == 0) ? Mv1 : Mv2
```

```
mvL1 = (predListFlagA == 0) ? Mv2 : Mv1
```

predListFlagAは、P1領域の予測リストフラグである。

【0241】

図31は、映像処理方法3100のフローチャートである。方法3100は、工程3102において、現在の映像ブロックと映像のビットストリーム表現との間で変換するため、動き候補を動き候補リストに挿入する順序を、優先規則に基づいて決定することを含

10

20

30

40

50

み、現在のブロックは、ジオメトリ分割モード（`geometry partition mode`）を使用して符号化される。

【0242】

方法3100は、工程3104において、決定および動き候補リストに基づいて変換を行うことを含む。

【0243】

図32は、映像処理方法3200のフローチャートである。方法3200は、工程3202において、現在の映像ブロックと映像のビットストリーム表現との間で変換するための動き候補リストに、1つの正規の動き候補から導出された`List0`および`List1`からの1つ以上の平均化された単一予測動き候補を挿入することを含み、現在のブロックは、ジオメトリ分割モードを使用して符号化される。

10

【0244】

方法3200は、工程3204において、動き候補リストに基づいて変換を行うことを含む。

【0245】

図33は、映像処理方法3300のフローチャートである。方法3300は、工程3302において、現在の映像ブロックと映像のビットストリーム表現との間で変換するための動き候補リストに、単一予測を有する正規の動き候補から導出された単一予測を有する1つ以上の平均化動き候補を挿入することを含み、現在のブロックは、ジオメトリ分割モードを使用して符号化される。

20

【0246】

方法3300は、工程3304において、動き候補リストに基づいて変換を行うことを含む。

【0247】

図34は、映像処理方法3400のフローチャートである。方法3400は、工程3402において、現在の映像ブロックと映像のビットストリーム表現との間で変換するための動き候補リストに、利用可能な動き候補から導出された1つ以上の仮想動き候補を挿入することを含み、現在のブロックは、ジオメトリ分割モードを使用して符号化される。

【0248】

方法3400は、工程3404において、動き候補リストに基づいて変換を行うことを含む。

30

【0249】

図35は、映像処理方法3500のフローチャートである。方法3500は、工程3502において、優先規則に基づいて、ジオメトリ分割モードを使用して符号化された現在の映像ブロックと映像のビットストリーム表現との間で変換を行うために、動き候補を動き候補リストに挿入する順序を決定することであって、この優先規則は、複数の動き候補のうちの1つの動き候補を導出する位置またはこの動き候補に関連付けられた1つ以上の参照ピクチャリストに基づく、決定することを含む。

【0250】

方法3500は、工程3504において、決定および動き候補リストに基づいて、変換を行うことを含む。

40

【0251】

図36は、映像処理方法3600のフローチャートである。方法3600は、工程3602において、現在の映像ブロックと映像のビットストリーム表現との間で変換するために、動き候補を動き候補リストに挿入する順序を優先規則に基づいて決定することを含み、現在のブロックは、ジオメトリ分割モードを使用して符号化され、順番または挿入は、シーケンス間、ピクチャ間、スライス間、タイルグループ間、または現在のブロックから映像の次のブロックに亘って変更可能である。

【0252】

方法3600は、工程3604において、決定および動き候補リストに基づいて、変換

50

を行うことを含む。

【0253】

図37は、映像処理方法3700のフローチャートである。方法3700は、工程3702において、ジオメトリ分割モードを使用して符号化された現在の映像ブロックを複数の分割に分割することを含む。

【0254】

方法3700は、工程3704において、複数の動き候補リストを構築することを含み、各動き候補リストは複数の分割の各々に対応する。

【0255】

方法3700は、工程3706において、複数の動き候補リストに基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことを含む。

10

【0256】

図38は、映像処理方法3800のフローチャートである。方法3800は、工程3802において、ジオメトリ分割モードを使用して符号化された現在の映像ブロックを複数の分割に分割することを含む。

【0257】

方法3800は、動作3804において、複数の動き候補リストを構築することを含み、複数の分割の各々は、複数の動き候補リストのうち同じ動き候補リストを選択するように構成される。

【0258】

方法3800は、工程3806において、選択された同じ動き候補リストに基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことを含む。

20

【0259】

図39は、映像処理方法3900のフローチャートである。方法3900は、工程3902において、ビデオユニットにおける現在の映像ブロックと映像のビットストリーム表現との間で変換を行うことを含み、ビットストリーム表現は、映像領域において有効にされるジオメトリ分割モードのための動き候補リストにおける許容される動き候補の最大数を示すフィールドを含む。

【0260】

図40は、映像処理方法4000のフローチャートである。方法4000は、ステップ4002において、ジオメトリ分割モードを利用して複数の分割に分割される現在の映像ブロックのために、複数の重み係数群から1つの重み係数群を選択し、現在のブロックの、少なくとも幅または高さに基づいて1つの重み係数群を選択することを含む。

30

【0261】

方法4000は、動作4004において、現在のブロックと映像のビットストリーム表現との間の変換の一部として、複数の分割のうちの少なくとも2つの共通の境界に沿ったサンプルに重み係数群を適用することを含む。

【0262】

図41は、映像処理方法4100のフローチャートである。方法4100は、動作4102において、ジオメトリ分割モードを使用して複数の分割に分割される現在の映像ブロック内のサンプルについて、複数の分割のうちの少なくとも2つの共通の境界の角度に基づいて、少なくとも1つの重み係数を判定することを含む。

40

【0263】

方法4100は、ステップ4104において、少なくとも1つの重み係数に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことを含む。

【0264】

図42は、映像処理方法4200のフローチャートである。方法4200は、動作4202において、ジオメトリ分割モードを使用して複数の分割に分割された現在の映像ブロックに対して、複数の分割のうち少なくとも2つの共通の境界に沿ったサンプルに対して、4×4個のサンプルサイズとは異なるサンプルサイズの動き補償処理を行うことを含む。

50

【0265】

方法4200は、動作4204において、動き補償処理に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことを含む。

【0266】

図43は、映像処理方法4300のフローチャートである。方法4300は、動作4302において、ジオメトリ分割モードを使用して、現在の映像ブロックを複数の分割に分割することを含む。

【0267】

方法4300は、動作4304において、複数の分割に関連付けられた動き情報から導出された、現在のブロック内の $K \times L$ 領域のための1つの動き情報のセットを記憶することを含む。

10

【0268】

方法4300は、ステップ4306において、複数の分割のうち少なくとも1つの分割に関連付けられた少なくとも1つの動き情報のセットを使用して、現在のブロックと映像のビットストリーム表現との間で変換を行うことを含む。

【0269】

図44は、映像処理方法4400のフローチャートである。方法4400は、動作4402において、現在の映像ブロックを複数の分割に分割することを含む。

【0270】

方法4400は、動作4404において、ジオメトリ分割モードを使用して符号化された現在のブロックと映像のビットストリーム表現との間で変換を行うことを含み、現在のブロックの各 $M \times N$ ブロックユニットごとに1つの動き情報のセットが記憶され、この1つの動き情報のセットは、複数の分割の各々に関連付けられた動き情報から導出され、 M または N のいずれかが4に等しくなく、 $M \times N$ ブロックユニットにおける各サンプルが同じ動き情報を共有する。

20

【0271】

図45は、映像処理方法4500のフローチャートである。方法4500は、動作4502において、現在の映像ブロックを複数の分割に分割することを含む。

【0272】

方法4500は、動作4504において、現在のブロックと映像のビットストリーム表現との間の変換の一部として、記憶されるべき第1のサブ領域に関連付けられた第2の動き情報とは異なる第1の動き情報に基づいて、現在のブロックの第1のサブ領域に対して動き補償処理を行うことを含む。

30

【0273】

図46は、映像処理方法4600のフローチャートである。方法4600は、動作4602において、ジオメトリ分割モードを使用して、現在の映像ブロックを複数の分割に分割することを含む。

【0274】

方法4600は、動作4604において、複数の分割に関連付けられた分割方向または復号化されたマージンデックスまたはマージ候補リストに基づいて、第1のサンプルセットおよび第2のサンプルセットの動き情報を記憶することを含み、第1のサンプルセットは、複数の分割のうちの少なくとも2つの共通の境界上にあり、第2のサンプルセットは、複数の分割のうちの少なくとも2つの内部にある。

40

【0275】

方法4600は、動作4606において、記憶された動き情報に基づいて、現在のブロックと映像のビットストリーム表現との間の変換、または後続の映像のブロックとビットストリーム表現との間で変換を行うことを含む。

【0276】

図47は、映像処理方法4700のフローチャートである。方法4700は、動作4702において、現在の映像ブロックを複数の分割に分割することを含む。

50

【0277】

方法4700は、動作4704において、複数の分割のうちの少なくとも2つの分割の動き情報に基づく仮想双予測動き情報を記憶することを含む。

【0278】

方法4700は、ステップ4706において、仮想双予測動き情報に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことを含む。

【0279】

図48は、映像処理方法4800のフローチャートである。方法4800は、動作4802において、現在の映像ブロックを複数の分割に分割することを含む。

【0280】

方法4800は、ステップ4804において、複数の分割の第1の分割(MvInfo1)の動き情報および複数の分割の第2の分割(MvInfo2)の動き情報を、低遅延チェックフラグに基づいて記憶することを含み、低遅延チェックフラグは、現在のブロックを構成する現在のピクチャのPOC値以下のピクチャオーダカウンタ(POC)値を有するすべての参照ピクチャを示す。

【0281】

方法4800は、ステップ4806において、記憶された動き情報に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことを含む。

【0282】

いくつかの実施形態において、次の技術案を実装することができる。

A1. 優先規則に基づいて、現在の映像ブロックとこの映像のビットストリーム表現との間で変換を行うために、動き候補を動き候補リストに挿入する順序を決定し、現在のブロックをジオメトリ分割モードを使用して符号化することと、この決定およびこの動き候補リストに基づいて、この変換を行うことと、を含む、映像処理方法。

A2. 優先規則は、動き候補のうちの少なくとも1つの動き候補の予測方向に基づく、案A1に記載の方法。

A3. 挿入する順序は、リストYからの単一予測を伴う任意の正規の動き候補を挿入する前に、リストXからの単一予測を伴う正規の動き候補を挿入することを含む、案A2に記載の方法。

A4. 挿入する順序は、単一予測を有する任意の動き候補を挿入する前に、双予測を有する動き候補を挿入することを含む、案A2に記載の方法。

A5. 挿入する順序は、双予測を伴う正規の動き候補から導出されたリストXの動き候補、双予測を伴う正規の動き候補から導出されたリストXの動き候補、単一予測を伴う正規の動き候補の順で定義され、 $X = 0$ または $X = 1$ である、案A2に記載の方法。

A6. 挿入する順序は、第1の正規の動き候補から導出された動き候補を双予測挿入することと、その後第2の正規の動き候補から導出された動き候補を双予測挿入することと、を含み、現在のブロックの挿入する順序は、非ジオメトリ分割モードを使用して符号化された映像ブロックの挿入する順序と同一である、A2に記載の方法。

A7. 挿入する順序は、第1の双予測正規の動き候補から導出された動き候補と、第2の双予測正規の動き候補から導出された動き候補とをインターリーブ方式で挿入することを含む、案A2に記載の方法。

A8. インターリーブ方式は、CAから導出された短縮List0予測候補、CBから導出された短縮List1予測候補、CAから導出された短縮List1予測候補、およびCBから導出された短縮List0予測候補を挿入することを含み、CAおよびCBは2つの正規の動き候補である、案A7に記載の方法。

A9. インターリーブ方式は、CAから導出されたList1で短縮List1予測候補、CBから導出された短縮List0予測候補、CAから導出された短縮List1予測候補、およびCBから導出された短縮List1を挿入することを含み、CAおよびCBは2つの正規の動き候補である、案A7に記載の方法。

A10. 正規の動き候補は、非ジオメトリ分割モードを使用して符号化された映像のブロ

10

20

30

40

50

ックに対して導出された動き候補である、案 A 1 ~ A 9 のいずれか 1 案に記載の方法。

A 1 1 . 優先規則は、正規の動き候補に関連付けられた符号化モード情報に基づく、案 A 1 に記載の方法。

A 1 2 . 符号化モード情報は、マージモードまたは高度動きベクトル予測 (AMVP) モードを使用する符号化を含む、案 A 1 1 に記載の方法。

A 1 3 . 挿入する順序は、マージモードで符号化ブロックから導出された動き候補を挿入する前に、AMVP モードで符号化ブロックから導出された動き候補を挿入することを含む、案 A 1 2 に記載の方法。

A 1 4 . 挿入する順序は、AMVP モードで符号化ブロックから導出された動き候補を挿入する前に、マージモードで符号化ブロックから導出された動き候補を挿入することを含む、案 A 1 2 に記載の方法。

10

A 1 5 . 符号化モード情報は、参照インデックスまたはピクチャオーダカウンタ (POC) 差を含む、案 A 1 1 に記載の方法。

A 1 6 . 挿入する順序は、第 1 の参照インデックスに関連付けられた動き候補を挿入することと、その後、第 1 の参照インデックスよりも大きい第 2 の参照インデックスに関連付けられた動き候補を挿入することを含む、案 A 1 5 に記載の方法。

A 1 7 . 挿入する順序は、第 1 の POC 差に関連付けられた動き候補を挿入することと、その後、第 1 の POC 差よりも大きい第 2 の POC 差参照インデックスに関連付けられた動き候補を挿入することを含む、案 A 1 5 に記載の方法。

A 1 8 . 符号化モード情報は、現在のブロックを含むピクチャ、スライスまたはタイルグループに関連付けられた参照ピクチャの量子化パラメータ (QP) または時間層インデックスを含む、案 A 1 1 に記載の方法。

20

A 1 9 . 挿入する順序は、第 1 の QP に関連付けられた動き候補を挿入することと、その後、第 1 の QP よりも大きい第 2 の QP に関連付けられた動き候補を挿入することを含む、案 A 1 8 に記載の方法。

A 2 0 . 挿入する順序は、すべての短縮 List 1 - predicted 候補を、いずれかの短縮 List 0 - predicted 候補の前に挿入することを含む、案 A 1 に記載の方法。

A 2 1 . 挿入する順序は、利用可能な幾何学的動き候補に基づいて、1 つ以上の短縮 List 0 予測候補および 1 つ以上の短縮 List 1 予測候補を挿入することを含む、案 A 1 に記載の方法。

30

A 2 2 . 挿入する順序は、利用可能な幾何学的動き候補に基づいて、双予測を伴う第 1 の動き候補を挿入した後、双予測を伴う第 2 の動き候補を挿入することを含む、案 A 1 に記載の方法。

A 2 3 . 現在の映像ブロックと該映像のビットストリーム表現との間で変換を行うための動き候補リストに、1 つの正規の動き候補から導出された、List 0 と List 1 からの 1 つ以上の平均化された単一予測動き候補を挿入することを含み、現在のブロックは、ジオメトリ分割モードを使用して符号化する、挿入することと、この動き候補リストに基づいて、この変換を行うことと、を含む、映像処理方法。

A 2 4 . 動き候補リストへの挿入のために、1 つ以上の平均化された単一予測動き候補から 1 つの平均化された単一予測動き候補を選択することをさらに含む、案 A 2 3 に記載の方法。

40

A 2 5 . 選択することは、動き候補リストにおける利用可能な幾何学的動き候補に基づく、案 A 2 4 に記載の方法。

A 2 6 . 選択することは、List 0 および List 1 の参照ピクチャインデックスに基づく、案 A 2 4 に記載の方法。

A 2 7 . 選択することは、List 0 および List 1 のための参照ピクチャと現在のピクチャとの間のピクチャオーダカウンタ (POC) 距離に基づく、案 A 2 4 に記載の方法。

A 2 8 . List 1 からの任意の平均化された単一予測動き候補を挿入する前に、List 0 からのすべての平均化された単一予測動き候補を挿入する、案 A 2 3 に記載の方法。

50

A 2 9 . L i s t 0からの任意の平均化された単一予測動き候補を挿入する前に、L i s t 1からのすべての平均化された単一予測動き候補を挿入する、案 A 2 3 に記載の方法。

A 3 0 . 現在の映像ブロックと該映像のビットストリーム表現との間で変換を行うための動き候補リストに、単一予測を伴う正規の動き候補から導出された、単一予測を伴う1つ以上の平均化動き候補を挿入することを含み、現在のブロックは、ジオメトリ分割モードを使用して符号化することと、この動き候補リストに基づいて、この変換を行うことと、を含む、映像処理方法。

A 3 1 . リスト L Xからの単一予測を伴う平均化動き候補が、リスト L Xからの単一予測を伴う正規の動き候補から導出され、 $X = 0$ または $X = 1$ である、案 A 3 0 に記載の方法。

A 3 2 . リスト L Yからの単一予測を伴う平均動き候補が、スケーリングされた正規の動き候補から導出され、リスト L Xからの正規の動き候補がリスト L Yにスケーリングされて、スケーリングされた正規の動き候補を生成し、 $X = 0$ または $X = 1$ であり、 $Y = 1 - X$ である、案 A 3 0 に記載の方法。

A 3 3 . 現在の映像ブロックと映像のビットストリーム表現との間で変換を行うための動き候補リストに、利用可能な動き候補から導出された1つ以上の仮想動き候補を挿入することであって、現在のブロックは、ジオメトリ分割モードを使用して符号化されている、挿入することと、動き候補リストに基づいて、この変換を行うことと、を含む、映像処理方法。

A 3 4 . 1つ以上の仮想動き候補は、まず、動き候補リストにおける動き候補の短縮リストを生成することによって導出される、案 A 3 3 に記載の方法。

A 3 5 . 1つ以上の仮想動き候補は、単一予測を有する動き候補のみに基づいて導出される、案 A 3 3 に記載の方法。

A 3 6 . 1つ以上の仮想動き候補が、単一予測された動き候補と、リスト 1 から予測された動き候補の短縮リストとにのみ基づいて導出される、案 A 3 3 の方法。

A 3 7 . 所定の参照ピクチャインデックスを有するか、またはピクチャオーダカウンタ (P O C) 距離が一定の範囲内にある動き候補に基づいて、1つ以上の仮想動き候補を導出する、案 A 3 3 に記載の方法。

A 3 8 . プルーニング操作を適用せずに、1つ以上の仮想動き候補を導出する、案 A 3 3 に記載の方法。

A 3 9 . 案 A 1 ~ A 3 8 のいずれか1案に記載の方法において、現在のブロックは、分割パターンに基づいて複数の分割に分割される。

A 4 0 . 複数の分割のうちの少なくとも1つの分割が、非正方形であり、かつ非長方形である、案 A 1 ~ A 3 9 のいずれかに記載の方法。

A 4 1 . ジオメトリ分割モードは、三角形分割モードを含む、案 A 1 ~ A 3 9 のいずれか1案に記載の方法

A 4 2 . 変換は、ビットストリーム表現から現在のブロックを生成する、案 A 1 ~ A 4 1 のいずれか1案に記載の方法。

A 4 3 . 変換は、現在のブロックからビットストリーム表現を生成する、案 A 1 ~ A 4 1 のいずれか1案に記載の方法。

A 4 4 . 処理装置と、命令を搭載した非一時的メモリとを含む、映像システムにおける装置であって、処理装置が実行する命令は、処理装置に、解決策 A 1 ~ A 4 3 のいずれか1案に記載の方法を実装させる映像システムにおける装置。

A 4 5 . 非一時的なコンピュータ可読媒体に記憶されたコンピュータプログラム製品であって、案 A 1 ~ A 4 3 のいずれか1案に記載の方法を実行するためのプログラムコードを含む、コンピュータプログラム製品。

【 0 2 8 3 】

いくつかの実施形態において、次の技術案を実装することができる。

B 1 . ジオメトリ分割モードを使用して符号化された映像の現在のブロックと映像のビットストリーム表現との間で変換を行うための動き候補リストへの動き候補の挿入する順序について、優先規則に基づいて決定することであって、優先規則が、動き候補の動き候補

10

20

30

40

50

が導出された位置、または動き候補に関連付けられた1つ以上の参照ピクチャリストに基づくものである、決定することと、決定および動き候補リストに基づいて、変換を行うことを含む、映像処理方法。

B 2 . 第 1 の動き候補 C A がブロック A から導出されたものである場合、第 2 の動き候補 C B がブロック B から導出され、ブロック A の後にブロック B がチェックされる、挿入する順序は、C A から導出された動き候補を動き候補リストに挿入した後に、C B から導出された動き候補を挿入することを含む、案 B 1 に記載の映像処理方法。

B 3 . 正規の動き候補リストの構築処理において、ブロック A の後のブロック B をチェックすることをさらに含む、案 B 2 に記載の方法。

B 4 . 挿入する順序は、C A から導出されたリスト X からの予測を有する動き候補を挿入した後、C B から導出されたリスト X からの予測を有する動き候補を挿入することをさらに含み、 $X = 0$ または $X = 1$ である案 B 2 の方法。

B 5 . 挿入する順序は、C A から導出されたリスト Y からの予測を有する動き候補を挿入した後、C B から導出されたリスト Y からの予測を有する動き候補を挿入することをさらに含み、 $X = 0$ または $X = 1$ であり、 $Y = (1 - X)$ である、案 B 2 に記載の方法。

B 6 . 挿入する順序は、C A から導出されたリスト Y からの予測を有する動き候補を挿入した後、C B から導出されたリスト X からの予測を有する動き候補を挿入することをさらに含み、 $X = 0$ または $X = 1$ であり、 $Y = (1 - X)$ である、案 B 2 に記載の方法。

B 7 . 現在のブロックを分割パターンに基づいて複数の分割に分割する、案 B 1 ~ B 6 のいずれか 1 つに記載の方法。

B 8 . ジオメトリ分割モードは、三角形分割モードを含む、案 B 1 ~ B 6 のいずれか 1 案に記載の方法。

B 9 . 現在のブロックが、非正方形および非長方形である少なくとも 1 つの分割を有する複数の分割に分割される、案 B 1 ~ B 6 のいずれかに記載の方法。

B 10 . 変換は、ビットストリーム表現から現在のブロックを生成する、案 B 1 ~ B 9 のいずれか 1 案に記載の方法。

B 11 . 変換は、現在のブロックからビットストリーム表現を生成する、案 B 1 ~ B 9 のいずれか 1 案に記載の方法。

B 12 . 処理装置と、命令を搭載した非一時的メモリとを備え、映像システムにおける装置であって、処理装置が実行する命令は、処理装置に、案 B 1 ~ B 11 のいずれか 1 案に記載の方法を実装させることを特徴とする装置。

B 13 . 非一時的なコンピュータ可読媒体に記憶されたコンピュータプログラム製品であって、コンピュータプログラム製品は、案 B 1 ~ B 11 のいずれか 1 案に記載の方法を実行するためのプログラムコードを含む。

【0284】

いくつかの実施形態において、次の技術案を実装することができる。

C 1 . 現在の映像ブロックと映像のビットストリーム表現との間で変換するために、動き候補を動き候補リストに挿入する順序を優先規則に基づいて決定することであって、現在のブロックは、ジオメトリ分割モードを使用して符号化され、順番または挿入は、シーケンス間、ピクチャ間、スライス間、タイルグループ間、または現在のブロックから映像の次のブロックに亘って変更可能である。そして、決定および動き候補リストに基づいて、この変換を行うことを含む、映像処理方法。

C 2 . 挿入する順序は、現在のブロック、または現在のブロックを含むシーケンス、ピクチャ、スライス若しくはタイルグループに基づく、案 C 1 に記載の方法。

C 3 . 優先規則は、現在のブロックのサイズ、形状または分割パターンのうちの少なくとも 1 つに基づく、案 C 2 に記載の方法。

C 4 . 挿入する順序は、シーケンスパラメータセット (SPS)、映像パラメータセット (VPS)、ピクチャパラメータセット (PPS)、スライスヘッダ、タイルグループヘッダ、タイル、符号化ツリーユニット (CTU)、またはビットストリーム表現における符号化ユニット (CU) において、エンコーダからデコーダに信号通知される、案 C 1 ~

10

20

30

40

50

C 3 のいずれか 1 案に記載の方法。

C 5 . ジオメトリ分割モードを使用して符号化された現在の映像ブロックを複数の分割に分割することと、複数の分割の各々に対応する複数の動き候補リストを構築することと、複数の動き候補リストに基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む映像処理方法。

C 6 . 複数の分割のうちの第 1 の分割のための複数の動き候補リストのうちの第 1 のリストは、リスト X から予測された動き候補のみを含み、複数の分割のうちの第 2 の分割のための複数の動き候補リストのうちの第 2 のリストは、リスト Y から予測された動き候補のみを含み、 $X = 0$ または $X = 1$ であり、 $Y = (1 - X)$ である、案 C 5 に記載の方法。

C 7 . 複数の動き候補リストに挿入される 1 つ以上の動き候補は、異なる空間的ブロックまたは時間的ブロックに関連付けられた動き情報に基づく、案 C 5 に記載の方法。

C 8 . 複数の分割の第 1 の分割のための 1 つ以上の動き候補を導出するために使用される異なる空間的または時間的ブロックの位置は、現在のブロックにおける複数の分割の第 1 の分割または第 2 の分割の位置に基づく、案 C 7 に記載の方法。

C 9 . 複数の分割の第 1 の分割または第 2 の分割のための 1 つ以上の動き候補を導出するために使用される異なる空間的または時間的ブロックの位置は、現在のブロックの分割パターンに基づく、案 C 7 に記載の方法。

C 10 . 分割パターンは、それぞれ 45° の分割パターンまたは 135° の分割パターンで表される、右上隅から左下隅へ、または左上隅から右下隅への分割を含む、案 C 9 に記載の方法。

C 11 . 分割パターンは、 135° の分割パターンを含み、方法は、現在のブロックの右上隅を含む分割について、より多くのブロックをチェックすることをさらに含む、案 C 10 に記載の方法。

C 12 . 分割パターンは、 135° の分割パターンを含み、方法は、現在のブロックの左下隅を含む区画について、より多くの左ブロックをチェックすることをさらに含む、案 C 10 に記載の方法。

C 13 . 複数の分割に対応する複数の動き候補リストの各々のためのインデックスを信号通知する、案 C 5 に記載の方法。

C 14 . 複数の動き候補リストの各々のためのインデックスを含むビットストリーム表現は、変換のために構文解析される、案 C 5 に記載の方法。

C 15 . ジオメトリ分割モードを使用して符号化された現在の映像ブロックを複数の分割に分割することと、複数の動き候補リストを構築することであって、複数の分割の各々は、複数の動き候補リストのうち同じ動き候補リストを選択するように構成される、構築することと、選択された同じ動き候補リストに基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む映像処理方法。

C 16 . ビットストリーム表現は、複数の動き候補リストの第 1 の動き候補リストのインデックスを含む、案 C 15 に記載の方法。

C 17 . ビットストリーム表現は、第 1 の動き候補リストからの動き候補のインデックスをさらに含み、動き候補のインデックスは、第 1 の動き候補リストのインデックスの後に信号通知される、案 C 16 に記載の方法。

C 18 . ビットストリーム表現は、第 1 の動き候補リストのインデックスに続く、第 1 の動き候補リストからの動き候補のインデックスをさらに含み、ビットストリーム表現は、変換のために構文解析される、案 C 16 に記載の方法。

C 19 . 第 1 の動き候補リストのインデックスを、ビットストリーム表現における第 1 の動き候補リストからの動き候補のインデックスと結合して符号化する、案 C 15 に記載の方法。

C 20 . 第 1 の動き候補リストからの動き候補のインデックスと結合して符号化された第 1 の動き候補リストのインデックスを含むビットストリームを、変換のために構文解析する、案 C 15 に記載の方法。

C 21 . 少なくとも 1 つの分割パターンに基づいて、現在のブロックを複数の区画に分割

10

20

30

40

50

する、案 C 1 ~ C 2 0 のいずれか 1 つに記載の方法。

C 2 2 . ジオメトリ分割モードは三角形分割モードを含む、案 C 1 ~ C 2 1 のいずれか 1 案に記載の方法。

C 2 3 . 現在のブロックを、少なくとも 1 つの区画が非正方形かつ非長方形である複数の区画に分割する、案 C 1 ~ C 2 1 のいずれかに記載の方法。

C 2 4 . 変換は、ビットストリーム表現から現在のブロックを生成する、案 C 1 ~ C 2 3 のいずれか 1 案に記載の方法。

C 2 5 . 変換は、現在のブロックからビットストリーム表現を生成する、案 C 1 ~ C 2 3 のいずれか 1 案に記載の方法。

C 2 6 . 処理装置と、命令を搭載した非一時的メモリとを含む映像システムにおける装置であって、処理装置が実行する命令は、処理装置に、案 C 1 ~ C 2 5 のいずれか 1 案に記載の方法を実装させることを特徴とする装置。

C 2 7 . 非一時的なコンピュータ可読媒体に記憶されたコンピュータプログラム製品であって、コンピュータプログラム製品は、案 C 1 ~ C 2 5 のいずれか 1 案に記載の方法を実行するためのプログラムコードを含む。

【 0 2 8 5 】

いくつかの実施形態において、次の技術案を実装することができる。

D 1 . ビデオユニットにおける現在の映像ブロックと映像のビットストリーム表現との間で変換を行うことを含み、ビットストリーム表現は、映像領域において有効にされるジオメトリ分割モードのための動き候補リストにおける許容される動き候補の最大数を示すフィールドを含む、映像処理方法。

D 2 . フィールドは、ビットストリーム表現における許容される動き候補の最大数の表示の明確な信号通知を含む、案 D 1 に記載の方法。

D 3 . フィールドは、許容される動き候補の最大数が、非ジオメトリ分割モードを使用して符号化される別のブロックの動き候補リストにおける許容される動き候補の最大数に等しいことを暗黙的に示す、案 D 1 に記載の方法。

D 4 . 案 D 1 ~ D 3 のいずれか 1 つに記載の方法において、現在のブロックは、マージモードまたは高度動きベクトル予測 (A M V P) モードを使用してさらに符号化される。

D 5 . 映像ユニットは、シーケンス、映像、ピクチャ、スライス、タイルグループ、最大符号化ユニット (L C U) 行、または最大符号化ユニット (L C U) のグループを含む、案 D 1 ~ D 3 のいずれかに記載の方法。

D 6 . フィールドにおける許容される動き候補の最大数の表示は、シーケンスパラメータセット (S P S) 、映像パラメータセット (V P S) 、ピクチャパラメータセット (P P S) 、ピクチャヘッダ、スライスヘッダ、タイルグループヘッダ、最大符号化ユニット (L C U) 行または最大符号化ユニット (L C U) 群において信号通知される、案 D 1 ~ D 3 のいずれかに記載の方法。

D 7 . フィールドにおける許容される動き候補の最大数の表示が、単項符号化または短縮単項符号化で 2 値化される、案 D 1 ~ D 6 のいずれかに記載の方法。

D 8 . フィールドにおける許容される動き候補の最大数の表示を明確に信号通知することは、M と許容される動き候補の最大数との間の差を信号通知することを含み、M は整数である、案 D 2 に記載の方法。

D 9 . ジオメトリ分割モードを使用して符号化された 1 つのブロックのための許容される動き候補の最大数は、M から構文解析されたインジケータを引いたものに設定される、案 D 8 に記載の方法。

D 1 0 . M = 5 または M = 6 である、案 D 8 または 9 に記載の方法。

D 1 1 . フィールドにおける許容される動き候補の最大数の表示を含むビットストリーム表現は、変換のために構文解析される、案 D 1 ~ D 8 のいずれか 1 案に記載の方法。

D 1 2 . ビットストリーム表現は、1 つのビットフラグを含み、この 1 つのビットフラグは、ジオメトリ分割モードを使用して符号化された映像のブロックに対して許可される動き候補の最大数が、許可される正規の動き候補またはサブブロック動き候補の最大数と同

10

20

30

40

50

じであることを示す、案 D 1 に記載の方法。

D 1 3 . 1 つのビットフラグを含むビットストリーム表現は、変換のために構文解析される、案 D 1 2 に記載の方法。

D 1 4 . 現在のブロックは、分割パターンに基づいて複数の分割に分割される、案 D 1 ~ D 1 3 のいずれか 1 案に記載の方法。

D 1 5 . ジオメトリ分割モードは、三角形分割モードを含む案 D 1 ~ D 1 4 のいずれか 1 案に記載の方法。

D 1 6 . 現在のブロックを、複数の分割に分割し、複数の分割のうちの少なくとも 1 つの分割は、非正方形であり、かつ非長方形である、案 D 1 ~ D 1 4 のいずれかに記載の方法。

D 1 7 . 変換は、ビットストリーム表現から現在のブロックを生成する、案 D 1 ~ D 1 6 のいずれか 1 案に記載の方法。

D 1 8 . 変換は、現在のブロックからビットストリーム表現を生成する、案 D 1 ~ D 1 6 のいずれか 1 案に記載の方法。

D 1 9 . 処理装置と、命令を搭載した非一時的メモリとを含む、映像システムにおける装置であって、処理装置が実行する命令は、処理装置に、案 D 1 ~ D 1 8 のいずれか 1 案に記載の方法を実装させることを特徴とする装置。

D 2 0 . 非一時的なコンピュータ可読媒体に記憶されたコンピュータプログラム製品であって、コンピュータプログラム製品は、案 D 1 ~ D 1 8 のいずれか 1 案に記載の方法を実行するためのプログラムコードを含む。

【 0 2 8 6 】

いくつかの実施形態において、次の技術案を実装することができる。

E 1 . 映像処理方法。ジオメトリ分割モードを使用して複数の分割に分割される現在の映像ブロックに対して、複数の重み係数群から重み係数群を選択することであって、重み係数群は、現在のブロックの少なくとも 1 つの幅または高さに基づいて選択される、選択することと、現在のブロックと映像のビットストリーム表現との間の変換の一部として、重み係数群を、複数の分割のうちの少なくとも 2 つの共通の境界に沿ったサンプルに適用することと、を含む。

E 2 . 選択することは、幅と高さとの間の比が閾値よりも大きいことに基づく、案 E 1 に記載の方法。

E 3 . 選択することは、高さとの間の比が閾値よりも大きいことに基づく、案 E 1 に記載の方法。

E 4 . 複数の重み係数群は、現在のブロックの幅または高さに基づいて予め規定される、案 E 1 に記載の方法。

E 5 . 複数の重み係数群から 1 つまたは 2 つの重み係数群を選択する、案 E 4 に記載の方法。

E 6 . ジオメトリ分割モードを使用して、複数の分割に分割された現在の映像ブロック内のサンプルについて、複数の分割のうちの少なくとも 2 つの共通の境界の角度に基づいて、少なくとも 1 つの重み係数を判定することと、少なくとも 1 つの重み係数に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む映像処理方法。

E 7 . 判定することは、サンプルの位置にさらに基づいている、案 E 6 に記載の方法。

E 8 . 少なくとも 1 つの重み係数に基づいて、現在のブロックにおけるサンプルの最終予測値を 2 つの予測値の重み付け和として導出する、案 E 1 ~ 7 のいずれか 1 案に記載の方法。

E 9 . 現在のブロックの最終予測ブロックが、 $(f(x, y) \times P1(x, y) + (2M - f(x, y)) \times P2(x, y) + offset) > M$ であり、 $P1(x, y)$ および $P2(x, y)$ は、それぞれ、座標 (x, y) を有するサンプルの第 1 の予測値および第 2 の予測値であり、 M およびオフセットは整数であり、 $f(x, y)$ は、第 1 の予測値に適用された重みを戻す、案 E 8 に記載の方法。

E 1 0 . $offset = (1 < (M - 1))$ である、案 E 9 に記載の方法。

10

20

30

40

50

E 1 1 . 現在のブロックは、分割パターンに基づいて複数の分割に分割される、案 E 1 ~ E 1 0 のいずれか 1 案に記載の方法。

E 1 2 . ジオメトリ分割モードは、三角形分割モードを含む、案 E 1 ~ E 1 1 のいずれか 1 案に記載の方法。

E 1 3 . ジオメトリ分割モードは、1つのブロックを2つの区画に分割することを含む、案 E 1 2 に記載の方法。

E 1 2 . 現在のブロックが、複数の区画に分割され、複数の区画のうちの少なくとも1つの区画は、非正方形であり、かつ非長方形である、案 E 1 ~ E 1 0 のいずれかに記載の方法。

E 1 3 . 変換は、ビットストリーム表現から現在のブロックを生成する、案 E 1 ~ E 1 2 のいずれか 1 案に記載の方法。

E 1 4 . 変換は、現在のブロックからビットストリーム表現を生成する、案 E 1 ~ E 1 2 のいずれか 1 案に記載の方法。

E 1 5 . 処理装置と、命令を搭載した非一時的メモリとを含む、映像システムにおける装置であって、処理装置が実行する命令は、処理装置に、案 E 1 ~ E 1 4 のいずれか 1 案に記載の方法を実装させることを特徴とする装置。

E 1 6 . 非一時的なコンピュータ可読媒体に記憶されたコンピュータプログラム製品であって、コンピュータプログラム製品は、案 E 1 ~ E 1 4 のいずれか 1 案に記載の方法を実行するためのプログラムコードを含む、コンピュータプログラム製品。

【0287】

いくつかの実施形態において、次の技術案を実装することができる。

F 1 . ジオメトリ分割モードを使用して、複数の分割に分割された現在の映像ブロックに対して、複数の分割のうち少なくとも2つの共通の境界に沿った、サンプルサイズが 4×4 のサンプルサイズとは異なるサンプルに対する動き補償処理を行うことと、動き補償処理に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む映像処理方法。

F 2 . 変換に必要なメモリ帯域幅を低減するために、 8×8 個のサンプルサイズで動き補償処理を行う、案 F 1 に記載の方法。

F 3 . 動き補償処理は、 8×4 個のサンプルサイズまたは 4×8 個のサンプルサイズで行われ、変換に必要なメモリ帯域幅を低減することを特徴とする案 F 1 に記載の方法。

F 4 . サンプルサイズは、現在のブロックの高さまたは幅に基づく、案 F 3 に記載の方法。

F 5 . サンプルサイズが、現在のブロックの複数の区画のための分割パターンに基づく、案 F 3 に記載の方法。

F 6 . 分割パターンは、それぞれ 45° の分割パターンまたは 135° の分割パターンで表される、右上隅から左下隅へ、または左上隅から右下隅への分割を含む、案 F 5 に記載の方法。

F 7 . ジオメトリ分割モードを使用して、現在の映像ブロックを複数の分割に分割することと、複数の分割に関連付けられた動き情報から導出された、現在のブロックにおける $K \times L$ 領域のための1つの動き情報のセットを記憶することと、複数の分割のうちの少なくとも1つに関連付けられた少なくとも1つの動き情報のセットを使用して、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む、映像処理方法。

F 8 . $MVInfo_i$ が i 番目の分割の動き情報を表し、 $i = 1$ および $i = 2$ であり、 $MVInfo_1$ が第1の参照ピクチャリスト LX および $MVInfo_2$ からのものであると判定されると、1つの動き情報のセットは、 $MVInfo_1$ と $MVInfo_2$ との組み合わせに基づいて、現在のブロックの双予測および関連する動き情報を含み、 $X = 0$ または $X = 1$ であり、 $Y = (1 - X)$ である、案 F 7 に記載の方法。

F 9 . $MVInfo_i$ が、 i 番目の分割の動き情報を表し、 $i = 1$ および $i = 2$ であり、 $MVInfo_1$ および $MVInfo_2$ の両方が参照リスト LX からのものであると判定された場合、 $MVInfo_j$ に基づいて、現在のブロックのための1つの動き情報単一予測および関連する動き情報のセットであり、 $j = 1$ または $j = 2$ であり、 $X = 0$ または $X =$

10

20

30

40

50

1である、案F 7に記載の方法。

F 1 0 . 動きベクトルとM V I n f o 1およびM V I n f o 2の参照ピクチャインデックスとを組み合わせることによって、双予測を含む1つの動き情報のセットを導出する、案F 8に記載の方法。

F 1 1 . 単一予測を含む1つの動き情報のセットは、M V I n f o 1に基づく、案F 9に記載の方法。

F 1 2 . 単一予測を含む1つの動き情報のセットは、M V I n f o 2に基づく、案F 9に記載の方法。

F 1 3 . M V I n f o 1は、右上隅のサンプルを覆う分割に関連する動き情報のセットを含み、M V I n f o 2は、右上隅のサンプルと左下隅のサンプルとが2つの異なる分割にあると判定された場合、左下隅のサンプルを覆う分割に関連付けられた動き情報のセットを含む、案F 1 ~ F 1 2のいずれか1案に記載の方法。

F 1 4 . 現在のブロックの分割方向が左上隅から右下隅である、案F 1 3に記載の方法。

F 1 5 . M V I n f o 1は、左上隅のサンプルを覆う分割に関連する動き情報のセットを含み、M V I n f o 2は、左上隅のサンプルと右下隅のサンプルとが2つの異なる分割にあると判定された場合、右下隅のサンプルを覆う分割に関連付けられた動き情報のセットを含む、案F 1 ~ F 1 2のいずれか1案に記載の方法。

F 1 6 . 現在のブロックの分割方向が右上隅から左下隅である、案F 1 5に記載の方法。

F 1 7 . 記憶された1つの動き情報のセットが、映像の後続のブロックの時間的動き予測、空間的動き予測、または現在のブロックのフィルタリング処理のうちの1つに使用される、案F 7に記載の方法。

F 1 8 . 記憶された1つの動き情報のセットが、異なるピクチャにおける他のブロックの処理に用いられる、案F 7に記載の方法。

F 1 9 . フィルタリング処理は、デブロックングまたは適応ループフィルタリング (A L F) を含む、案F 1 7に記載の方法。

F 2 0 . M V I n f o iおよびM V I n f o jが、それぞれi番目の分割およびj番目の分割の動き情報を表し、M V I n f o iおよびM V I n f o jの両方が、参照ピクチャリストL Xからのものであり、 $X = 0$ または $X = 1$ であり、1つの動き情報のセットは、M V I n f o iと、M V I n f o jを参照ピクチャリストL Yにスケーリングすることによって生成されるスケーリングされた動きベクトルとを組み合わせることに基づいており、 $Y = (1 - X)$ である、案7に記載の方法。

F 2 1 . Xの値が予め判定されるか、またはビットストリーム表現で信号通知される、案F 2 0に記載の方法。

F 2 2 . M V I n f o iおよびM V I n f o jが、それぞれi番目の分割およびj番目の分割の動き情報を表し、M V I n f o iおよびM V I n f o jの両方が、参照ピクチャリストL Xからのものであり、 $X = 0$ または $X = 1$ であり、1つの動き情報のセットは、M V I n f o iと、M V I n f o jを参照ピクチャリストL Yにマッピングすることによって生成されるマッピングされた動きベクトルとを組み合わせることに基づいており、 $Y = (1 - X)$ である、案F 7に記載の方法。

F 2 3 . 1つの動き情報のセットは、双予測動きベクトルを含む、案F 2 2に記載の方法。

F 2 4 . M V I n f o jをリストL Yにおける参照ピクチャにマッピングすることは、スケーリング演算を除外する、案F 2 2に記載の方法。

F 2 5 . 1つの動き情報のセットを記憶することは、複数の分割に関連付けられた復号化された動き情報に基づく、案F 7に記載の方法。

F 2 6 . M V I n f o iおよびM V I n f o jが、それぞれi番目の分割およびj番目の分割の動き情報を表し、M V I n f o iおよびM V I n f o jは、いずれも同じ参照ピクチャリストL Xからのものであり、 $X = 0$ または $X = 1$ であり、1つの動き情報のセットが単一予測の動き情報を含む、案F 2 5に記載の方法。

F 2 7 . M V I n f o iおよびM V I n f o jが、i番目の分割およびj番目の分割の動き情報をそれぞれ表し、M V I n f o iが、参照ピクチャリストL XおよびM V I n f o

10

20

30

40

50

j からのものであり、 $X = 0$ または $X = 1$ であり、 $Y = (1 - X)$ であり、1つの動き情報のセットが双予測動き情報を含む、案 F 25 に記載の方法。

F 28. $MVInfo_i$ および $MVInfo_j$ が、それぞれ i 番目および j 番目の分割の動き情報を表し、 $MVInfo_i$ および $MVInfo_j$ の両方が参照ピクチャリスト L_X からのものであり、 $X = 0$ または $X = 1$ であり、参照ピクチャリスト L_Y が $MVInfo_i$ または $MVInfo_j$ を含み、 $Y = (1 - X)$ であり、1つの動き情報のセットが双予測動き情報を含む、案 F 25 に記載の方法。

F 29. 1つの動き情報のセットを記憶することは、複数の分割のうちの少なくとも2つの共通の境界に沿って位置する現在のブロックのサブ領域に基づく、案 F 7 に記載の方法。

F 30. 異なるサブ領域が異なる動き情報を記憶する、案 F 29 に記載の方法。

F 31. サブ領域は、単一予測または双予測動き情報を記憶する、案 F 29 に記載の方法。

F 32. 単一の動き情報のセットを記憶することは、現在のブロックにおける1つのサブブロックの位置に基づいており、このサブブロックは、単一予測動き情報のみを記憶する、案 F 7 に記載の方法。

F 33. 現在のブロックに適用される動き補償処理において1つの動き情報のセットを使用する、案 F 1 ~ F 32 のいずれかに記載の方法。

F 34. $MVInfo_i$ および $MVInfo_j$ が、それぞれ i 番目および j 番目の分割の動き情報を表し、 $MVInfo_i$ および $MVInfo_j$ の両方が参照ピクチャリスト L_X からのものであり、 $X = 0$ または $X = 1$ であり、1つの動き情報のセットが、(a) $MVInfo_i$ 、(b) $MVInfo_j$ または (c) $MVInfo_i$ および $MVInfo_j$ を平均することから導出された参照ピクチャリスト L_X のための双予測動きベクトルを含む、案 F 7 に記載の方法。

F 35. $MVInfo_i$ および $MVInfo_j$ が、それぞれ i 番目および j 番目の分割の動き情報を表し、 $MVInfo_i$ および $MVInfo_j$ の両方が参照ピクチャリスト L_X からのものであり、 $X = 0$ または $X = 1$ であり、且つ1つの動き情報のセットが $MVInfo_i$ および $MVInfo_j$ の平均または重み付け平均を含み、且つ1つの動き情報のセットに関連付けられた参照ピクチャが $MVInfo_i$ の参照ピクチャまたは $MVInfo_j$ の参照ピクチャのいずれかを含む、案 F 7 に記載の方法。

F 36. $K \times L$ 領域が、複数の分割のうちの少なくとも2つの分割間の共通の境界に少なくとも1つのサンプルを含む、案 F 1 ~ F 35 のいずれかに記載の方法。

F 37. ジオメトリ分割モードは三角形分割モードを含む、案 F 1 ~ F 36 のいずれか1案に記載の方法。

F 38. 複数の分割の少なくとも1つの分割が、非正方形であり、かつ非長方形である、案 F 1 ~ F 36 のいずれか1案に記載の方法。

F 39. 変換は、ビットストリーム表現から現在のブロックを生成する、案 F 1 ~ F 38 のいずれか1案に記載の方法。

F 40. 変換は、現在のブロックからビットストリーム表現を生成する、案 F 1 ~ F 38 のいずれか1案に記載の方法。

F 41. 処理装置と、命令を搭載した非一時的メモリとを含む、映像システムにおける装置であって、処理装置が実行する命令は、処理装置に、案 F 1 ~ F 40 のいずれか1案に記載の方法を実装させることを特徴とする装置。

F 42. 非一時的なコンピュータ可読媒体に記憶されたコンピュータプログラム製品であって、コンピュータプログラム製品は、案 F 1 ~ F 40 のいずれか1案に記載の方法を実行するためのプログラムコードを含む、コンピュータプログラム製品。

【0288】

いくつかの実施形態において、次の技術案を実装することができる。

G 1. 現在の映像ブロックを複数の分割に分割することと、ジオメトリ分割モードを使用して符号化された現在のブロックと映像のビットストリーム表現との間で変換を行うこととを含み、現在のブロックの $M \times N$ 個のブロック単位ごとに1つの動き情報のセットが記憶され、この1つの動き情報のセットは、複数の分割の各々に関連付けられた動き情報が

10

20

30

40

50

ら導出されたものであり、MまたはNが4に等しくなく、 $M \times N$ 個のブロック単位における各サンプルが同じ動き情報を共有する、映像処理方法。

G 2. M = 8であり、N = 8である、案G 1に記載の方法。

G 3. $M \times N$ ブロックユニットの第1の部分は、複数の区画のうちの少なくとも2つの共通の境界上にある第1のサンプルセットを備え、 $M \times N$ ブロックユニットの第2の部分は、複数の分割のうちの少なくとも2つの分割のうちの1つの内部にある第2のサンプルセットを含む、案G 1に記載の方法。

G 4. $MVInfo_i$ がi番目の分割の動き情報を表し、 $i = 1$ および $i = 2$ であり、 $MVInfo_1$ が第1の参照ピクチャリストLXからのものであり、 $MVInfo_2$ がLYからのものであると判定されると、同じ動き情報が、複数の分割の $MVInfo_i$ に基づいて、現在のブロックの双予測および関連する動き情報を含み、 $X = 0$ または $X = 1$ であり、 $Y = (1 - X)$ である、案G 3に記載の方法。

G 5. $MVInfo_i$ が、i番目の分割の動き情報を表し、 $i = 1$ および $i = 2$ であり、 $MVInfo_1$ および $MVInfo_2$ の両方が参照ピクチャリストLXからのものであると判定されると、同じ動き情報が、 $MVInfo_j$ に基づいて、現在のブロックのための単一予測および関連する動き情報を含み、 $j = 1$ または $j = 2$ であり、 $X = 0$ または $X = 1$ である、案G 3に記載の方法。

G 6. $MVInfo_i$ がi番目の分割の動き情報を表し、 $i = 1$ 、 $i = 2$ であり、同じ動き情報が、 $MVInfo_j$ に基づいて、現在のブロックのための単一予測および関連する動き情報を含み、 $j = 1$ または $j = 2$ である、案G 3に記載の方法。

G 7. 現在の映像ブロックを複数の分割に分割することと、現在のブロックと映像のビットストリーム表現との間の変換の一部として、記憶されるべき第1のサブ領域に関連付けられた第2の動き情報とは異なる第1の動き情報に基づいて、現在のブロックの第1のサブ領域に対して動き補償処理を行うこととを含む、映像処理方法。

G 8. 第1のサブ領域は、複数の分割のうちの少なくとも2つの共通の境界上に $M \times N$ 個のサンプルセットを含み、動き補償処理は、双予測を含み、かつ双予測のList 0またはList 1からの1つの動き情報のセットのみが記憶される、案G 7に記載の方法。

G 9. 1つの動き情報のセットは、マージモードまたは高度動きベクトル予測 (AMVP) モードを使用して符号化される近傍のブロックの空間的動き候補を導出するために使用される、案G 8に記載の方法。

G 10. 1つの動き情報のセットは、現在のブロックを含む現在のピクチャと異なるピクチャにおける後続のブロックの時間的動き候補を導出するために用いられる、案G 8に記載の方法。

G 11. 1つの動き情報のセットは、現在のブロックのインループ処理に使用される、案G 8に記載の方法。

G 12. インループ処理は、デブロッキングまたは適応ループフィルタリングを含む、案G 11に記載の方法。

G 13. 第2のサブ領域に関連付けられた第3の動き情報に基づいて、現在のブロックの第2のサブ領域に対する動き補償処理、および、第1のサブ領域と第2のサブ領域とが、複数の分割を含む同じ符号化ユニット (CU) 内にある、案G 7に記載の方法。

G 14. ジオメトリ分割モードを使用して、現在の映像ブロックを複数の分割に分割することと、複数の分割に関連付けられた分割方向または復号化されたマージインデックスまたはマージ候補リストに基づいて、第1のサンプルセットおよび第2のサンプルセットの動き情報を記憶することとあって、第1のサンプルのセットは、複数の分割の少なくとも2つの共通の境界上にあり、第2のサンプルのセットは、複数の分割の少なくとも2つのうちの1つの内部にある、記憶することと、記憶された動き情報に基づいて、現在のブロックと映像のビットストリーム表現との間で変換、または映像の後続のブロックとビットストリーム表現との間で変換を行うこととを含む、映像処理方法。

G 15. 記憶された動き情報は、選択された復号化マージインデックスに関連付けられた複数の分割のうちの1つの分割に関連付けられた動き情報に対応する、案G 14に記載の

10

20

30

40

50

方法。

G 1 6 . 記憶された動き情報は、選択されたマージインデックスセットが変数 k である動き候補リストにおける動き候補に対応する、案 G 1 4 に記載の方法。

G 1 7 . $k = 0$ である、案 G 1 6 に記載の方法。

G 1 8 . 記憶された動き情報は、1つのマージ候補リストにおける k 番目のマージ候補の動き情報に対応する、案 G 1 6 に記載の方法。

G 1 9 . 選択された復号化マージインデックスは、分割方向に基づく、案 G 1 5 に記載の方法。

G 2 0 . 選択された復号化マージインデックスに等しいインデックスを有するマージ候補が双予測に基づくものであると判定されると、記憶された動き情報が双予測動き情報を含む、案 G 1 5 に記載の方法。

10

G 2 1 . 選択されたデコードされたマージインデックスに等しいインデックスを有するマージ候補が単一予測に基づくものであると判定されると、記憶された動き情報が単一予測の動き情報を含む、案 G 1 5 に記載の方法。

G 2 2 . 第1のサブ領域は、複数の分割のうちの少なくとも2つの共通の境界上に $M \times N$ 個のサンプルを含み、動き補償処理は、双予測を含み、かつ双予測の $L i s t 0$ または $L i s t 1$ からの1つの動き情報のセットのみを記憶する、案 G 6 に記載の方法。

G 2 3 . 第1のサブ領域は、複数の分割のうちの少なくとも2つの共通の境界上に $M \times N$ 個のサンプルを含み、動き補償処理は、双予測を含み、 $L i s t X$ からの単一予測情報のみが記憶され、 $X = 0$ および $X = 1$ である、案 G 6 に記載の方法。

20

G 2 4 . 第1のサブ領域は、複数の分割のうちの少なくとも2つの共通の境界上に $M \times N$ 個のサンプルを含み、動き補償処理は、単一予測を含み、第2の動き情報は、記憶される $L i s t X$ からの単一予測情報を含み、 $X = 0$ および $X = 1$ である、案 G 6 に記載の方法。

G 2 5 . 現在の映像ブロックを複数の分割に分割することと、複数の分割のうちの少なくとも2つの分割の動き情報に基づく仮想双予測動き情報を記憶することと、仮想双予測動き情報に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む映像処理方法。

G 2 6 . $M v I n f o 1$ および $M v I n f o 2$ が、それぞれ第1の分割および第2の分割の動き情報であり、仮想双予測動き情報は、 $M v I n f o A$ の予測方向を $L i s t X$ から $L i s t Y$ に修正することに基づくものであり、 $A = 1$ または $A = 2$ であり、 $X = 0$ または $X = 1$ であり、 $Y = (1 - X)$ である、案 G 2 5 に記載の方法。

30

G 2 7 . $M v I n f o A$ の動きベクトルおよび参照インデックスが不変に維持される、案 G 2 6 に記載の方法。

G 2 8 . $M v I n f o A$ の参照インデックスは変更されず、 $M v I n f o A$ の動きベクトルは反対の値に設定される、案 G 2 6 に記載の方法。

G 2 9 . $M v I n f o 1$ および $M v I n f o 2$ が同じ予測方向からのものである、案 G 2 6 ~ G 2 8 のいずれか1つに記載の方法。

G 3 0 . 現在の映像ブロックを複数の分割に分割することと、低遅延チェックフラグに基づいて、複数の分割の第1の分割 ($M v I n f o 1$) の動き情報および複数の分割の第2の分割 ($M v I n f o 2$) の動き情報を記憶することであって、低遅延チェックフラグは、現在のブロックを構成する現在のピクチャの $P O C$ 値以下のピクチャオーダカウント ($P O C$) 値を有するすべての参照ピクチャを示す、記憶することと、記憶した動き情報に基づいて、現在のブロックと映像のビットストリーム表現との間で変換を行うことと、を含む、映像処理方法。

40

G 3 1 . $M v I n f o 1$ と $M v I n f o 2$ が異なる予測方向からのものであり、動き情報を記憶することは、 $M v I n f o 1$ と $M v I n f o 2$ を組み合わせて生成された双予測動き情報を記憶することを含む、案 G 3 0 に記載の方法。

G 3 2 . $M v I n f o 1$ および $M v I n f o 2$ が同じ予測方向 ($L i s t X$) からのものであり、 $X = 0$ または $X = 1$ であり、 $M v I n f o 2$ の双予測リストを $L i s t X$ から $L i s t Y$ に設定することで、 $M v I n f o 2$ を修正することをさらに含み、 $Y = (1 - X$

50

)であり、動き情報を記憶することは、M v I n f o 1と修正されたM v I n f o 2とを結合することによって生成される双予測動作を記憶することを含む、案G 3 0に記載の方法。

G 3 3. 動き情報を記憶することは、現在のブロックのサブブロックの動き情報を記憶することを含む、案G 2 5～G 3 2のいずれか1案に記載の方法。

G 3 4. サブブロックは、複数の分割のうちの少なくとも2つの共通の境界上のサンプルを含む、案G 3 3に記載の方法。

G 3 5. サブブロックは、現在のブロックにおける対角線または反対角線を含む、案G 3 3に記載の方法。

G 3 6. サブブロックは、現在のブロックの右下隅にある、案G 3 3に記載の方法。

G 3 7. サブブロックは、現在のブロックの右列または下行にある、案G 3 3に記載の方法。

G 3 8. 複数の分割の少なくとも1つの分割が、非正方形であり、かつ非長方形である、案G 1～G 3 7のいずれか1案に記載の方法。

G 3 9. 変換は、ビットストリーム表現から現在のブロックを生成する、案G 1～G 3 8のいずれか1案に記載の方法。

G 4 0. 変換は、現在のブロックからビットストリーム表現を生成する、案G 1～G 3 8のいずれか1案に記載の方法。

G 4 1. 処理装置と、命令を搭載した非一時的メモリとを含む、映像システムにおける装置であって、処理装置が実行する命令は、処理装置に、案G 1～G 4 0のいずれか1案に記載の方法を実装させることを特徴とする装置。

G 4 2. 非一時的なコンピュータ可読媒体に記憶されたコンピュータプログラム製品であって、コンピュータプログラム製品は、案G 1～G 4 0のいずれか1案に記載の方法を実行するためのプログラムコードを含む、コンピュータプログラム製品。

【0289】

図49は、映像処理装置4900のブロック図である。装置4900は、本明細書で説明される方法の1つ以上を実装するために使用してもよい。装置4900は、スマートフォン、タブレット、コンピュータ、モノのインターネット(IoT(Internet of Things))受信機等を実施されてもよい。装置4900は、1つ以上の処理装置4902と、1つ以上のメモリ4904と、映像処理ハードウェア4906と、を含んでもよい。1つ以上の処理装置4902は、本明細書に記載される1つ以上の方法を実装するように構成されてもよい。メモリ(複数可)4904は、本明細書で説明される方法および技術を実装するために使用されるデータおよびコードを記憶するために使用されてもよい。映像処理ハードウェア4906は、本明細書に記載される技術をハードウェア回路にて実装するために使用してもよい。

【0290】

いくつかの実施形態において、映像符号化方法は、図49を参照して説明したように、ハードウェアプラットフォームに実装される装置を使用して実施されてもよい。

【0291】

開示される技術の一部の実施形態は、映像処理ツールまたはモードを有効にする決定または判定を行うことを含む。一例において、映像処理ツールまたはモードが有効される場合、エンコーダは、1つの映像ブロックを処理する際に、このツールまたはモードを使用するまたは実装するが、このツールまたはモードの使用に基づいて、結果として得られるビットストリームを必ずしも修正しなくてもよい。すなわち、映像のブロックから映像のビットストリーム表現への変換は、決定または判定に基づいて映像処理ツールまたはモードが有効にされる場合に、この映像処理ツールまたはモードを使用する。別の例において、映像処理ツールまたはモードが有効にされる場合、デコーダは、ビットストリームが映像処理ツールまたはモードに基づいて修正されたことを認識して、ビットストリームを処理する。すなわち、決定または判定に基づいて有効にされた映像処理ツールまたはモードを使用して、映像のビットストリーム表現から映像のブロックへの変換を行う。

【0292】

開示される技術の一部の実施形態は、映像処理ツールまたはモードを無効にする決定または判定を行うことを含む。一例において、映像処理ツールまたはモードが無効にされている場合、エンコーダは、映像のブロックを映像のビットストリーム表現に変換する際に、このツールまたはモードを使用しない。別の例において、映像処理ツールまたはモードが無効にされる場合、デコーダは、ビットストリームが、決定または判定に基づいて有効にされた映像処理ツールまたはモードを使用して修正されていないことを認識して、ビットストリームを処理する。

【0293】

図50は、本明細書で開示される様々な技術が実装され得る例示的な映像処理システム5000を示すブロック図である。様々な実装形態は、システム5000のモジュールの一部または全部を含んでもよい。システム5000は、映像コンテンツを受信するための入力ユニット5002を含んでもよい。映像コンテンツは、未加工または非圧縮フォーマット、例えば、8または10ビットのマルチモジュール画素値で受信されてもよく、または圧縮または符号化フォーマットで受信されてもよい。入力ユニット5002は、ネットワークインターフェース、周辺バスインターフェース、または記憶インターフェースを表してもよい。ネットワークインターフェースの例は、イーサネット（登録商標）、パッシブ光ネットワーク（PON）等の有線インターフェース、およびWi-Fiまたはセルラーインターフェース等の無線インターフェースを含む。

【0294】

システム5000は、本明細書に記載される様々な符号化または符号化方法を実装することができる符号化モジュール5004を含んでもよい。符号化モジュール5004は、入力ユニット5002からの映像の平均ビットレートを符号化モジュール5004の出力に低減し、映像の符号化表現を生成してもよい。従って、この符号化技術は、映像圧縮または映像コード変換技術と呼ばれることがある。符号化モジュール5004の出力は、モジュール5006によって表されるように、記憶されてもよいし、接続された通信を介して送信されてもよい。入力ユニット5002において受信された、記憶されたまたは通信された映像のビットストリーム（または符号化）表現は、モジュール5008によって使用されて、表示インターフェースユニット5010に送信される画素値または表示可能な映像を生成してもよい。ビットストリーム表現からユーザが見ることができる映像を生成する処理は、映像伸張（映像展開）と呼ばれることがある。さらに、特定の映像処理動作を「符号化」動作またはツールと呼ぶが、符号化ツールまたは動作は、エンコーダおよびそれに対応するデコーダツールで使用され、復号化の結果を逆にする復号化ツールまたは動作が、デコーダによって行われることが理解されよう。

【0295】

周辺バスインターフェースユニットまたは表示インターフェースユニットの例は、ユニバーサルシリアルバス（USB）または高精細マルチメディアインターフェース（HDMI（登録商標））またはディスプレイポート等を含んでもよい。ストレージインターフェースの例は、シリアルアドバンスドテクノロジーアタッチメント（SATA）、PCI、IDEインターフェース等を含む。本明細書に記載される技術は、携帯電話、ノートパソコン、スマートフォン、またはデジタルデータ処理および／または映像表示を実施可能な他のデバイス等の様々な電子デバイスに実施されてもよい。

【0296】

本明細書における説明に関して、現在の映像ブロックのビットストリーム表現は、必ずしも連続したビットの形式である必要はなく、ビットストリーム構文の異なる位置に存在するビット（例えば、ヘッダフィールドおよびパラメータセット）に基づいてもよい。

【0297】

さらに、いくつかの開示された実施形態において、ATMVPおよびアフィンマージを含むサブブロック符号化モードの場合、1つのマージリスト構築処理が共有される。なお、ATMVPおよびアフィンマージ候補を順に追加してもよい。

【0298】

いくつかの実施形態において、三角形予測モードにおいて、2つの分割に対する1つのマージリスト構築処理は、2つの分割が独自のマージ候補インデックスを選択しても共有される。このマージリストを構築するとき、空間的に近傍のブロックとこのブロックの2つの時間的ブロックとをチェックする。空間的近傍および時間的ブロックから導出された動き情報を、本発明者らのIDFにおいて正規の動き候補と呼ぶ。これらの正規の動き候補をさらに利用して、複数のTPM候補を導出する。この変換は、ブロックレベル全体で行われてもよく、2つの分割は、それら自身の予測ブロックを生成するために異なる動きベクトルを使用してもよい。

【0299】

残りの符号化ブロックについては、1つのマージリスト構築処理を共有する。空間的／時間的／HMVP、ペアワイズ合成双予測マージ候補、および動きゼロ候補を順に挿入してもよい。

【0300】

映像符号化または復号化方法（圧縮／展開）が開示されていることが、当業者には理解されるであろう。映像ブロックの幾何学的（または幾何学的）分割は、動き情報を正確に追跡し、符号化された映像の圧縮性能を改善するために使用してもよい。さらに、開示された方法を使用することによって、実施形態は、今日の映像符号化／復号化技術に比べて、さらなる符号化利得を実現し、かつ／または実装形態の複雑性を低減することができることが理解される。

【0301】

本明細書に開示されたそして他の解決案、例、実施形態、モジュール、および機能操作は、本明細書に開示された構造およびその構造的等価物を含め、デジタル電子回路、またはコンピュータソフトウェア、ファームウェア、若しくはハードウェアで実装されてもよく、またはそれらの1つ以上の組み合わせでしてもよい。開示された、そして他の実施形態は、1つ以上のコンピュータプログラム製品、すなわち、データ処理装置によって実装されるため、またはデータ処理装置の操作を制御するために、コンピュータ可読媒体上に符号化されたコンピュータプログラム命令の1つ以上のモジュールとして実施することができる。このコンピュータ可読媒体は、機械可読記憶装置、機械可読記憶基板、メモリデバイス、機械可読伝播信号をもたらす物質の組成物、またはこれらの1つ以上の組み合わせであってもよい。「データ処理装置」という用語は、例えば、プログラマブル処理装置、コンピュータ、または複数の処理装置若しくはコンピュータを含め、データを処理するためのすべての装置、デバイス、および機械を含む。この装置は、ハードウェアの他に、当該コンピュータプログラムの実行環境を作るコード、例えば、処理装置ファームウェア、プロトコルスタック、データベース管理システム、オペレーティングシステム、またはこれらの1つ以上の組み合わせを構成するコードを含んでもよい。伝播信号は、人工的に生成した信号、例えば、機械で生成した電気、光、または電磁信号であり、適切な受信装置に送信するための情報を符号化するように生成される。

【0302】

コンピュータプログラム（プログラム、ソフトウェア、ソフトウェアアプリケーション、スクリプト、またはコードとも呼ばれる）は、コンパイルされた言語または解釈された言語を含む任意の形式のプログラミング言語で記述することができ、それは、スタンドアロンプログラムとして、またはコンピューティング環境で使用するのに適したモジュール、コンポーネント、サブルーチン、または他のユニットを含む任意の形式で展開することができる。コンピュータプログラムは、必ずしもファイルシステムにおけるファイルに対応するとは限らない。プログラムは、他のプログラムまたはデータを保持するファイルの一部（例えば、マークアップ言語文書に格納された1つ以上のスクリプト）に記録されていてもよいし、当該プログラム専用の1つのファイルに記憶されていてもよいし、複数の調整ファイル（例えば、1つ以上のモジュール、サブプログラム、またはコードの一部を格納するファイル）に記憶されていてもよい。1つのコンピュータプログラムを、1つの

10

20

30

40

50

サイトに位置する1つのコンピュータ、または複数のサイトに分散され通信ネットワークによって相互接続される複数のコンピュータで実行させるように展開可能である。

【0303】

本明細書に記載された処理およびロジックフローは、入力データ上で動作し、出力を生成することによって機能を行うための1つ以上のコンピュータプログラムを実行する1つ以上のプログラマブル処理装置によって行うことができる。処理およびロジックフローはまた、特定用途のロジック回路、例えば、FPGA（フィールドプログラマブルゲートアレイ）またはASIC（特定用途向け集積回路）によって行うことができ、装置はまた、特別目的のロジック回路として実装することができる。

【0304】

コンピュータプログラムの実行に適した処理装置は、例えば、汎用および専用マイクロ処理装置の両方、並びに任意の種類のデジタルコンピュータの任意の1つ以上の処理装置を含む。一般的に、処理装置は、読み出し専用メモリまたはランダムアクセスメモリまたはその両方から命令およびデータを受信する。コンピュータの本質的な要素は、命令を実行するための処理装置と、命令およびデータを記憶するための1つ以上のメモリデバイスとである。一般的に、コンピュータは、データを記憶するための1つ以上の大容量記憶デバイス、例えば、磁気、光磁気ディスク、または光ディスクを含んでもよく、またはこれらの大容量記憶デバイスからデータを受信するか、またはこれらにデータを転送するように動作可能に結合されてもよい。しかしながら、コンピュータは、このようなデバイスを有する必要はない。コンピュータプログラム命令およびデータを記憶するのに適したコンピュータ可読媒体は、あらゆる形式の不揮発性メモリ、媒体、およびメモリデバイスを含み、例えば、EPROM、EEPROM、フラッシュメモリデバイス、磁気ディスク、例えば内部ハードディスクまたはリムーバブルディスク、光磁気ディスク、およびCD-ROMおよびDVD-ROMディスク等の半導体メモリデバイスを含む。処理装置およびメモリは、特定用途のロジック回路によって補完されてもよく、または特定用途のロジック回路に組み込まれてもよい。

【0305】

本特許明細書は多くの詳細を含むが、これらは、任意の主題または特許請求され得るものの範囲を限定するものと解釈されるべきではなく、特定の技術の特定の実施形態に特有であり得る特徴の説明と解釈されるべきである。本特許明細書において別の実施形態の文脈で説明されている特定の技術的特徴は、1つの例において組み合わせて実装してもよい。逆に、1つの例の文脈で説明された様々な特徴は、複数の実施形態において別個にまたは任意の適切なサブコンビネーションで実装してもよい。さらに、特徴は、特定の組み合わせで作用するものとして上記に記載され、最初にそのような主張されていてもよいが、主張された組み合わせからの1つ以上の特徴は、場合によっては、組み合わせから抜粋されることができ、主張された組み合わせは、サブ組み合わせまたはサブコンビネーションのバリエーションに向けられてもよい。

【0306】

同様に、動作は図面において特定の順番で示されているが、これは、所望の結果を達成するために、このような動作が示された特定の順番でまたは連続した順番で行われること、または示された全ての操作が行われることを必要とするものと理解されるべきではない。また、本特許明細書に記載されている例における様々なシステムモジュールの分離は、全ての実施形態においてこのような分離を必要とするものと理解されるべきではない。

【0307】

いくつかの実装形態および例のみが記載されており、本特許明細書に記載され図示されている内容に基づいて、他の実施形態、拡張および変形が可能である。

10

20

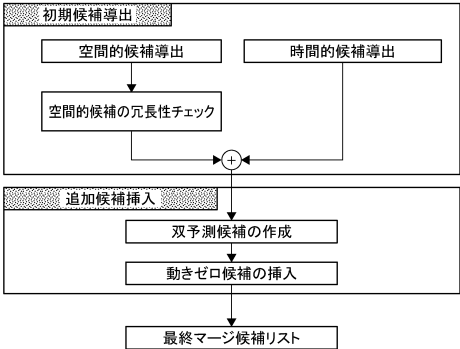
30

40

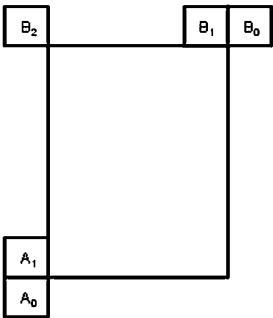
50

【図面】

【図 1】

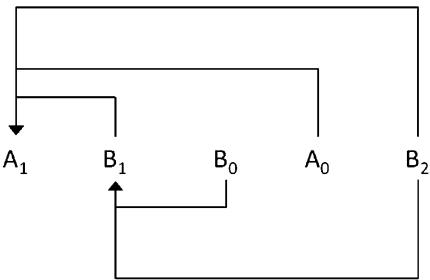


【図 2】

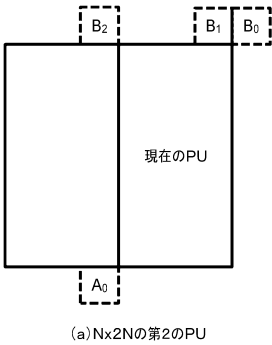


10

【図 3】



【図 4 A】



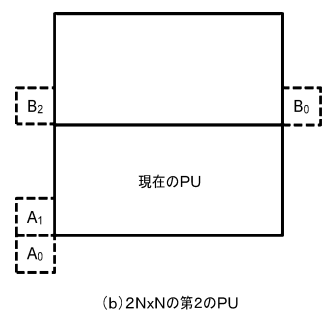
20

30

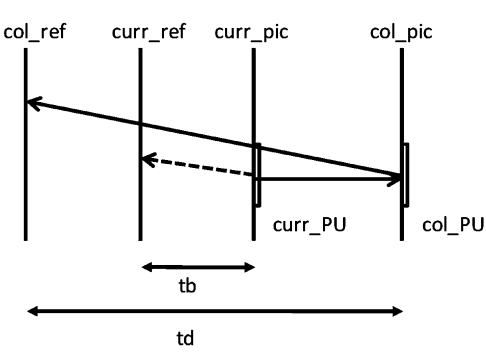
40

50

【図 4 B】

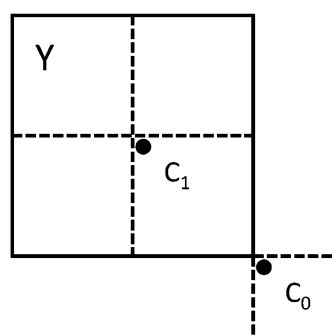


【図 5】



10

【図 6】



【図 7】

オリジナルマージ候補リスト				合併候補追加後のマージ候補リスト			
マージインデックス		L0	L1	マージインデックス		L0	L1
0		$mvL0_A, ref0$		0		$mvL0_A, ref0$	
1		-	$mvL1_B, ref0$	1		$mvL1_B, ref0$	$mvL1_B, ref0$
2				2		$mvL0_A, ref0$	$mvL1_B, ref0$
3				3			
4				4			

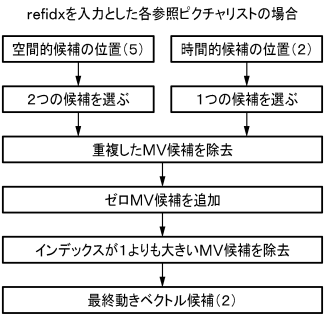
20

30

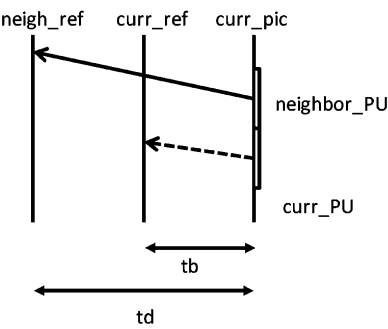
40

50

【図 8】

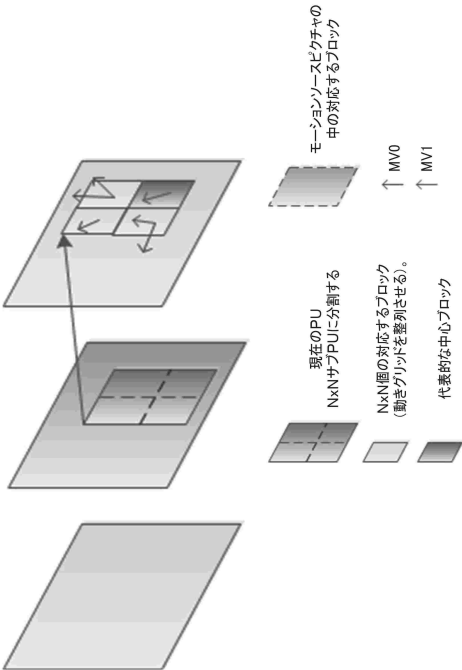


【図 9】

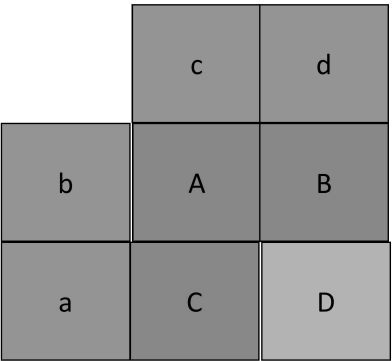


10

【図 10】



【図 11】



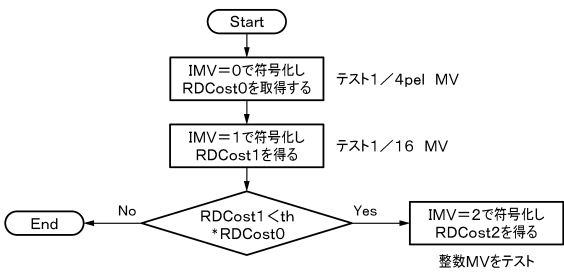
20

30

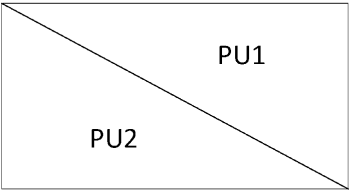
40

50

【図 1 2】

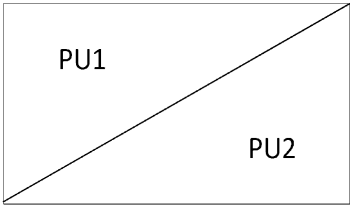


【図 1 3 A】

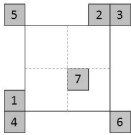


10

【図 1 3 B】

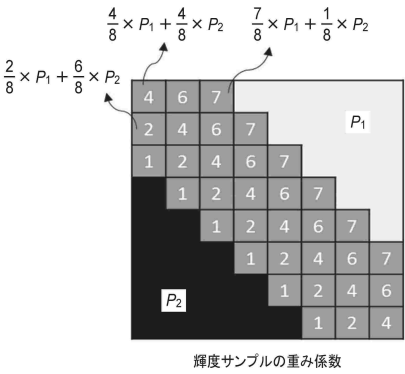


【図 1 4】

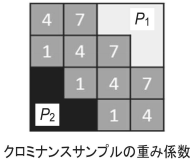


20

【図 1 5 A】



【図 1 5 B】

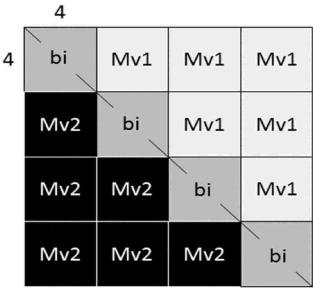


30

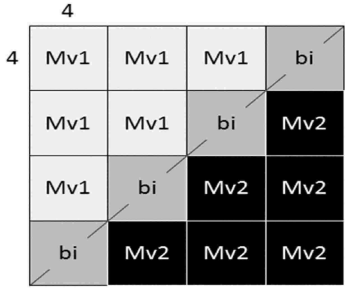
40

50

【図 1 6 A】

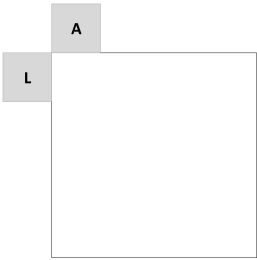


【図 1 6 B】

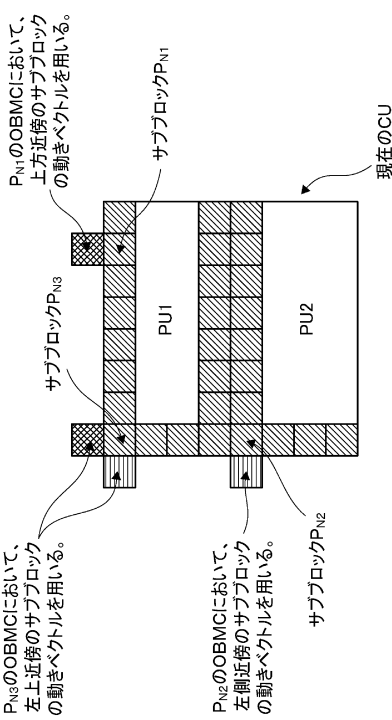


10

【図 1 7】



【図 1 8 A】



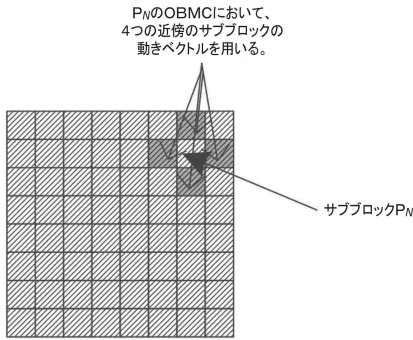
20

30

40

50

【図 1 8 B】

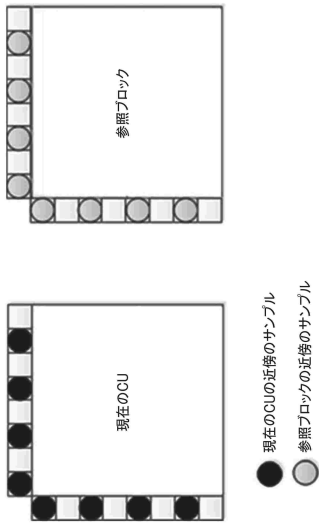


【図 1 8 C】

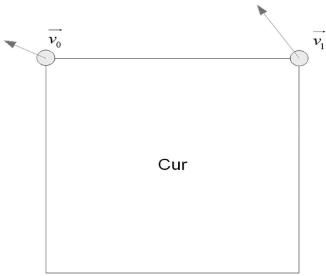
マージインデックス	L0 MV		L1 MV	
	0	1	2	3
0	x			
1		x		
2	x			
3		x		
4	x			

10

【図 1 9】



【図 2 0 A】



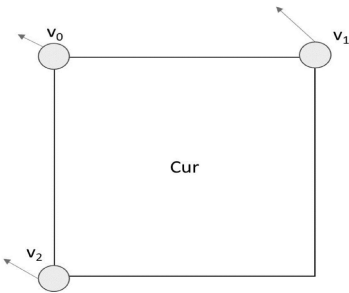
20

30

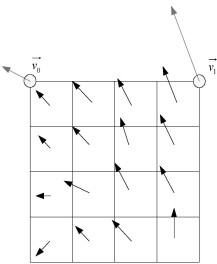
40

50

【図 2 0 B】

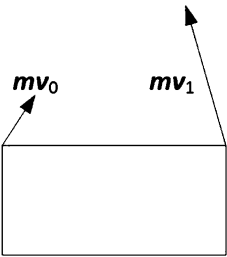


【図 2 1】

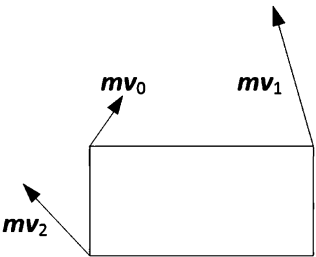


10

【図 2 2 A】

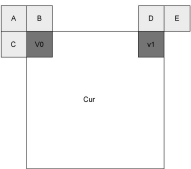


【図 2 2 B】

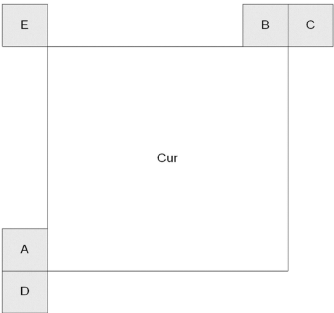


20

【図 2 3】



【図 2 4 A】

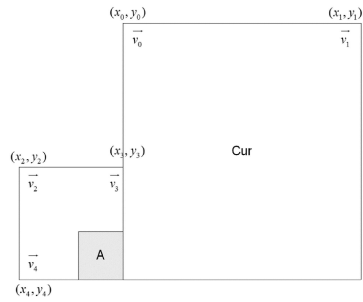


30

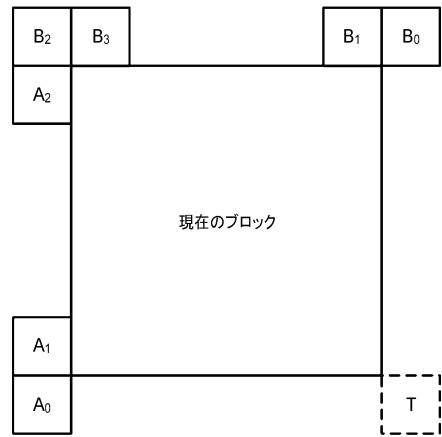
40

50

【図 2 4 B】

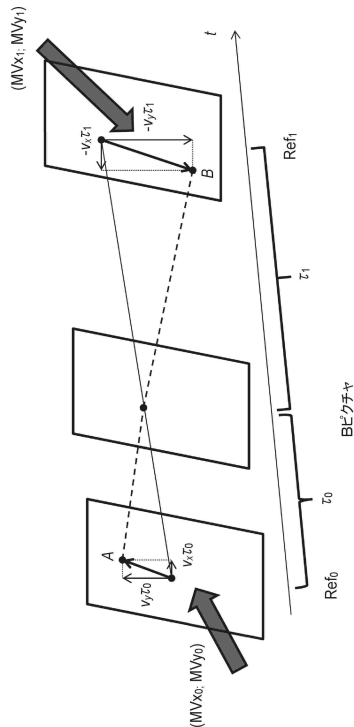


【図 2 5】

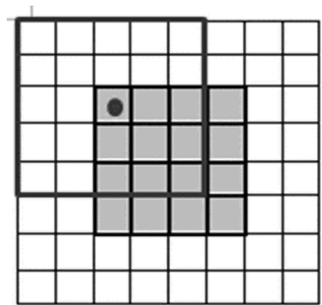


10

【図 2 6】



【図 2 7 A】



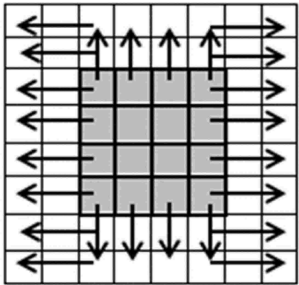
20

30

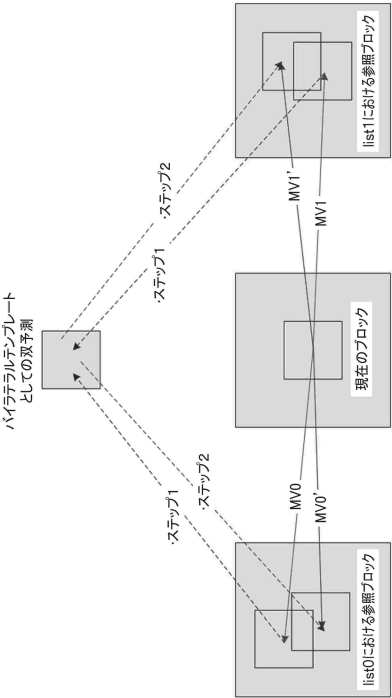
40

50

【図 2 7 B】



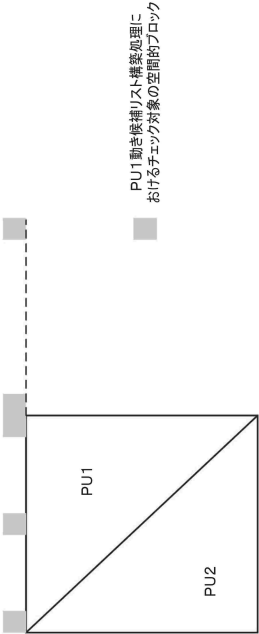
【図 2 8】



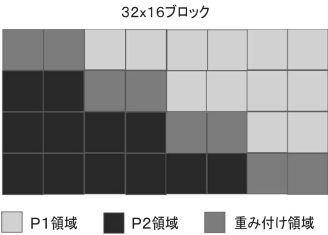
10

20

【図 2 9】



【図 3 0 A】

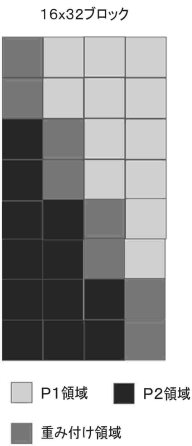


30

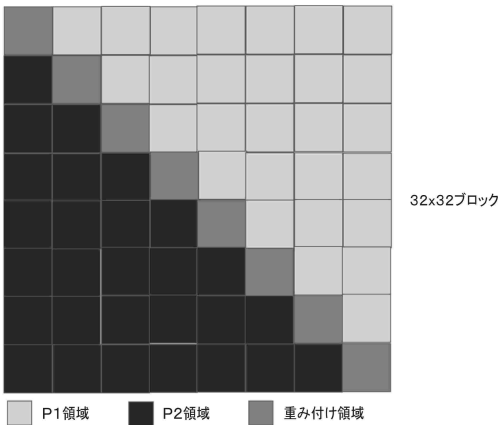
40

50

【図 3 0 B】

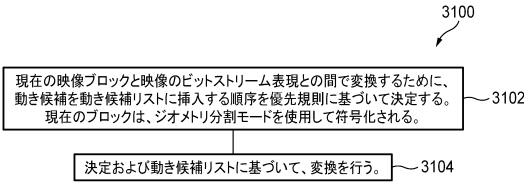


【図 3 0 C】

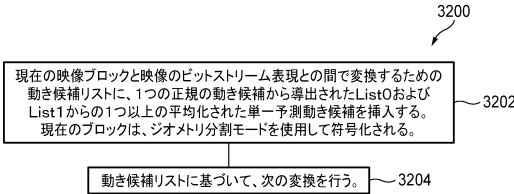


10

【図 3 1】

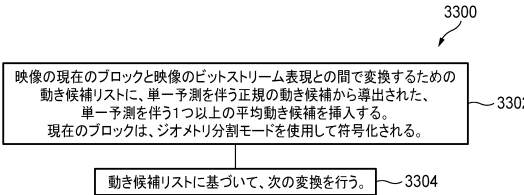


【図 3 2】

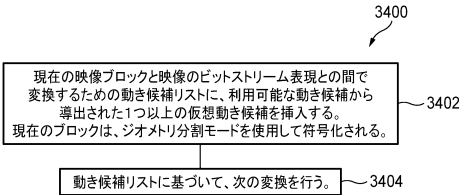


20

【図 3 3】



【図 3 4】

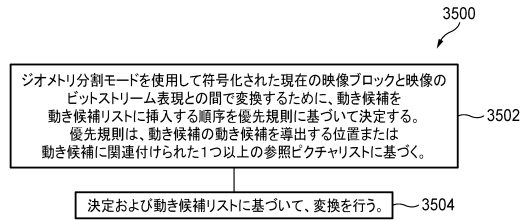


30

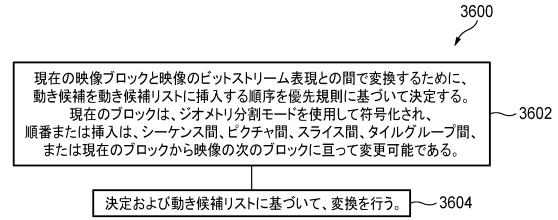
40

50

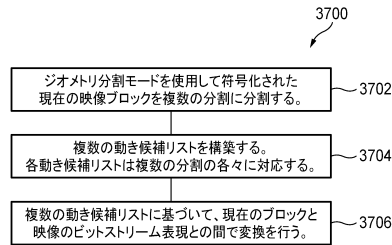
【図 35】



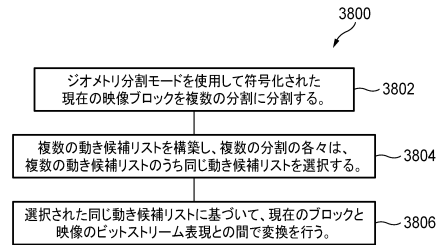
【図 36】



【図 37】

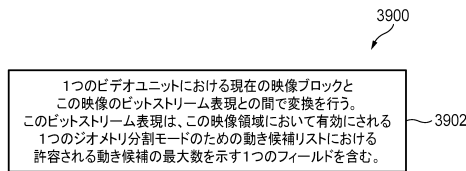


【図 38】

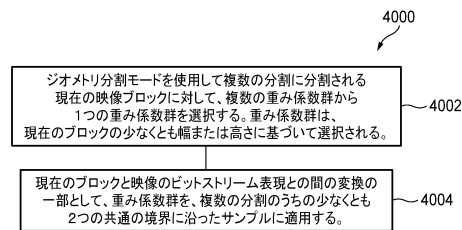


10

【図 39】

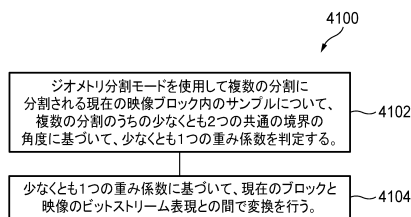


【図 40】

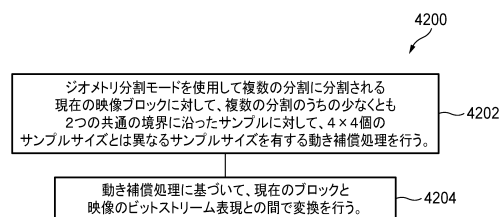


20

【図 41】



【図 42】

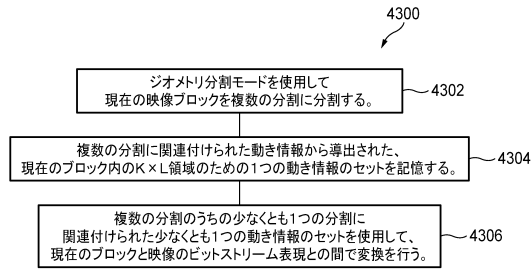


30

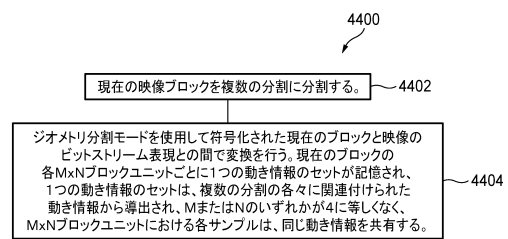
40

50

【図 4 3】

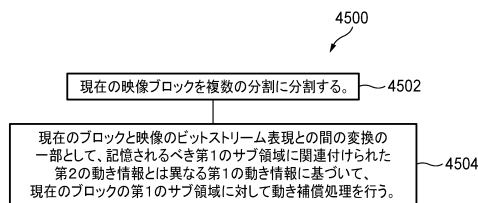


【図 4 4】

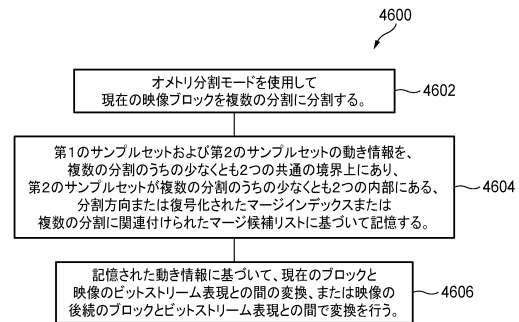


10

【図 4 5】

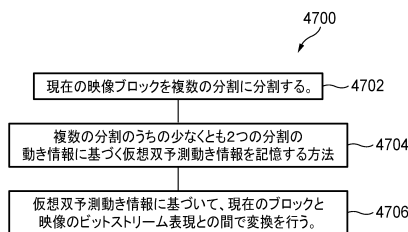


【図 4 6】

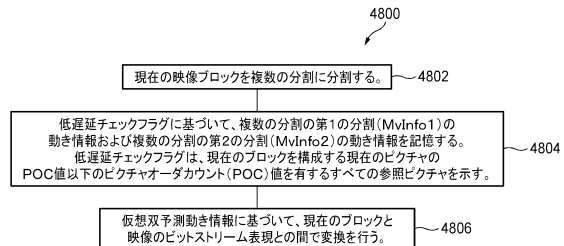


20

【図 4 7】



【図 4 8】

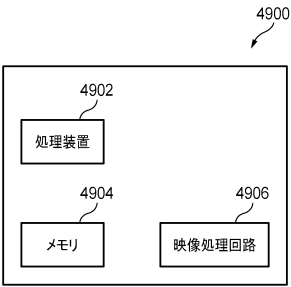


30

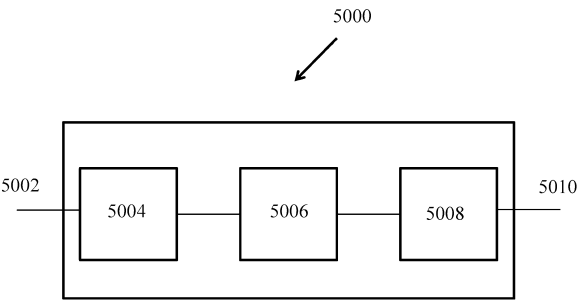
40

50

【図 49】



【図 50】



10

20

30

40

50

フロントページの続き

中国(CN)

(31)優先権主張番号 PCT/CN2019/087372

(32)優先日 令和1年5月17日(2019.5.17)

(33)優先権主張国・地域又は機関

中国(CN)

(31)優先権主張番号 PCT/CN2019/090127

(32)優先日 令和1年6月5日(2019.6.5)

(33)優先権主張国・地域又は機関

中国(CN)

(31)優先権主張番号 PCT/CN2019/092151

(32)優先日 令和1年6月20日(2019.6.20)

(33)優先権主張国・地域又は機関

中国(CN)

BYTEDANCE INC.

アメリカ合衆国 カリフォルニア州 90066, ロサンゼルス, ウェスト ジェファーソン ブールヴァード 12655, シックス フロア, スイート ナンバー・137

12655 West Jefferson Boulevard, Sixth Floor, Suite No. 137 Los Angeles, California 90066 United States of America

(74)代理人 110002000

弁理士法人栄光事務所

(72)発明者 ジャン リー

アメリカ合衆国 90066 カリフォルニア州, ロサンゼルス, ウェスト ジェファーソン ブールバード 12655, シックス フロア, スイート ナンバー137

(72)発明者 ジャン カイ

アメリカ合衆国 90066 カリフォルニア州, ロサンゼルス, ウェスト ジェファーソン ブールバード 12655, シックス フロア, スイート ナンバー137

(72)発明者 リウ ホンビン

中華人民共和国 100080 ベイジン, ハイディアン ディストリクト, チーチン ロード, チャイナ サテライト コミュニケーションズ タワー, ナンバー63, ジンリトウティアオ ポストオフィス

(72)発明者 ワン ユエ

中華人民共和国 100080 ベイジン, ハイディアン ディストリクト, チーチン ロード, チャイナ サテライト コミュニケーションズ タワー, ナンバー63, ジンリトウティアオ ポストオフィス

(72)発明者 ジャン ナ

中華人民共和国 100080 ベイジン, ハイディアン ディストリクト, チーチン ロード, チャイナ サテライト コミュニケーションズ タワー, ナンバー63, ジンリトウティアオ ポストオフィス

合議体

審判長 畑中 高行

審判官 片岡 利延

審判官 圓道 浩史

(56)参考文献 Ru-Ling Liao and Chong Soon Lim, CE10. 3. 1. b: Triangular prediction unit mode, Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JVET-L0124-v2, 12th Meeting: Macao, CN, 2018年10月, pp. 1-8

(58)調査した分野 (Int.Cl., DB名)

H04N 19/00-19/98