

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号

特許第7328330号

(P7328330)

(45)発行日 令和5年8月16日(2023.8.16)

(24)登録日 令和5年8月7日(2023.8.7)

(51)国際特許分類

F I

H 0 4 N 19/52 (2014.01)

H 0 4 N 19/52

請求項の数 21 (全50頁)

(21)出願番号	特願2021-523109(P2021-523109)	(73)特許権者	520476341
(86)(22)出願日	令和1年7月1日(2019.7.1)		北京字節跳動網絡技術有限公司
(65)公表番号	特表2021-530183(P2021-530183 A)		Beijing Bytedance Network Technology Co., Ltd.
(43)公表日	令和3年11月4日(2021.11.4)		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/IB2019/055581		Room B-0035, 2/F, No. 30 Building, No. 30, Shixing Road, Shijingshan District Beijing 100041 China
(87)国際公開番号	WO2020/003274		
(87)国際公開日	令和2年1月2日(2020.1.2)		
審査請求日	令和3年2月24日(2021.2.24)		
(31)優先権主張番号	PCT/CN2018/093663	(73)特許権者	520477474
(32)優先日	平成30年6月29日(2018.6.29)		バイトダンス インコーポレイテッド
(33)優先権主張国・地域又は機関	中国(CN)		最終頁に続く
(31)優先権主張番号	PCT/CN2018/101220		
(32)優先日	平成30年8月18日(2018.8.18)		
	最終頁に続く		

(54)【発明の名称】 LUTにおける動き候補のチェック順序

(57)【特許請求の範囲】

【請求項1】

1 または複数のテーブルを維持することであって、各テーブルは、コーディングされた1 または複数の映像ブロックから導出された1 または複数の動き候補を含む、ことと、

映像の現在の映像ブロックに対する候補リストを構成することであって、前記テーブルにおける少なくとも1つの動き候補は、前記少なくとも1つの動き候補の少なくとも1つのインデックスの順にてチェックされ、前記チェックすることは、条件を満たした際に終了され、前記条件は、前記現在の映像ブロックのコーディングモードに基づく、ことと、

前記候補リストを用いて、前記現在の映像ブロックの動き情報を決定することと、

前記決定された動き情報に基づいて、前記現在の映像ブロックをコーディングすることとを有し、

a_0 、 $a_0 - T_0$ 、 $a_0 - T_0 - T_1$ 、 $a_0 - T_0 - T_1 - T_2$ 、...、 $a_0 - T_0 - T_1 - T_2 - \dots - T_{K-1}$ と等しいインデックスを有する、前記テーブルからのK個の動き候補は、K個のインデックスの順にチェックされ、 a_0 と T_i (i は0...K-1である)は整数値であり、 a_0 は前記テーブルにおける最大のインデックスを有する最後の動き候補のインデックスであり、

新しい動き候補が前記テーブルに追加された際に、前記新しい動き候補は、前記テーブルにおける他の動き候補よりも大きいインデックスを有し、

前記条件は、前記候補リストにおける動き候補の数が、前記候補リストの最大許容候補数－閾値に達することを含み、前記閾値は、前記現在の映像ブロックの前記コーディングモ

10

20

ードに依存する、映像処理方法。

【請求項 2】

前記コーディングすることは、前記現在の映像ブロックをビットストリーム表現に符号化することを有する、請求項 1 に記載の方法。

【請求項 3】

前記コーディングすることは、前記現在の映像ブロックをビットストリーム表現から復号化することを有する、請求項 1 に記載の方法。

【請求項 4】

前記各テーブルにおける前記動き候補の配置は、前記テーブルへの前記動き候補の追加の順に基づき、

前記テーブルにおける前記動き候補のインデックスは、前記テーブルへの前記動き候補の追加の前記順に関連付けられている、請求項 1 に記載の方法。

【請求項 5】

前記テーブルにおいて、K 個の最大のインデックスを有する最後の K 個の動き候補、または、K 個の最小のインデックスを有する最初の K 個の動き候補がチェックされる、請求項 1 に記載の方法。

【請求項 6】

前記テーブルの前記少なくとも 1 つの動き候補は、前記少なくとも 1 つのインデックスの降順に基づいてチェックされる、請求項 1 に記載の方法。

【請求項 7】

T_i は、同じ値を有する、請求項 1 に記載の方法。

【請求項 8】

前記候補リストは、マージ候補リストである、請求項 6 に記載の方法。

【請求項 9】

前記テーブルの前記少なくとも 1 つの動き候補は、少なくとも 1 つのインデックスの昇順に基づいてチェックされる、請求項 1 に記載の方法。

【請求項 10】

チェック対象の前記少なくとも 1 つの動き候補は、前記現在の映像ブロックのコーディングされた情報に基づいて決定され、

前記コーディングされた情報は、前記現在の映像ブロックのコーディングモード、前記テーブルにおける動き候補の数、前記テーブルから新たな動き候補が追加される前の候補リストにおける動き候補の数、前記現在の映像ブロックと前記テーブルの前記動き候補と関連する位置との間の距離、前記現在の映像ブロックのブロック形状、前記現在の映像ブロックのブロックサイズ、の少なくとも 1 つを含む、請求項 1 に記載の方法。

【請求項 11】

前記現在の映像ブロックの前記コーディングモードは、AMVP モード、マージモード、アフィンモード、非アフィン AMVP モード、非アフィンマージモード、アフィン AMVP (インター) モード、アフィンマージモード、CPR (Current Picture Reference) モード、非 CPR モード、のうちの 1 つである、請求項 10 に記載の方法。

【請求項 12】

前記条件は更に、テーブルからのチェック対象の許容動き候補の最大数、前記候補リストの許容動き候補の最大数、の少なくとも 1 つに基づく、請求項 1 に記載の方法。

【請求項 13】

チェック対象の前記テーブルにおける動き候補の最大数は、前記現在の映像ブロックのコーディングモード、前記テーブルにおける動き候補の数、前記テーブルから追加される許容動き候補の最大数、前記テーブルから前記動き候補をチェックする前の候補リストにおける利用可能な動き候補の数、前記現在の映像ブロックのブロックサイズ、前記現在の映像ブロックのブロック形状、スライスタイプ、の少なくとも 1 つに基づく請求項 1 に記載の方法。

10

20

30

40

50

【請求項 14】

前記現在の映像ブロックのコーディングモードが A M V P モードである場合、チェック対象の前記テーブルにおける動き候補の最大数は、4 以下である、請求項 1 に記載の方法。

【請求項 15】

チェック対象の前記テーブルにおける動き候補の最大数は、予め定義される、または、信号通知される、請求項 1 に記載の方法。

【請求項 16】

前記現在の映像ブロックが A M V P モードにてコーディングされた場合、前記テーブルから前記候補リストに追加される許容動き候補の最大数は、1、2、または 4 のうちのいずれかである、請求項 1 に記載の方法。

【請求項 17】

前記テーブルにおける動き候補は、予測方向、参照ピクチャインデックス、動きベクトル値、強度補償フラグ、アフィンフラグ、動きベクトル差精度、動きベクトル差分値、のうちの少なくとも 1 つを含む動き情報に関連付けられる、請求項 1 に記載の方法。

【請求項 18】

前記チェックされた動き候補に基づいて前記候補リストを更新するか否かは、前記チェックの結果に基づく、請求項 1 に記載の方法。

【請求項 19】

プロセッサと、命令を有する非一時的メモリとを備える映像処理装置であって、

前記命令は、前記プロセッサによって実行された際に、前記プロセッサに、

1 または複数のテーブルを維持することであって、各テーブルは、コーディングされた 1 または複数の映像ブロックから導出された 1 または複数の動き候補を含む、ことと、

映像の現在の映像ブロックに対する候補リストを構成することであって、前記テーブルにおける少なくとも 1 つの動き候補は、前記少なくとも 1 つの動き候補の少なくとも 1 つのインデックスの順にてチェックされ、前記チェックすることは、条件を満たした際に終了され、前記条件は、前記現在の映像ブロックのコーディングモードに基づく、ことと、

前記候補リストを用いて、前記現在の映像ブロックの動き情報を決定することと、

前記決定された動き情報に基づいて、前記現在の映像ブロックをコーディングすることと
を行わせ、

a_0 、 $a_0 - T_0$ 、 $a_0 - T_0 - T_1$ 、 $a_0 - T_0 - T_1 - T_2$ 、 $\dots$ 、 $a_0 - T_0 - T_1 - T_2 - \dots - T_{K-1}$ と等しいインデックスを有する、前記テーブルからの K 個の動き候補は、K 個のインデックスの順にチェックされ、 a_0 と T_i (i は $0 \dots K-1$ である) は整数値であり、 a_0 は前記テーブルにおける最大のインデックスを有する最後の動き候補のインデックスであり、

新しい動き候補が前記テーブルに追加された際に、前記新しい動き候補は、前記テーブルにおける他の動き候補よりも大きいインデックスを有し、

前記条件は、前記候補リストにおける動き候補の数が、前記候補リストの最大許容候補数 - 閾値に達することを含み、前記閾値は、前記現在の映像ブロックの前記コーディングモードに依存する、映像処理装置。

【請求項 20】

プロセッサに、

1 または複数のテーブルを維持することであって、各テーブルは、コーディングされた 1 または複数の映像ブロックから導出された 1 または複数の動き候補を含む、ことと、

映像の現在の映像ブロックに対する候補リストを構成することであって、前記テーブルにおける少なくとも 1 つの動き候補は、前記少なくとも 1 つの動き候補の少なくとも 1 つのインデックスの順にてチェックされ、前記チェックすることは、条件を満たした際に終了され、前記条件は、前記現在の映像ブロックのコーディングモードに基づく、ことと、

前記候補リストを用いて、前記現在の映像ブロックの動き情報を決定することと、

前記決定された動き情報に基づいて、前記現在の映像ブロックをコーディングすることと

10

20

30

40

50

を行わせ、

a_0 、 $a_0 - T_0$ 、 $a_0 - T_0 - T_1$ 、 $a_0 - T_0 - T_1 - T_2$ 、 $\dots$ 、 $a_0 - T_0 - T_1 - T_2 - \dots - T_{K-1}$ と等しいインデックスを有する、前記テーブルからのK個の動き候補は、K個のインデックスの順にチェックされ、 a_0 と T_i (i は $0 \dots K-1$ である)は整数値であり、 a_0 は前記テーブルにおける最大のインデックスを有する最後の動き候補のインデックスであり、

新しい動き候補が前記テーブルに追加された際に、前記新しい動き候補は、前記テーブルにおける他の動き候補よりも大きいインデックスを有し、

前記条件は、前記候補リストにおける動き候補の数が、前記候補リストの最大許容候補数－閾値に達することを含み、前記閾値は、前記現在の映像ブロックの前記コーディングモードに依存する、命令を格納した非一時的コンピュータ可読記憶媒体。

10

【請求項21】

映像のビットストリームを格納するための方法であって、

1または複数のテーブルを維持することであって、各テーブルは、コーディングされた1または複数の映像ブロックから導出された1または複数の動き候補を含む、ことと、

映像の現在の映像ブロックに対する候補リストを構成することであって、前記テーブルにおける少なくとも1つの動き候補は、前記少なくとも1つの動き候補の少なくとも1つのインデックスの順にてチェックされ、前記チェックすることは、条件を満たした際に終了され、前記条件は、前記現在の映像ブロックのコーディングモードに基づく、ことと、

前記候補リストを用いて、前記現在の映像ブロックの動き情報を決定することと、

20

前記決定された動き情報に基づいて、前記現在の映像ブロックから前記ビットストリームを生成することと、

非一時的コンピュータ記録媒体に前記ビットストリームを格納することと、

を有し、

a_0 、 $a_0 - T_0$ 、 $a_0 - T_0 - T_1$ 、 $a_0 - T_0 - T_1 - T_2$ 、 $\dots$ 、 $a_0 - T_0 - T_1 - T_2 - \dots - T_{K-1}$ と等しいインデックスを有する、前記テーブルからのK個の動き候補は、K個のインデックスの順にチェックされ、 a_0 と T_i (i は $0 \dots K-1$ である)は整数値であり、 a_0 は前記テーブルにおける最大のインデックスを有する最後の動き候補のインデックスであり、

新しい動き候補が前記テーブルに追加された際に、前記新しい動き候補は、前記テーブルにおける他の動き候補よりも大きいインデックスを有し、

30

前記条件は、前記候補リストにおける動き候補の数が、前記候補リストの最大許容候補数－閾値に達することを含み、前記閾値は、前記現在の映像ブロックの前記コーディングモードに依存する、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6月29日出願の国際特許出願PCT/CN2018/093663号、2018年8月18日出願の国際特許出願PCT/CN2018/101220号、2018年9月12日出願の国際特許出願PCT/CN2018/105193、および2018年11月27日出願の国際特許出願PCT/CN2018/117627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。米国の法律の下、あらゆる目的のために、国際特許出願PCT/CN2018/093663号、国際特許出願第PCT/CN2018/101220号、国際特許出願PCT/CN2018/105193号および国際特許出願第PCT/CN2018/117627の開示の全文は、本願の開示の一部として参照により援用される。

40

【0002】

本特許明細書は、映像符号化および復号化技術、デバイスおよびシステムに関する。

50

【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期待される。

【発明の概要】

【0004】

本明細書は、デジタル映像を符号化および復号化するための方法、システム、およびデバイスを開示する。

【0005】

一つの例示的な態様において、映像処理方法が開示される。この方法は、テーブルを維持することであって、各テーブルは、動き候補のセットを含み、各動き候補は、対応する動き情報に関連付けられている、ことと、少なくとも1つ以上のテーブルに基づいて、第1の映像ブロックと、第1の映像ブロックを含む映像のビットストリーム表現との間の変換を行うことであって、変換を行うことは、チェック終了判定基準が満たされるまで規則で決定された順にテーブル内の動き候補をチェックすることを有する。

【0006】

さらに別の例示的な態様において、本明細書で説明される映像符号化方法を実装する映像エンコーダ装置が開示される。

【0007】

さらに別の代表的な態様では、本明細書で説明される様々な技法は、非一時的なコンピュータ可読媒体に記憶されるコンピュータプログラム製品として実施され得る。このコンピュータプログラム製品は、本明細書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。

【0008】

さらに別の代表的な態様において、映像デコーダ装置は、本明細書で説明されるような方法を実装してもよい。

【0009】

1つ以上の実装形態の詳細は、添付の添付ファイル、図面、および以下の説明に記載されている。他の特徴は、説明および図面、並びに特許請求の範囲の記載から明らかとなるう。

【図面の簡単な説明】

【0010】

【図1】映像エンコーダの実装形態の例を示すブロック図である。

【図2】H. 264映像符号化規格におけるマクロブロックの分割を示す。

【図3】符号化ブロック(CB: Coding Block)を予測ブロック(PU: Prediction Block)に分割する例を示す。

【図4】CTBをCBおよび変換ブロック(TB)に細分するための例示的な実装形態を示す。実線はCB境界を示し、点線はTB境界を示し、その分割を含むCTBの例、および対応する4分木を含む。

【図5】映像データを分割するための4分木2分木(QTBT: Quad Tree Binary Tree)構造の一例を示す。

【図6】映像ブロックの分割の例を示す。

【図7】4分木の分割の例を示す。

【図8】ツリー型信号通知の例を示す。

【図9】マージ候補リスト構築のための導出処理の一例を示す。

【図10】空間的マージ候補の位置の例を示す。

【図11】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12】 $N \times 2N$ および $2N \times N$ パーティションの第2のPUの位置の例を示す。

10

20

30

40

50

- 【図 1 3】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を示す。
- 【図 1 4】時間的マージ候補の候補位置とその同一位置のピクチャを示す。
- 【図 1 5】結合双方向予測マージ候補の例を示す。
- 【図 1 6】動きベクトル予測候補の導出処理の例を示す。
- 【図 1 7】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を示す。
- 【図 1 8】CUの動き予測のための例示的なATMVP (Alternative Temporal Motion Vector Prediction) を示す。
- 【図 1 9】ソースブロックおよびソースピクチャの識別の一例を絵で示す。
- 【図 2 0】4つのサブブロックおよび近傍のブロックを有する1つのCUの例を示す。
- 【図 2 1】バイラテラルマッチングの例を示す。
- 【図 2 2】テンプレートマッチングの例を示す。
- 【図 2 3】FRUC (Frame Rate Up Conversion) における一方の動き推定 (ME: Motion Estimation) の例を示す。
- 【図 2 4】バイラテラルテンプレートマッチングに基づくDMVRの例を示す。
- 【図 2 5】空間的マージ候補を導出するために使用する空間的に近傍のブロックの例を示す。
- 【図 2 6】ルックアップテーブル更新のための代表的な位置の選択方法の一例を示す。
- 【図 2 7 A】新しい動き情報のセットでルックアップテーブルを更新する例を示す。
- 【図 2 7 B】新しい動き情報のセットでルックアップテーブルを更新する例を示す。
- 【図 2 8】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 2 9】映像ビットストリーム処理の例示的な方法のフローチャートである。
- 【図 3 0】提案されるHMVP方法による復号化フローチャートの一例を示す。
- 【図 3 1】提案されるHMVP方法を用いたテーブルの更新例を示す。
- 【図 3 2 A】冗長性除去に基づくLUT更新方法 (1つの冗長性動き候補を除去する) の例を示す。
- 【図 3 2 B】冗長性除去に基づくLUT更新方法 (1つの冗長性動き候補を除去する) の例を示す。
- 【図 3 3 A】冗長性削除に基づくテーブル更新方法 (複数の冗長性動き候補を除去する) の例を示す。
- 【図 3 3 B】冗長性削除に基づくテーブル更新方法 (複数の冗長性動き候補を除去する) の例を示す。
- 【発明を実施するための形態】

【0011】

映像の圧縮率を改善するために、研究者らは、映像を符号化する新しい技術を絶えず求めている。

【0012】

1. 導入

【0013】

本明細書は、映像符号化技術に関する。具体的には、映像符号化における動き情報の符号化 (例えば、マージモード、AMVPモード) に関する。HEVCのような既存の映像符号化規格に適用してもよいし、規格 (Versatile Video Coding) を確定させるために適用してもよい。本発明は、将来の映像符号化規格または映像コーデックにも適用可能である。

【0014】

簡単な説明

【0015】

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を作り、ISO/IECはMPEG-

10

20

30

40

50

1とMPEG-4 Visualを作り、両団体はH.262/MPEG-2 VideoとH.264/MPEG-4 AVC (Advanced Video Coding)とH.265/HEVC規格を共同で作った。H.262以来、映像符号化規格は、時間予測と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。典型的なHEVCエンコーダフレームワークの一例を図1に示す。

【0016】

2.1 パーティション構造

【0017】

2.1.1 H.264/AVCにおけるパーティションツリー構造

【0018】

以前の規格における符号化層のコアは、 16×16 ブロックの輝度サンプルを含み、通常の4:2:0カラーサンプリングの場合、2つの対応する 8×8 ブロックの彩度サンプル含むマクロブロックであった。

【0019】

イントラ符号化されたブロックは、画素間の空間的相関を利用するために空間予測を使用する。2つのパーティションを規定する。 16×16 および 4×4 である。

【0020】

インター符号化されたブロックは、ピクチャ間の動きを推定することで、空間的予測の代わりに時間予測を用いる。動きは、 16×16 マクロブロックまたはそのサブマクロブロックパーティションのいずれかに対して独立して推定できる。 16×8 、 8×16 、 8×8 、 8×4 、 4×8 、 4×4 (図2参照)。1つのサブマクロブロックパーティション当たり1つの動きベクトル(MV: Motion Vector)のみが許可される。

【0021】

2.1.2 HEVCにおけるパーティションツリー構造

【0022】

HEVCにおいて、CTUは、様々な局所的特徴に適応するように、符号化ツリーと呼ばれる4分木構造を用いてCUに分割される。インターピクチャ(時間的)予測またはイントラピクチャ(空間的)予測を使用した、ピクチャ領域を符号化するかどうかの決定は、CUレベルで行われる。各CUは、PU分割タイプに応じて1つ、2つまたは4つのPUに更に分割することができる。1つのPUの内部では、同じ予測処理が適用され、PU単位で関連情報がデコーダに送信される。PU分割タイプに基づく予測処理を適用して残差ブロックを得た後、CUのための符号化ツリーに類似した別の4分木構造に基づいて、CUを変換ユニット(TU)に分割することができる。HEVC構造の重要な特徴の1つは、CU、PU、TUを含む複数のパーティション概念を有することである。

【0023】

以下、HEVCを使用したハイブリッド映像符号化に関連する様々な特徴に焦点を当てる。

【0024】

1) 符号化ツリーユニットおよび符号化ツリーブロック(CTB)構造: HEVCにおける類似した構造は、符号化ツリーユニット(CTU)であり、この符号化ツリーユニットは、エンコーダによって選択されたサイズを有し、従来のマクロブロックよりも大きくてもよい。CTUは、輝度CTBと、対応する彩度CTBおよび構文要素とからなる。輝度CTBのサイズ $L \times L$ は、 $L = 16$ 、 32 、または 64 のサンプルとして選択することができ、より大きいサイズは、一般的に、より優れた圧縮を有効にする。HEVCは、次いで、ツリー構造および4分木様の信号通知を使用して、CTBをより小さなブロックに分割することをサポートする。

【0025】

2) 符号化ユニット(CU)および符号化ブロック(CB): CTUの4分木の構文は、その輝度および彩度CBのサイズおよび位置を指定する。4分木のルートはCTUに関連付けられる。従って、輝度CTBのサイズは、輝度CBに対してサポートされる最大の

10

20

30

40

50

サイズである。C T Uを輝度C Bおよび彩度C Bに分割することは、共に信号通知されることである。1つの輝度C Bおよび通常2つの彩度C Bは、関連する構文と共に、1つの符号化ユニット(C U)を形成する。C T Bは、1つのC Uのみを含んでもよく、または複数のC Uを形成するように分割されてもよく、各C Uは、それに関連付けられた予測ユニット(P U)への分割と、1つの変換ユニット(T U)のツリーとを有する。

【0026】

3) 予測ユニットおよび予測ブロック(P B)：インターピクチャまたはイントラピクチャ予測を使用してピクチャ領域を符号化するかどうかの決定は、C Uレベルで行われる。P Uのパーティショニング構造は、そのルートがC Uレベルにある。基本的な予測タイプの決定に基づいて、次に、輝度および彩度C Bのサイズをさらに分割し、輝度および彩度予測ブロック(P B)から予測することができる。H E V Cは、 64×64 から 4×4 までの可変P Bサイズのサンプルをサポートする。図3は、 $M \times M$ のC Uのための許可されたP Bの例を示す。

10

【0027】

4) T Uおよび変換ブロック：予測残差は、ブロック変換を使用して符号化される。T Uツリー構造は、そのルートがC Uレベルにある。この輝度C B残差は、輝度変換ブロック(T B)と同一であってもよいし、小さな輝度T Bにさらに分割されてもよい。彩度T Bについても同様である。正方形T Bサイズ 4×4 、 8×8 、 16×16 、および 32×32 に対して、離散コサイン変換(D C T)の整数基底関数に類似した整数基底関数が規定される。輝度イントラピクチャ予測残差の 4×4 変換のために、離散サイン変換(D S T)の形式から導出される整数変換が代替的に指定される。

20

【0028】

図4は、C T BをC B [及び変換ブロック(T B)] に細分する例を示す。実線はC B境界を示し、点線はT B境界を示す。(a) C T Bとその分割 (b) 対応する4分木。

【0029】

2. 1. 2. 1 変換ブロックおよびユニットへのツリー構造の分割

【0030】

残差符号化の場合、C Bは、変換ブロック(T B)に再帰的に分割することができる。この分割は、残差4分木によって信号通知される。図4に示すように、1つのブロックを再帰的に象限に分割することができるように、正方形のC BおよびT Bの分割のみを指定する。サイズ $M \times M$ の所与の輝度C Bに対して、フラグは、それがサイズ $M/2 \times M/2$ の4つのブロックに分割されるかどうかを信号通知する。さらなる分割が可能である場合、S P Sに示される残留4分木の最大深さによって信号通知されるように、各象限には、それが4つの象限に分割されているかどうかを示すフラグが割り当てられる。残差4分木の結果得られる葉ノードブロックは、変換符号化によってさらに処理される変換ブロックである。エンコーダは、それが使用することになる最大および最小輝度T Bサイズを示す。C Bサイズが最大T Bサイズよりも大きい場合、分割は非明示的に行われる。分割により、示された最小値よりも小さい輝度T Bサイズとなる場合、分割は、非明示的に行われない。輝度T Bサイズが 4×4 である場合を除き、彩度T Bサイズは、各次元において輝度T Bサイズの半分であり、この場合、4つの 4×4 輝度T Bによって覆われる領域には1つの 4×4 彩度T Bが使用される。イントラピクチャ予測C Uの場合、最近の近傍のT B(C B内またはC B外)の復号サンプルを、イントラピクチャ予測のための参照データとして用いる。

30

40

【0031】

従来の規格とは対照的に、H E V C設計により、インターピクチャ予測C Uのために1つのT Bが複数のP Bにまたがることが可能となり、4分木構造のT Bの分割の潜在的な符号化効率の利点が最大となる。

【0032】

2. 1. 2. 2 親子ノード

【0033】

50

CTBは、4分木構造に基づいて分割され、そのノードは符号化ユニットである。4分木構造における複数のノードは、葉ノードおよび非葉ノードを含む。葉ノードは、ツリー構造内に子ノードを持たない（すなわち、葉ノードはそれ以上分割されない）。非葉ノードは、ツリー構造のルートノードを含む。ルートノードは、映像データの最初の映像ブロック（例えば、CTB）に対応する。複数のノードのうちのそれぞれの非ルートノードごとに、それぞれの非ルートノードは、それぞれの非ルートノードのツリー構造における親ノードに対応する映像ブロックのサブブロックである映像ブロックに対応する。複数の非葉ノードのそれぞれの非葉ノードは、ツリー構造において1つ以上の子ノードを有する。

【0034】

2.1.3 JEMにおけるより大きいCTUを有する4分木+2分木ブロック構造

【0035】

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JVET (Joint Video Exploration Team) 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、JEM (Joint Exploration Mode) と呼ばれる参照ソフトウェアに組み込まれてきた。

【0036】

2.1.3.1 QTBTブロックの分割構造

【0037】

HEVCとは異なり、QTBT構造は、複数のパーティションタイプを削除する。すなわち、CU、PU、TUのコンセプトの切り離しを取り除き、CUパーティションの形状の柔軟性を向上させる。QTBTブロック構造において、CUは正方形または長方形のいずれかを有することができる。図5に示すように、まず、符号化ツリーユニット (CTU) を4分木構造で分割する。4分木の葉ノードは、2分木構造によってさらに分割される。2分木の分割には、対称水平分割と対称垂直分割の2つの分割タイプがある。2分木の葉ノードは、符号化ユニット (CU) と呼ばれ、このセグメント化は、それ以上の分割を行うことなく、予測および変換処理に使用される。これは、QTBTの符号化ブロック構造において、CU、PUおよびTUが同じブロックサイズを有することを意味する。JEMにおいて、CUは、しばしば異なる色成分の符号化ブロック (CB) からなり、例えば、4:2:0彩度フォーマットのPおよびBスライスの場合、1つのCUは1つの輝度CBおよび2つの彩度CBを含み、また、CUは、しばしば単一の成分のCBからなり、例えば、Iスライスの場合、1つのCUは、1つの輝度CBのみ、または、2つの彩度CBのみを含む。

【0038】

QTBT分割スキームに対して以下のパラメータを規定する。

- CTUのサイズ：1つの4分木のルートノードのサイズ、HEVCと同じ概念
- MinQTSIZE：最小許容の4分木の葉ノードサイズ
- MaxBTSIZE：最大許容の2分木のルートノードサイズ
- MaxBTDepth：最大許容の2分木の深さ
- MinBTSIZE：最小許容の2分木の葉ノードのサイズ

【0039】

QTBTの分割構造の一例において、CTUのサイズを、2つの対応する64×64ブロックの彩度サンプルを有する128×128の輝度サンプルとして設定し、MinQTSIZEを16×16として設定し、MaxBTSIZEを64×64として設定し、MinBTSIZE（幅および高さの両方について）を4×4として設定し、MaxBTDepthを4として設定する。4分木の分割は、まずCTUに適用され、4分木の葉ノードを生成する。4分木の葉ノードのサイズは、16×16（即ち、MinQTSIZE）から128×128（即ち、CTUサイズ）までのサイズを有することが可能である。葉4分木のノードが128×128である場合、サイズがMaxBTSIZE（すなわち、64×64）を超えるため、2分木によってさらに分割されない。そうでない場合、葉4

10

20

30

40

50

分木のノードは、2分木によってさらに分割されてもよい。従って、この4分木の葉ノードは、2分木のルートノードでもあり、その2分木の深さは0である。2分木の深さが $MaxBTDepth$ （すなわち、4）に達した場合、それ以上の分割は考慮されない。2分木のノードの幅が $MinBTSize$ （すなわち、4）に等しい場合、それ以上の水平分割は考慮されない。同様に、2分木のノードの高さが $MinBTSize$ に等しい場合、それ以上の垂直分割は考慮されない。2分木の葉ノードは、さらに分割することなく、予測および変換処理によってさらに処理される。JEMにおいて、最大CTUサイズは、 256×256 個の輝度サンプルである。

【0040】

図5（左）はQTBTを用いたブロックの分割の例を示し、図5（右）は対応するツリー表現を示す。実線は4分木の分割を表し、点線は2分木の分割を表す。2分木の各分割（即ち、非葉）ノードにおいて、1つのフラグが、どの分割タイプ（即ち、水平または垂直）が使用されるかを示すために信号通知される。ここで、0は、水平分割を表し、1は、垂直分割を表す。4分木の分割の場合、4分木の分割は常にブロックを水平および垂直に分割し、等分したサイズの4つのサブブロックを生成するため、分割タイプを示す必要がない。

【0041】

さらに、QTBT方式は、輝度および彩度が別個のQTBT構造を有する能力をサポートする。現在、PおよびBスライスの場合、1つのCTUにおける輝度および彩度CTBは、同じQTBT構造を共有する。しかしながら、Iスライスの場合、輝度CTBはQTBT構造によってCUに分割され、彩度CTBは別のQTBT構造によって彩度CUに分割される。これは、1つのIスライスにおける1つのCUが1つの輝度成分の1つの符号化ブロックまたは2つの彩度成分の1つの符号化ブロックからなり、1つのPまたはBスライスにおける1つのCUが3つの色成分すべての符号化ブロックからなることを意味する。

【0042】

HEVCにおいて、小さなブロックのためのインター予測は、動き補償のメモリアクセスを低減するために制限され、その結果、 4×8 および 8×4 ブロックのために双方向予測はサポートされず、 4×4 ブロックのためにインター予測はサポートされない。JEMのQTBTにおいて、これらの制限は取り除かれる。

【0043】

2.1.4 VVCの3分木

【0044】

いくつかの実装形態において、4分木および2分木以外のツリータイプがサポートされる。本実装形態において、図6の（d）、（e）部に示すように、3分木（TT）分割を2つ以上、すなわち、水平および垂直の中心側の3分木を導入する。

【0045】

図6は、（a）4分木分割、（b）垂直2分木分割、（c）水平2分木分割、（d）垂直中心側3分木分割、（e）水平中心側3分木分割を示す。

【0046】

いくつかの実装形態において、2つのレベルのツリー、すなわち、領域ツリー（4分木）および予測ツリー（2分木または3分木）がある。CTUは、まず、領域ツリー（RT）によって分割される。RTリーフは、予測ツリー（PT）によってさらに分割されてもよい。PT葉はまた、最大PT深さに達するまで、PTでさらに分割されてもよい。PT葉が基本符号化ユニットである。便宜上、ここでもCUと呼ぶ。1つのCUをさらに分割することはできない。予測および変換は両方ともJEMと同様にCUに適用される。パーティション構造全体を「マルチタイプツリー」と呼ぶ。

【0047】

2.1.5 分割構造の例

【0048】

10

20

30

40

50

この応答で使用されるツリー構造は、マルチツリータイプ (Multi-Tree Type: MTT) と呼ばれ、QTB Tを一般化したものである。QTB Tにおいて、図5に示すように、まず、符号化ツリーユニット (CTU) を4分木構造で分割する。4分木の葉ノードは、2分木構造によってさらに分割される。

【0049】

MTTの基本構造は、2つのタイプのツリーノードを構成する。図7に示すように、領域ツリー (RT) および予測ツリー (PT) は、9つのタイプのパーティションをサポートする。

【0050】

図7は、(a) 4分木分割、(b) 垂直2分木分割、(c) 水平2分木分割、(d) 垂直3分木分割、(e) 水平3分木分割、(f) 水平上方非対称2分木分割、(g) 水平下方非対称2分木分割、(h) 垂直左非対称2分木分割、(i) 垂直右非対称2分木分割を示す。

【0051】

1つの領域ツリーは、1つのCTUを4×4サイズの領域ツリーの葉ノードになるように正方形のブロックに再帰的に分割することができる。領域ツリーにおける各ノードにおいて、予測ツリーは、2分木 (BT)、3分木 (TT)、および非対称2分木 (ABT) の3つのツリータイプのうちの1つから形成されることができる。PT分割において、予測ツリーの枝に4分木のパーティションを有することは禁止される。JEMにおけるように、輝度ツリーおよび彩度ツリーは、I個のスライスに分けられる。RTおよびPTの信号通知方法を図8に示す。

【0052】

2.2 HEVC/H.265におけるインター予測

【0053】

各インター予測された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、interpret_idcを使用して信号通知されてもよい。動きベクトルは、予測因子に関連する差分として明確に符号化されてもよく、このような符号化モードは、AMVPモードと呼ばれる。

【0054】

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、符号化された動きベクトル差分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としては、動きパラメータの明確な送信であり、各参照ピクチャリストおよび参照ピクチャリストの使用に対する参照ピクチャインデックスに対応する動きベクトルをPUごとに明確に信号通知することである。

【0055】

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、サンプルのうちの1つのブロックから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。

【0056】

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、サンプルのうちの2つのブロックからPUを生成する。これを「双方向予測」と呼ぶ。Bスライスだけに双予測が利用可能である。

【0057】

以下、HEVC [2] 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。

【0058】

2.2.1 マージモード

【0059】

2.2.1.1 マージモードの候補の導出

【0060】

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：初期候補の導出
 - ステップ1.1：空間的候補の導出
 - 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：時間的候補の導出
- ・ステップ2：追加候補の挿入
 - ステップ2.1：双方向予測候補の作成
 - ステップ2.2：動きゼロ候補の挿入

【0061】

これらのステップは図9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、候補数がスライスヘッダで信号通知されるマージ候補の最大数（MaxNumMergeCand）に達しない場合、追加候補を生成する。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は、短縮された単項2値化（TU：truncated unary binarization）を使用して符号化される。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。

【0062】

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。

【0063】

2.2.1.2 空間的候補の導出

【0064】

空間的マージ候補の導出において、図10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尔に属しているため）、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。代わりに、図11において矢印でリンクされた対のみを考慮し、冗長性チェックに使用される対応する候補が同じ動き情報を有していない場合にのみ、その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、 $2N \times 2N$ とは異なるパーティションに関連付けられた「第2のPU」である。一例として、図12は、それぞれ $N \times 2N$ および $2N \times N$ の場合の第2のPUを示す。現在のPUを $N \times 2N$ に分割する場合、リスト構成に位置 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、同じ動き情報を有する2つの予測ユニットが導かれることとなり、1つの符号化ユニットに1つのPUのみを有するためには冗長である。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。

【0065】

2.2.1.3 時間的候補の導出

【0066】

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のPU (co-located 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、同一位置の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13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、同一位置の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差として規定され、 t_d は、同一位置のPUの参照ピクチャと同一位置のピクチャの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、双方向予測マージ候補を形成する。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説明。

【0067】

参照フレームに属する同一位置のPU (Y) において、図14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。

【0068】

2.2.1.4 追加候補の挿入

【0069】

空間的-時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、結合双方向予測マージ候補およびゼロマージ候補がある。空間的-時間的マージ候補を利用して、結合双方向予測マージ候補を生成する。結合双方向予測マージ候補は、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、結合双方向予測候補を生成する。これら2つのタプルが異なる動きの仮説を提供する場合、これらのタプルは、新しい双方向予測候補を形成する。一例として、図15は、オリジナルリスト（左側）における、 $mvL0$ および $refIdxL0$ 、または $mvL1$ および $refIdxL1$ を有する2つの候補を用いて、最終リスト（右側）に加えられる結合双方向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。

【0070】

ゼロ動き候補を挿入し、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、双方向予測の場合は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。

【0071】

2.2.1.5 並列処理のための動き推定領域

【0072】

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: Motion Estimation Region) を規定

10

20

30

40

50

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。

7.3.2.3 ピクチャパラメータセットRBS P構文

7.3.2.3.1 一般ピクチャパラメータセットRBS P構文

【0073】

【表1】

<code>pic_parameter_set_rbsp()</code> {	記述子
<code>pps_pic_parameter_set_id</code>	<code>ue(v)</code>
<code>pps_seq_parameter_set_id</code>	<code>ue(v)</code>
<code>dependent_slice_segments_enabled_flag</code>	<code>u(1)</code>
...	
<code>pps_scaling_list_data_present_flag</code>	<code>u(1)</code>
<code>if(pps_scaling_list_data_present_flag)</code>	
<code>scaling_list_data()</code>	
<code>lists_modification_present_flag</code>	<code>u(1)</code>
<code>log2_parallel_merge_level_minus2</code>	<code>ue(v)</code>
<code>slice_segment_header_extension_present_flag</code>	<code>u(1)</code>
<code>pps_extension_present_flag</code>	<code>u(1)</code>
...	
<code>rbp_trailing_bits()</code>	
}	

【0074】

`log2_parallel_merge_level_minus2+2`は、8.5.3.2.2.2節で指定されたマージモードの輝度動きベクトルの導出処理と、8.5.3.2.3節で指定された空間的マージ候補の導出処理で使用される変数`Log2ParMrgLevel`の値を指定する。`log2_parallel_merge_level_minus2`の値は、 $0 \sim CtbLog2SizeY - 2$ を含む範囲内とする。変数`Log2ParMrgLevel`は、以下のように導出される。

$$Log2ParMrgLevel = log2_parallel_merge_level_minus2 + 2 \quad (7-37)$$

注3：`Log2ParMrgLevel`の値は、マージ候補リストを並列に導出する組み込み能力を示す。例えば、`Log2ParMrgLevel`が6に等しい場合、 64×64 ブロックに含まれたすべての予測ユニット（PU）および符号化ユニット（CU）のためのマージ候補リストを並列に導出することができる。

【0075】

2.2.2 AMVPモードにおける動きベクトル予測

【0076】

動きベクトル予測は、動きベクトルと近傍のPUとの間の空間的・時間的相関を利用し、これを動きパラメータの明確な伝送に用いる。まず、左側、上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築する。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。この場合のエンコーディング対象の最大値は2である（例えば、図2～図8）。以下の章では、動きベクトル

ル予測候補の導出処理の詳細を説明する。

【0077】

2.2.2.1 動きベクトル予測候補の導出

【0078】

図16に、動きベクトル予測候補の導出処理をまとめる。

【0079】

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11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。

【0080】

時間的動きベクトル候補の導出のために、2つの異なる同一位置の配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。空間的・時間的候補の最初の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。可能性のある候補の数が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。空間的・時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。

【0081】

2.2.2.2 空間的動きベクトル候補

【0082】

空間的動きベクトル候補の導出において、図11に示したような位置にあるPUから導出された5つの可能性のある候補のうち、動きマージと同じ位置にあるものを最大2つの候補を考慮する。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が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、かつ、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、かつ、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、かつ、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、かつ、異なる参照ピクチャ (異なるPOC)

【0083】

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、左側および上側の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。

【0084】

空間的スケーリング処理において、図17に示すように、時間的スケーリングと同様にして、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。

【0085】

10

20

30

40

50

2.2.2.3 時間的動きベクトル候補

【0086】

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。

【0087】

2.2.2.4 AMVP情報の信号通知

【0088】

AMVPモードの場合、ビットストリームにおいて、4つの部分、すなわち、予測方向、参照インデックス、MVD、およびmv予測因子候補インデックスが信号通知される。

構文テーブル：

【0089】

【表2】

prediction_unit(x0, y0, nPbW, nPbH) {	記述子
if(cu_skip_flag[x0][y0]) {	
if(MaxNumMergeCand > 1)	
merge_idx [x0][y0]	ae(v)
} else { /* MODE_INTER */	
merge_flag [x0][y0]	ae(v)
if(merge_flag[x0][y0]) {	
if(MaxNumMergeCand > 1)	
merge_idx [x0][y0]	ae(v)
} else {	
if(slice_type == B)	
inter_pred_idc [x0][y0]	ae(v)
if(inter_pred_idc[x0][y0] != PRED_L1) {	
if(num_ref_idx_l0_active_minus1 > 0)	
ref_idx_l0 [x0][y0]	ae(v)
mvd_coding(x0, y0, 0)	
mvp_l0_flag [x0][y0]	ae(v)
}	
if(inter_pred_idc[x0][y0] != PRED_L0) {	
if(num_ref_idx_l1_active_minus1 > 0)	
ref_idx_l1 [x0][y0]	ae(v)
if(mvd_l1_zero_flag && inter_pred_idc[x0][y0] == PRED_BI)	
{	
MvdL1[x0][y0][0] = 0	
MvdL1[x0][y0][1] = 0	
} else	
mvd_coding(x0, y0, 1)	
mvp_l1_flag [x0][y0]	ae(v)
}	
}	
}	
}	
}	

【0090】

7.3.8.9 動きベクトル差構文

【0091】

10

20

30

40

50

【表 3】

mvd_coding(x0, y0, refList) {	記述子
abs_mvd_greater0_flag[0]	ae(v)
abs_mvd_greater0_flag[1]	ae(v)
if(abs_mvd_greater0_flag[0])	
abs_mvd_greater1_flag[0]	ae(v)
if(abs_mvd_greater0_flag[1])	
abs_mvd_greater1_flag[1]	ae(v)
if(abs_mvd_greater0_flag[0]) {	
if(abs_mvd_greater1_flag[0])	
abs_mvd_minus2[0]	ae(v)
mvd_sign_flag[0]	ae(v)
}	
if(abs_mvd_greater0_flag[1]) {	
if(abs_mvd_greater1_flag[1])	
abs_mvd_minus2[1]	ae(v)
mvd_sign_flag[1]	ae(v)
}	
}	

10

20

【0092】

2.3 JEM (Joint Exploration Model) における新しいインター予測方法

【0093】

2.3.1 サブCUに基づく動きベクトル予測

【0094】

QTTB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のセットを有することができる。エンコーダにおいて、大きなCUをサブCUに分割し、大きな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。ATMVP (Alternative Temporal Motion Vector Prediction) 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。STMVP (Spatial-Temporal Motion Vector Prediction) 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。

30

【0095】

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。

40

【0096】

2.3.1.1 代替の時間的動きベクトル予測

【0097】

ATMVP (Alternative Temporal Motion Vector Prediction) において、TMVP (Temporal Motion Vector Prediction) 法は、現在のCUより小さいブロックから複数セットの動き情報（動きベクトルおよび参照インデックスを含む）をフェッチすることで修正される。図18に示すように、サブCUは、正方形のN×Nブロックである（デフォルトでは、Nは4に設定される）。

50

【0098】

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動きソースピクチャと呼ぶ。第2のステップは、図18に示すように、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の動きベクトルならびに参照インデックスを取得する。

【0099】

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決定する。近傍のブロックの繰り返し走査処理を回避するために、dCUのマージ候補リストにおける最初のマージ候補を用いる。最初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動き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。1つの例において、最初のマージ候補が左側の近傍のブロック（即ち、図19のA₁）からのものである場合、関連するMVおよび参照ピクチャを利用して、ソースブロックおよびソースピクチャを特定する。

【0100】

図19は、ソースブロックおよびソースピクチャの特定の例を示す。

【0101】

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、動き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×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（すなわち、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。

【0102】

2.3.1.2 空間的-時間的動きベクトル予測

【0103】

この方法において、サブCUの動きベクトルは、ラスタスキャンの順に沿って再帰的に導出される。図20にこの概念を示す。4つの4×4サブCUである、A、B、C、およびDを含む8×8CU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。

【0104】

サブCU Aの動き導出は、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×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×N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の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、またはイントラコーディングされている場合、サブCUのAの左側の他のブロックをチェックする（ブロックbから始まり、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、所与のリストの最初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（Temporal Motion Vector Predictor）導出と同様の手順に従って、サブブロックAのTMVPを導出する。位置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、

10

20

30

40

50

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。

【0105】

図20は、4つのサブブロック（A-D）およびその近傍のブロックを有する1つのCUの例を示す。

【0106】

2.3.1.3 サブCU動き予測モード信号通知

【0107】

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。

【0108】

JEMにおいて、マージインデックスのすべてのピンは、CABACによって符号化されたコンテキストである。一方、HEVCにおいては、最初のピンのみが符号化されたコンテキストであり、残りのピンはバイパス符号化されたコンテキストである。

【0109】

2.3.2 適応型動きベクトル差解像度

【0110】

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分（MVD: Motion Vector Difference）（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、LAMVR（Locally Adaptive Motion Vector Resolution）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コーディングできる。MVD解像度は符号化ユニット（CU）レベルで制御され、MVD解像度フラグは、少なくとも1つの非ゼロMVDの構成要素を有する各CUに対して条件付きで信号通知される。

【0111】

少なくとも1つの非ゼロMVDの構成要素を有するCUの場合、 $1/4$ 輝度サンプルMV精度がCUにおいて使用されるか否かを示すために、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。

【0112】

CUの最初の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。

【0113】

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に用いるかを決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。

【0114】

10

20

30

40

50

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、更なる小範囲の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。

【0115】

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チェックは省略される。

【0116】

2.3.3 パターンマッチング動きベクトル導出

【0117】

PMMVD (Pattern Matched Motion Vector Derivation) モードは、FRUC (Frame-Rate Up Conversion) 技術に基づく特殊マージモードである。このモードでは、ブロックの動き情報は信号通知されず、デコーダ側で導出される。

【0118】

そのマージフラグが真である場合、FRUCフラグは、CUに信号通知される。FRUCフラグが偽である場合、マージインデックスは信号通知され、通常のマージモードが使用される。FRUCフラグが真である場合、追加のFRUCモードフラグを信号通知して、どの方法（バイラテラルマッチングまたはテンプレートマッチング）を使用してブロックの動き情報を導出するかを示す。

【0119】

エンコーダ側では、CUのためにFRUCマージモードを使用するかどうかの決定は、通常のマージ候補に対して行われるのと同じように、RDコスト選択に基づく。つまり、RDコスト選択を使用して、1つのCUに対して2つのマッチングモード（バイラテラルマッチングおよびテンプレートマッチング）を両方チェックする。最小コストに導くものが、更に、他のCUモードと比較される。FRUCマッチングモードが最も効率的なものである場合、CUに対してFRUCフラグを真に設定し、関連するマッチングモードを使用する。

【0120】

FRUCマージモードにおける動き導出処理は、2つのステップを有する。まず、CUレベルの動き探索を実行し、次に、サブCUレベルの動き改良を実行する。CUレベルでは、バイラテラルマッチングまたはテンプレートマッチングに基づいて、CU全体のための初期の動きベクトルを導出する。まず、MV候補のリストを生成し、最小マッチングコストに導く候補を、さらなるCUレベル改善の開始点として選択する。そして、開始点付近のバイラテラルマッチングまたはテンプレートマッチングに基づく局所検索を行い、最小マッチングコストとなるMV結果をCU全体のMVとする。続いて、導出されたCU動きベクトルを開始点として、サブCUレベルでの動き情報をさらに改良する。

【0121】

例えば、 $W \times H$ CU動き情報導出のために、以下の導出処理を行う。第1のステージにおいて、 $W \times H$ CU全体のためのMVが導出される。第2のステージにおいて、CUは、 $M \times M$ のサブCUにさらに分割される。Mの値は、(16)のように計算されるが、Dは、予め定義された分割深さであり、JEMにおいてデフォルトで3に設定される。そして、各サブCUのMVを導出する。

【0122】

【数1】

10

20

30

40

50

$$M = \max\{4, \min\{\frac{M}{2^D}, \frac{N}{2^D}\}\} \quad (1)$$

【0123】

図21に示すように、このバイラテラルマッチングは、2つの異なる参照ピクチャにおける現在のCUの動き軌跡に沿った2つのブロック間の最も近いマッチングを見出すことにより、現在のCUの動き情報を導出するために用いられる。連続した動き軌跡を仮定すると、2つの参照ブロックを指す動きベクトルMV0およびMV1は、現在のピクチャと2つの参照ピクチャとの間の時間的距離、例えばTD0およびTD1に比例する。特殊なケースとしては、現在のピクチャが時間的に2つの参照ピクチャの間にあり、現在のピクチャから2つの参照ピクチャまでの時間的な距離が同じである場合、バイラテラルマッチングはミラーに基づく双方向MVとなる。

【0124】

図22に示すように、現在のピクチャにおけるテンプレート（現在のCUの上側および／または左側の近傍のブロック）と、参照ピクチャにおけるブロック（テンプレートと同じサイズ）との間の最も近いマッチングを見出すことで、テンプレートマッチングを使用して、現在のCUの動き情報を導出する。前述のFRUCマージモード以外に、テンプレートマッチングは、AMVPモードにも適用される。JEMにおいて、HEVCと同様、AMVPは2つの候補を有する。テンプレートマッチング法を用いることで、新しい候補を導出する。テンプレートマッチングによって新規に導出された候補が、第1の既存のAMVP候補と異なる場合、AMVP候補リストの最初に挿入し、次に、リストサイズを2（第2の既存のAMVP候補を取り除くことを意味する）に設定する。AMVPモードに適用される場合、CUレベル検索のみが適用される。

【0125】

2. 3. 3. 1 CUレベルMV候補セット

【0126】

CUレベルのMV候補セットは、以下からなる。

- (i) 現在のCUがAMVPモードになっている場合の元のAMVP候補
- (ii) すべてのマージ候補、
- (iii) 補間MVフィールド内の複数のMV。
- (iv) 上と左の近傍の動きベクトル

【0127】

バイラテラルマッチングを使用する場合、マージ候補の各有効なMVを入力として使用して、バイラテラルマッチングを仮定してMV対を生成する。例えば、マージ候補の1つの有効なMVは、参照リストAにおいて(MVa, refa)である。そして、その対をなすバイラテラルMVの参照ピクチャrefbが他の参照リストBにおいて見出され、refaおよびrefbは、時間的に現在のピクチャの異なる側にある。参照リストBにおいてこのようなrefbが利用可能でない場合、refbをrefaとは異なる参照として決定し、現在のピクチャとの時間的距離はリストBにおける最小値である。refbを決定した後、現在のピクチャとrefa, refbとの時間距離に基づいてMVaをスケールリングすることでMVbを導出する。

【0128】

補間されたMVフィールドからの4つのMVもCUレベル候補リストに追加する。より具体的には、現在のCUの位置(0, 0)、(W/2, 0)、(0, H/2)、(W/2, H/2)の補間MVを加算する。

【0129】

AMVPモードでFRUCを適用する場合、元のAMVP候補をCUレベルMV候補セットにも加える。

【0130】

C Uレベルにおいて、A M V P C Uのための最大15個のM VおよびマージC Uのための最大13個のM Vを候補リストに加える。

【0131】

2.3.3.2 サブC UレベルM V候補セット

【0132】

サブC UレベルのM V候補セットは、以下からなる。

- (i) C Uレベルの検索から決定されたM V、
- (i i) 上、左、左上、右上の近傍のM V、
- (i i i) 照ピクチャからの並置されたM Vのスケーリングされたバージョン、
- (i v) 最大4つのA T M V P 候補、
- (v) 最大4つのS T M V P 候補

10

【0133】

参照ピクチャからのスケーリングされたM Vは、以下のように導出される。両方のリストにおける参照ピクチャをすべてトラバースする。参照ピクチャにおけるサブC Uの配列位置にあるM Vは、開始C UレベルM Vの参照に対してスケーリングされる。

【0134】

A T M V P およびS T M V P の候補は、最初の4つの候補に限定される

【0135】

サブC Uレベルにおいて、最大17個のM Vが候補リストに追加される。

20

【0136】

2.3.3.3 補間M Vフィールドの生成

【0137】

フレームを符号化する前に、一方のM Eに基づいてピクチャ全体に対して補間動きフィールドを生成する。そして、この動きフィールドを後にC UレベルまたはサブC UレベルのM V候補として使用してもよい。

【0138】

まず、両方の参照リストにおける各参照ピクチャの動きフィールドは、 4×4 ブロックレベルでトラバースされる。各 4×4 ブロックにおいて、現在のピクチャ（図23に示す）の 4×4 ブロックを通過するブロックに関連する動きで、補間動きがまだ割り当てられていない場合、時間的距離T D 0およびT D 1に基づいて（H E V CにおけるT M V P のM Vスケーリングと同様に）、参照ブロックの動きを現在のピクチャにスケーリングし、スケーリングされた動きを現在のフレームのブロックに割り当てる。 4×4 ブロックにスケーリングされたM Vが割り当てられていない場合、ブロックの動きは、補間された動きフィールドにおいて利用不可能であるとマークされる。

30

【0139】

2.3.3.4 補間およびマッチングコスト

【0140】

1つの動きベクトルが1つの分数のサンプル位置を指す場合、動きの補償された補間が必要である。複雑性を低減するために、通常の8タップH E V C補間の代わりに、バイラテラルマッチングおよびテンプレートマッチングの両方に双線形補間を使用する。

40

【0141】

マッチングコストの計算は、異なるステップでは少し異なる。C Uレベルの候補セットから候補を選択する場合、マッチングコストは、バイラテラルマッチングまたはテンプレートマッチングの差分の絶対値の和（S A D）である。開始M Vを決定した後、サブC Uレベル検索におけるバイラテラルマッチングのマッチングコストCを以下のように算出する。

【0142】

【数2】

50

$$C = SAD + w \cdot (|MV_x - MV_x^s| + |MV_y - MV_y^s|) \quad (2)$$

【0143】

ここで、 w は、経験的に4に設定された重み係数であり、 MV および MV^s は、それぞれ、現在の MV および開始 MV を示す。 SAD は、依然として、サブCUレベル検索におけるテンプレートマッチングのマッチングコストとして使用される。

【0144】

FRUCモードにおいて、 MV は、輝度サンプルのみを使用することによって導出される。導出された動きは、MCインター予測のために、輝度および彩度の両方に使用される。 MV を決定した後、輝度用の8タップ補間フィルタおよび彩度用の4タップ補間フィルタを使用して、最終的なMCを行う。

【0145】

2.3.3.5 MV の改良

【0146】

MV 改良は、バイラテラルマッチングコストまたはテンプレートマッチングコストの基準を有するパターンに基づく MV 検索である。JEMでは、2つの検索パターン、即ち、UCBDS (Unrestricted Center-Biased Diamond Search) およびCUレベルおよびサブCUレベルでの MV 改良のための適応クロス検索をそれぞれサポートする。CUおよびサブCUレベルの MV 改善の両方のために、 MV は、 $1/4$ 輝度サンプル MV の正確度で直接検索され、これに続いて $1/8$ 輝度サンプル MV の改良が行われる。CUおよびサブCUステップのための MV 改良の検索範囲は、8つの輝度サンプルに等しく設定される。

【0147】

2.3.3.6 テンプレートマッチングFRUCマージモードにおける予測方向の選択

【0148】

バイラテラルマッチングマージモードにおいては、2つの異なる参照ピクチャにおける現在のCUの動き軌跡に沿った2つのブロック間の最も近いマッチングに基づいて、CUの動き情報を導出するため、双方向予測が常に適用される。テンプレートマッチングマージモードについては、そのような制限はない。テンプレートマッチングマージモードにおいて、エンコーダは、 $list0$ からの単一予測、 $list1$ からの単一予測、またはCUのための双方向予測のうちから選択することができる。選択は、テンプレートマッチングコストに基づいて、以下のように行う。

$costBi \leq factor * \min(cost0, cost1)$ の場合

双方向予測を用いる。

それ以外の場合において、 $cost0 \leq cost1$ の場合

$list0$ からの単一予測を用いる。

そうでない場合、

$list1$ からの単一予測を用いる。

【0149】

ここで、 $cost0$ は $list0$ テンプレートマッチングの SAD であり、 $cost1$ は $list1$ テンプレートマッチングの SAD であり、 $costBi$ は双方向予測テンプレートマッチングの SAD である。 $factor$ の値が1.25である場合、選択処理が双方向予測に偏っていることを意味する。このインター予測方向選択は、CUレベルのテンプレートマッチング処理にのみ適用される。

【0150】

2.3.4 デコーダ側動きベクトル改良

【0151】

双方向予測演算において、1つのブロック領域を予測するために、 $list0$ の動きベ

10

20

30

40

50

クトル (MV) および list 1 の MV をそれぞれ使用して構成される 2 つの予測ブロックを組み合わせ、1 つの予測信号を形成する。DMVR (Decoder-side Motion Vector Refinement) 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、双方向予測の 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の歪みに基づく検索を行う。

【0152】

DMVR において、図 23 に示すように、list 0 の最初の MV0 と list 1 の MV1 とから、それぞれ 2 つの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、図 24 に示すような MV0' および MV1' を使用して、最終的な双予測結果を生成する。差分の絶対値の和 (SAD) をコスト尺度として使用する。

【0153】

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 との間の双方向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。

【0154】

2.3.5 バイラテラルマッチングの改良を伴うマージ／スキップモード [8]

【0155】

まず、利用可能な候補の数が最大候補サイズ 19 に達するまで、空間的に近傍のブロックおよび時間的に近傍のブロックの動きベクトルおよび参照インデックスを冗長性チェック付き候補リストに挿入することで、マージ候補リストを構築する。マージ／スキップモードのマージ候補リストは、予め規定された挿入順に基づいて、HEVC (結合候補およびゼロ候補) に用いられる空間的候補 (図 11)、時間的候補、アフィン候補、ATMVP (Advanced Temporal MVP) 候補、STMVP (Spatial Temporal MVP) 候補、および追加候補を挿入することで構築される。

【0156】

－ブロック 1～4 の空間的候補

【0157】

－ブロック 1～4 の外挿アフィン候補

【0158】

－ATMVP

【0159】

－STMVP

【0160】

－仮想アフィン候補

【0161】

－空間的候補 (ブロック 5) (利用可能な候補の数が 6 よりも少ない場合にのみ使用される)。

【0162】

－外挿アフィン候補 (ブロック 5)

【0163】

10

20

30

40

50

－時間的候補（H E V Cのように導出）

【0164】

－外挿アフィン変換候補に続く非隣接空間的候補の後にアフィン候補を外挿する（図25に示すブロック6～49）。

【0165】

－結合候補

【0166】

－ゼロ候補

【0167】

なお、I Cフラグは、S T M V Pおよびアフィンを除き、マージ候補から継承される。また、最初の4つの空間的候補について、双方向予測のものを単一予測のものの前に挿入する。

【0168】

いくつかの実装形態において、現在のブロックに接続されていないブロックにアクセスすることができる。非隣接ブロックが非イントラモードにて符号化されている場合、関連する動き情報を追加のマージ候補として追加してもよい。

【0169】

3. 本明細書に開示される実施形態が解決しようとする課題の例

【0170】

現在のH E V C設計は、動き情報をよりよく符号化するために、現在のブロックの近傍のブロック（現在のブロックの隣）の相関をとることができる。しかしながら、近傍のブロックが、異なる動き軌跡を有する異なる対象に対応する可能性がある。この場合、その近傍のブロックからの予測は効率的ではない。

【0171】

非隣接ブロックの動き情報からの予測は、全ての動き情報（一般的には4×4レベル）をキャッシュに記憶するコストをかけることになり、付加的な符号化利得をもたらし、ハードウェア実装の複雑性を大幅に増大させる。

【0172】

4. いくつかの例

【0173】

既存の実装形態の欠点を克服するために、様々な実施形態において、ブロックの動き情報を予測するために、少なくとも1つの動き候補が記憶された1つ以上のテーブル（例えばルックアップテーブル）を使用するL U Tに基づく動きベクトル予測技術を実装し、より高い符号化効率を有する映像符号化を提供することができる。ルックアップテーブルは、ブロックの動き情報を予測するために動き候補を含める際に使用できるテーブルの一例であり、他の実装形態も可能である。各L U Tは、それぞれが対応する動き情報に関連付けられた1つ以上の動き候補を含んでもよい。動き候補の動き情報は、予測方向、参照インデックス／ピクチャ、動きベクトル、L I Cフラグ、アフィンフラグ、M V D（Motion Vector Derivation）精度、および／またはM V D値の一部または全部を含んでもよい。動き情報は、動き情報がどこに由来しているかを示すために、ブロック位置情報をさらに含んでもよい。

【0174】

開示される技術に基づいたL U Tに基づく動きベクトル予測は、既存のおよび将来の映像符号化規格の両方を向上させることができ、様々な実施形態のために以下の例で説明される。L U Tは、履歴データ（例えば、既に処理されたブロック）に基づいて符号化／復合化処理を行うことを可能にするため、L U Tに基づく動きベクトル予測は、H M V P（History-based Motion Vector Prediction）法と呼ぶこともできる。L U Tに基づく動きベクトル予測方法において、以前に符号化ブロックからの動き情報を有する1つまたは複数のテーブルは、符号化／復号化処理の間、維持される。L U Tに記憶されたこれらの動き候補をH M V P候補と称する。1つのブロック

10

20

30

40

50

の符号化／復号化の間、LUTにおける関連付けられた動き情報を動き候補リスト（例えば、マージ／AMVP候補リスト）に追加して、1つのブロックを符号化／復号化した後に、LUTを使用してもよい。更新されたLUTは、その後、後続のブロックを符号化するために用いられる。つまり、LUTにおける動き候補の更新は、ブロックの符号化／復号化の順に基づく。

【0175】

以下の実施例は、一般的な概念を説明するための例であると考えられるべきである。これらの例は狭い意味で解釈されるべきではない。さらに、これらの例は、任意の方法で組み合わせることができる。

【0176】

一部の実施形態において、1つのブロックの動き情報を予測するために、少なくとも1つの動き候補が記憶された1つ以上のルックアップテーブルを用いてもよい。実施形態は、動き候補を用いて、ルックアップテーブルに記憶された動き情報のセットを示すことができる。従来のAMVPまたはマージモードの場合、実施形態では、動き情報を記憶するためにAMVPまたはマージ候補を使用してもよい。

【0177】

以下の実施例は、一般的な概念を説明する。

【0178】

ルックアップテーブルの例

【0179】

例A1：各ルックアップテーブルは、各候補がその動き情報に関連付けられた1つ以上の動き候補を含んでもよい。

a．動き候補の動き情報は、予測方向、参照インデックス／ピクチャ、動きベクトル、LICフラグ、アフィンフラグ、MVD精度、MVD値の一部または全部を含んでもよい。

b．動き情報は、動き情報がどこに由来しているかを示すために、ブロック位置情報および／またはブロック形状をさらに含んでもよい。

【0180】

LUTの選択

【0181】

例B1：1つのブロックを符号化する場合、1つのルックアップテーブルからの動き候補の一部または全部を順にチェックすることができる。1つのブロックを符号化する間に1つの動き候補をチェックする際に、この動き候補を動き候補リスト（例えば、AMVP、マージ候補リスト）に加えてもよい。

【0182】

例B2：ルックアップテーブルの選択は、ブロックの位置に依存してもよい。

【0183】

ルックアップテーブルの使用方法

【0184】

例C1：チェック対象のルックアップテーブルにおける動き候補の総数は、予め規定されてもよい。

【0185】

例C2：1つのルックアップテーブルに含まれる1つ以上の動き候補は、1つのブロックによって直接継承されてもよい。

a．それらをマージモード符号化に使用してもよい。すなわち、マージ候補リスト導出処理において動き候補をチェックしてもよい。

b．これらは、アフィンマージモード符号化に使用してもよい。

i．アフィンフラグが1である場合、ルックアップテーブルにおける動き候補をアフィンマージ候補として加えることができる。

c．以下の場合、ルックアップテーブルにおける動き候補のチェックを有効にしてもよい。

10

20

30

40

50

i. TMVP候補を挿入した後、マージ候補リストが満杯になっていない。

ii. 空間的マージ候補導出のために特定の空間的な隣接ブロックをチェックした後、マージ候補リストが満杯になっていない。

iii. すべての空間的マージ候補の後、マージ候補リストが満杯になっていない。

iv. 結合双方向予測マージ候補の後、マージ候補リストが満杯になっていない。

v. 他の符号化方式（例えば、HEVCデザインまたはJEMデザインのマージ導出処理）からマージ候補リストに入れられた空間的または時間的（例えば、隣接空間および非隣接空間、TMVP、STMVP、ATMVPなどを含む）マージ候補の数が、最大許容されるマージ候補から、所与の閾値を引いた数よりも少ない場合。

1. 一例において、閾値は、1または0に設定される。

2. 代替的に、閾値は、SPS/PPS/シーケンス、ピクチャ、スライスヘッダ/タイルにおいて信号通知されてもよく、または予め規定されてもよい。

3. 代替的に、閾値は、ブロックごとに適応的に変更されてもよい。例えば、それは、ブロックサイズ/ブロック形状/スライスタイプのような符号化されたブロック情報に依存してもよく、および/または利用可能な空間的または時間的マージ候補の数に依存してもよい。

4. 他の例において、既にマージ候補リストに追加されたある種のマージ候補の数が、最大許容マージ候補から、所与の閾値を引いた数未満である場合。「ある種のマージ候補」は、HEVCのような空間的候補であってもよいし、隣接しないマージ候補であってもよい。

vi. マージ候補リストに動き候補を追加する前に、ブルーニングを適用してもよい。本特許明細書に開示されたこの例および他の例の様々な実装形態において、ブルーニングは、a) 動き情報と既存のエントリとを一意性のために比較すること、またはb) 一意である場合、動き情報をリストに追加すること、またはc) 一意でない場合、c1) 動き情報を追加しない、またはc2) 動き候補を追加し、一致した既存のエントリを削除することを含んでもよい。

1. 一例において、動き候補は、マージ候補リストの他の符号化方法から利用可能な空間的または時間的（例えば、隣接空間および非隣接の空間、TMVP、STMVP、ATMVP等を含む）マージ候補の全部または一部にブルーニングされてもよい。

2. 動き候補は、サブブロックに基づく動き候補、例えば、ATMVP、STMVPにブルーニングされなくてもよい。

3. 一例において、現在の動き候補は、マージ候補リストにおける利用可能な動き候補（現在の動き候補の前に挿入された）の全部または一部にブルーニングされてもよい。

4. 動き候補に関連するブルーニング工程の数（例えば、動き候補をマージリストにおける他の候補と比較する必要がある回数）は、利用可能な空間的または時間的マージ候補の数に依存してもよい。例えば、新しい動き候補をチェックする際に、マージリストに利用可能な候補がM個ある場合、新しい動き候補を最初のK個（ $K \leq M$ ）の候補とのみ比較してもよい。ブルーニング関数が偽を返す（例えば、最初のK個の候補のいずれとも同一でない）場合、この新しい動き候補は、M個の候補のすべてと異なると見なされ、マージ候補リストに追加され得る。

5. 一例において、新しく付加された動き候補とマージ候補リストにおける第1のN個の候補とを比較するだけである。例えば、 $N = 3$ 、4または5である。Nは、エンコーダからデコーダに信号通知されてもよい。

6. 一例において、チェック対象の新しい動き候補は、マージ候補リストにおける最後のN個の候補と比較されるのみである。例えば、 $N = 3$ 、4または5である。Nは、エンコーダからデコーダに信号通知されてもよい。

7. 一例において、以前リストに追加された候補を選択し、テーブルからの新しい動き候補と比較する方法は、前回追加された候補がどこから導出されたかに依存してもよい。

a. 一例において、ルックアップテーブルにおける動き候補を、所与の時間的お

10

20

30

40

50

よび／または空間的に隣接ブロックから導出された候補と比較してもよい。

b. 一例において、ルックアップテーブルにおける動き候補の異なるエントリを、以前追加された異なる候補と比較してもよい（すなわち、異なる位置から導出された）。
【0186】

例C3：ルックアップテーブルに含まれる動き候補は、ブロックの動き情報を符号化する予測モジュールとして用いられてもよい使用してもよい。

a. それらをAMVPモード符号化に使用してもよい。すなわち、AMVP候補リスト導出処理において動き候補をチェックしてもよい。

b. 以下の場合、ルックアップテーブルにおける動き候補のチェックを有効にしてもよい。

i. TMVP候補を挿入した後、AMVP候補リストが満杯になっていない。

ii. AMVP候補リストは、空間的近傍から選択し、プルーニングした後で、TMVP候補を挿入する直前には、満杯になっていない。

iii. 上側の隣接ブロックからのAMVP候補がスケーリング無しで存在しない場合、および／または、左側の隣接ブロックからのAMVP候補がスケーリング無しで存在しない場合。

iv. AMVP候補リストに動き候補を追加する前に、プルーニングを適用してもよい。

v. 例C2の一部で述べたのと同様の規則をAMVPモードに適用してもよい。

c. 現在の参照ピクチャと同一の参照ピクチャを有する動き候補をチェックする。

i. 代替的に、現在の参照ピクチャとは異なる参照ピクチャを有する動き候補も（スケーリングされたMVと併せて）チェックする。

ii. 代替的に、まず、現在の参照ピクチャと同一の参照ピクチャを有するすべての動き候補をチェックし、次に、現在の参照ピクチャとは異なる参照ピクチャを有する動き候補をチェックする。

iii. 代替的に、マージでも同様に動き候補をチェックする。

【0187】

例C4：ルックアップテーブルにおける動き候補のチェック順序は、以下のように規定される（ K ($K \geq 1$) 個の動き候補をチェックすることができるとする）：

a. ルックアップテーブルにおける最後の K 個の動き候補は、例えば、LUTへのエントリインデックスの降順に配列される。

b. 最初の $K \% L$ 候補。 L は、 $K \geq L$ である場合のルックアップテーブルのサイズであり、例えば、LUTへのエントリインデックスの降順に配列される。

c. $K \geq L$ である場合、順番に基づいた、ルックアップテーブルにおけるすべての候補（ L 個の候補）。一例において、テーブルにおける最初の $K \% L$ 個の候補を、LUTへのエントリインデックスの降順にチェックし、次に、エントリインデックスの降順に最後の（ $L - K \% L$ ）個の候補をチェックする。

d. 代替的に、さらに、動き候補のインデックスの降順（すなわち、LUTへのエントリインデックス）に基づいてもよい。

e. 代替的に、動き候補のインデックスの昇順に基づいてもよい。

f. 代替的に、候補情報に基づいて、動き候補に関連付けられた位置の距離や現在のブロックなど、 K 個の動き候補を選択する。

i. 一例において、 K 個の最も近い動き候補を選択する。

ii. 一例において、候補情報は、距離を算出する際に、ブロックの形状をさらに考慮してもよい。

g. 一例において、 L 個の候補を含むテーブルからの K 個の動き候補のチェック順序は、次のように規定されてもよい： a_0 、 $a_0 + T_0$ 、 $a_0 + T_0 + T_1$ 、 $a_0 + T_0 + T_1 + T_2$ 、...、 $a_0 + T_0 + T_1 + T_2 + \dots + T_{K-1}$ のインデックスを有する候補を順に選択し、 a_0 と T_i （ i は $0 \dots K-1$ である）が整数値である。

i. 一例において、 a_0 は、0（すなわち、テーブルにおける動き候補の最初のエン

10

20

30

40

50

トリ)に設定される。代替的に、 a_0 は $(K - L / K)$ に設定される。演算「 $\div$ 」は、結果をゼロに切り捨てる整数除算として規定される。代替的に、 a_0 は、0と L / K との間の任意の整数に設定される。

1. 代替的に、 a_0 の値は、現在のブロックおよび隣接ブロックの符号化情報に依存してもよい。

i i. 一例において、すべての間隔 T_i (i は $0 \cdots K - 1$ である)は同じであり、例えば L / K である。演算「 $\div$ 」は、結果をゼロに切り捨てる整数除算として規定される。

i i i. 一例において、 (K, L, a_0, T_i) は、 $(4, 16, 0, 4)$ 、または $(4, 12, 0, 3)$ 、または $(4, 8, 0, 1)$ 、または $(4, 16, 3, 4)$ 、または $(4, 12, 2, 3)$ 、または $(4, 8, 1, 2)$ に設定される。 T_i はすべての i について同じである。

10

i v. このような方法は、 K が L よりも小さい場合にのみ適用されてもよい。

v. 代替的に、さらに、 K が閾値以上である場合、例C4のc部を適用してもよい。閾値は、 L として規定されてもよく、または K に依存してもよく、またはブロックごとに適応的に変更されてもよい。一例において、閾値は、ルックアップテーブルから新しい動き候補を追加する前のリストにおける利用可能な動き候補の数に依存してもよい。

h. 一例において、 L 個の候補を含むテーブルからの K 個の動き候補のチェック順序は、次のように規定されてもよい： a_0 、 $a_0 - T_0$ 、 $a_0 - T_0 + T_1$ 、 $a_0 - T_0 - T_1 - T_2$ 、 $\dots$ 、 $a_0 - T_0 - T_1 - T_2 - \dots - T_{K-1}$ のインデックスを有する候補を順に選択し、 a_0 と T_i (i は $0 \dots K - 1$ である)が整数値である。

20

i. 一例において、 a_0 は、 $L - 1$ に設定される (すなわち、テーブルにおける動き候補の最後のエントリ)。代替的に、 a_0 は、 $L - 1 - L / K$ と $L - 1$ との間の任意の整数に設定される。

i i. 一例において、すべての間隔 T_i (i は $0 \cdots K - 1$ である)は同じであり、例えば L / K である。

i i i. 一例において、 (K, L, a_0, T_i) は、 $(4, 16, L - 1, 4)$ 、または $(4, 12, L - 1, 3)$ 、または $(4, 8, L - 1, 1)$ 、または $(4, 16, L - 4, 4)$ 、または $(4, 12, L - 3, 3)$ 、または $(4, 8, L - 2, 2)$ に設定される。 T_i はすべての i について同じである。

i v. このような方法は、 K が L よりも小さい場合にのみ適用されてもよい。

30

代替的に、さらに、 K が閾値以上である場合、例C4のc部を適用してもよい。閾値は、 L として規定されてもよく、または K に依存してもよく、またはブロックごとに適応的に変更されてもよい。一例において、閾値は、ルックアップテーブルから新しい動き候補を追加する前のリストにおける利用可能な動き候補の数に依存してもよい。

i. ルックテーブルから動き候補を選択する数および／または方法は、符号化情報、例えばブロックサイズ／ブロック形状に依存してもよい。

i. 一例において、より小さいブロックサイズの場合、最後の K 個の動き候補を選択する代わりに、(最後から始まらない)他の K 個の動き候補を選択してもよい。

i i. 一例において、符号化情報は、AMVPモードであってもよいし、マージモードであってもよい。

40

i i i. 一例において、符号化情報は、アフィンモードまたは非アフィンAMVPモードまたは非アフィンマージモードであってもよい。

i v. 一例において、符号化情報は、アフィンAMVP (インター)モード、または、アフィンマージモード、または、非アフィンAMVPモード、または、非アフィンマージモードであってもよい。

v. 一例において、符号化情報は、CPR (Current Picture Reference)モードであってもよいし、CPRモードでなくてもよい。

v i. 代替的に、ルックアップテーブルから動き候補を選択する方法は、ルックアップテーブルにおける動き候補の数、および／または、ルックアップテーブルから新しい動き候補を追加する前のリストにおける利用可能な動き候補の数にさらに依存してもよい。

50

j. 一例において、チェック対象のルックアップテーブル内の利用可能な動き候補の最大数（すなわち、マージ／AMVP候補リストに追加されてもよい）は、ルックアップテーブル内の利用可能な動き候補の数（ $N_{avaiMCinLUT}$ によって示される）、および／または、ルックアップテーブルからの追加対象の最大許容動き候補の数（ NUM_{maxMC} によって示される）（これは、予め規定されていてもよいし、信号通知されていてもよい）、および／または、ルックアップテーブルから候補をチェックする前のリスト内の利用可能な動き候補の数（ N_{avaiC} によって示される）に依存していてもよい。

i. 一例において、チェック対象のルックアップテーブルにおける動き候補の最大数は、（ $N_{avaiMCinLUT}$ 、 NUM_{maxMC} 、 N_{avaiC} ）の最小値に設定される。

ii. 代替的に、チェック対象のルックアップテーブルにおける動き候補の最大数は、（ $N_{avaiMCinLUT}$ 、 $NUM_{maxMC} - N_{avaiC}$ ）の最小値に設定される。

iii. 一例において、 N_{avaiC} は、空間的または時間的（隣接および／または非隣接）な近傍のブロックから導出された挿入候補の数を示す。代替的に、サブブロック候補（AMTVP、STMVPなど）の数は、 N_{avaiC} に含まれていない。

iv. NUM_{maxMC} は、符号化モードに依存してもよく、例えば、マージモードおよびAMVPモードにおいて、 NUM_{maxMC} は異なる値に設定されてもよい。一例において、マージモードの場合、 NUM_{maxMC} は、4、6、8、10等に設定してもよく、AMVPモードの場合、 NUM_{maxMC} は、1、2、4などに設定してもよい。

v. 代替的に、 NUM_{maxMC} は、ブロックサイズ、ブロック形状、スライスタイプ等のような他の符号化情報に依存してもよい。

k. 異なるルックアップテーブルのチェック順序は、次のサブセクションのルックアップテーブルの使用法で規定されている。

l. 一旦、マージ／AMVP候補リストが最大許容候補数に達すると、このチェック処理は終了する。

m. 一旦、マージ／AMVP候補リストが、最大許容候補数から閾値（ Th ）を減算した値に達すると、このチェック処理は終了する。一例において、 Th は、例えば、1、2、または3の正の整数値として予め規定されてもよい。代替的に、 Th は、ブロックごとに適応的に変更されてもよい。代替的に、 Th は、SPS／PPS／スライスヘッダ等において信号通知されてもよい。代替的に、 Th は、さらに、ブロック形状／ブロックサイズ／符号化モードなどに依存してよい。代替的に、 Th は、LUTからの動き候補を追加する前の利用可能な候補の数に依存してもよい。

n. 代替的に、一旦、追加された動き候補の数が最大許容動き候補数に達すると、これは終了する。最大許容動き候補数は、信号通知されてもよく、または予め規定されてもよい。代替的に、最大許容動き候補数は、ブロック形状／ブロックサイズ／符号化モード等にさらに依存してもよい。

o. テーブルサイズを示す1つのシンタックス要素ならびにチェック対象の動き候補の数（すなわち、 $K=L$ ）は、SPS、PPS、スライスヘッダ、タイルヘッダにおいて信号通知してもよい。

【0188】

例C5：1つのブロックの動き情報符号化のためのルックアップテーブルの使用の有効／無効は、SPS、PPS、スライスヘッダ、タイルヘッダ、CTU、CTB、CU、またはPU、複数のCTU／CTB／CU／PUを含む領域において信号通知されてもよい。

【0189】

例C6：ルックアップテーブルからの予測を適用するかどうかは、さらに符号化情報に依存してもよい。1つのブロックに適用しないと推測される場合、予測の指示の追加の信号通知はスキップされる。代替的に、1つのブロックに適用しないと推測される場合、ルックアップテーブルの動き候補にアクセスする必要はなく、関連する動き候補のチェックは省略される。

【0190】

例C7：以前符号化されたフレーム／スライス／タイルにおけるルックアップテーブル

の動き候補を使用して、異なるフレーム／スライス／タイルにおけるブロックの動き情報を予測してもよい。

【0191】

ルックアップテーブルの更新

【0192】

例D1：動き情報を有するブロックを符号化した後（すなわち、IntraBCモード、インター符号化モード）に、1つ以上のルックアップテーブルを更新してもよい。

【0193】

追加の例示的な実施形態

【0194】

以前符号化されたブロックの動き情報としてHMVP候補を規定する、HMVP（History-based MVP）方法が提案される。符号化／復号化処理中、複数のHMVP候補を有するテーブルが維持される。新しいスライスに遭遇した場合、テーブルは空になる。インター符号化ブロックがあるときはいつでも、関連する動き情報を新しいHMVP候補としてテーブルの最後のエントリに加える。全体の符号化フローを図30に示す。

【0195】

一例において、テーブルサイズはL（例えば、L=16または6、または44）に設定され、これは、L個までのHMVP候補をテーブルに追加することができることを示す。

【0196】

1つの実施形態（例11.g.iに対応する）において、以前符号化されたブロックからのHMVP候補がL個よりも多く存在する場合、テーブルが常に最新の以前符号化されたL個の動き候補を含むように、先入れ先出し（FIFO：First-In-First-Out）規則が適用される。図31は、FIFO規則を適用してHMVP候補を除去し、提案される方法で使用するテーブルに新しいものを追加する例を示す。

【0197】

別の実施形態（例11.g.iiiに対応する）において、新しい動き候補を追加するときはいつでも（例えば、現在のブロックがインター符号化され、非アフィンモードであるなど）、まず、冗長性チェック処理を適用し、テーブル（例えば、LUT）に同じまたは類似した動き候補があるかどうかを識別する。

【0198】

いくつかの例を以下に示す。

【0199】

図32Aは、新しい動き候補を追加する前に、テーブルが満杯であった場合の例を示す。

【0200】

図32Bは、新しい動き候補を追加する前に、テーブルが満杯でなかった場合の例を示す。

【0201】

図32Aおよび図32Bは、ともに、冗長性削除に基づくテーブル更新方法（1つの冗長性動き候補を除去する）の例を示す。図32A（上）に示すように、更新前のL個のテーブルエントリがある。冗長性チェックの間、HMVP2は冗長である（例えば、重複している）ことが見出され、排除され、従って、L-1個のエントリ（下）を生成し、これに新しいエントリを追加することができる。図32Bは、同様に、冗長な候補の削除を示す。

【0202】

図33Aおよび図33Bは、冗長性削除に基づくテーブル更新方法（複数の冗長性動き候補を除去する、図では2つの候補を示す）の2つの場合の実装形態を示す。

【0203】

図33Aは、新しい動き候補を追加する前に、テーブルが満杯であった場合の例を示す。

【0204】

図 3 3 B は、新しい動き候補を追加する前に、テーブルが満杯でない場合の例を示す。

【 0 2 0 5 】

H M V P 候補は、マージ候補リスト構築処理において使用され得る。T M V P 候補の後に、テーブルにおける最後のエン트리から最初のエン트리（または最後の K 0 の H M V P、例えば、 $K 0 = 16$ または 6）までのすべての H M V P 候補を挿入する。H M V P 候補に対してプルーニングを適用する。利用可能なマージ候補の総数が信号通知された最大許容マージ候補に達すると、マージ候補リスト構築処理を終了する。代替的に、加算された動き候補の総数が、所与の値に達すると、L U T からの動き候補のフェッチを終了する。

【 0 2 0 6 】

同様に、H M V P 候補は、A M V P 候補リスト構築処理において使用されてもよい。T M V P 候補の後に、テーブルにおける最後の K 1 個の H M V P 候補の動きベクトルを挿入する。A M V P 対象参照ピクチャと同じ参照ピクチャを有する H M V P 候補のみを用いて、A M V P 候補リストを構築する。H M V P 候補に対してプルーニングを適用する。一例において、K 1 は 4 に設定される。

【 0 2 0 7 】

図 2 8 は、映像処理装置 2 8 0 0 のブロック図である。装置 2 8 0 0 は、本明細書に記載の方法の 1 つ以上を実装するために使用してもよい。装置 2 8 0 0 は、スマートフォン、タブレット、コンピュータ、I o T (I n t e r n e t o f T h i n g s) 受信機等により実装されてよい。装置 2 8 0 0 は、1 つ以上のプロセッサ 2 8 0 2 と、1 つ以上のメモリ 2 8 0 4 と、映像処理ハードウェア 2 8 0 6 と、を含んでよい。1 つまたは複数のプロセッサ 2 8 0 2 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。1 または複数のメモリ 2 8 0 4 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してもよい。映像処理ハードウェア 2 8 0 6 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用いられてもよい。

【 0 2 0 8 】

図 2 9 は、映像処理方法 2 9 0 0 の例のフローチャートである。方法 2 9 0 0 は、ステップ 2 9 0 2 でテーブルを維持することを含み、各テーブルは、動き候補のセットを含み、各動き候補は、対応する動き情報に関連付けられる。この方法 2 9 0 0 は、さらにステップ 2 9 0 4 において、少なくとも 1 つ以上のテーブルに基づいて、第 1 の映像ブロックと、第 1 の映像ブロックを含む映像のビットストリーム表現との間の変換を行うことを含み、変換を行うことは、チェック終了判定基準が満たされるまで規則で決定された順にテーブル内の動き候補をチェックすることを含む。

【 0 2 0 9 】

上述した方法／技術の特徴および実施形態を以下に説明する。

【 0 2 1 0 】

1. テーブルを維持することであって、各テーブルは、動き候補のセットを含み、各動き候補は、対応する動き情報に関連付けられる、ことと、少なくとも 1 つ以上のテーブルに基づき、第 1 の映像ブロックと、前記第 1 の映像ブロックを含む映像のビットストリーム表現との間の変換を行うことであって、前記変換を行うことは、チェック終了判定基準が満たされるまで規則で規定された順にテーブル内の動き候補をチェックすることと、を有する、映像処理方法。

【 0 2 1 1 】

2. 前記規則は、前記テーブルにおける最後の K 個の動き候補をチェックすることを指定し、K が 1 以上である、第 1 項に記載の方法。

【 0 2 1 2 】

3. 前記規則は、前記テーブルにおける L を法とする K ($K \% L$) 個の最初の候補がチェックされることを規定しており、K は L 以上であり、L は前記テーブルにおけるエントリの数を示す、第 1 項に記載の方法。

【 0 2 1 3 】

4. 前記規則は、前記テーブルにおけるL個の動き候補をすべてチェックすることを規定しており、Lは前記テーブルにおけるエントリの数を示す、第1項に記載の方法。

【0214】

5. 前記規則は、動き候補インデックスの降順に基づいて、前記テーブルにおける前記動き候補をチェックすることを規定する、第1項に記載の方法。

【0215】

6. 前記規則は、動き候補インデックスの昇順に基づいて、前記テーブルにおける前記動き候補をチェックすることを規定する、第1項に記載の方法。

【0216】

7. 前記規則は、候補情報に基づいて、K個の動き候補を選択することを特定し、Kが1以上である、第1項に記載の方法。

【0217】

8. 前記候補情報は、前記第1の映像ブロックと前記動き候補に関連付けられた位置との間の距離を含む、第7項に記載の方法。

【0218】

9. K個の最も近い動き候補を選択する、第7項に記載の方法。

【0219】

10. 前記候補情報は、距離を算出するように考慮されたブロック形状を含む、第8項に記載の方法。

【0220】

11. 前記K個の動き候補の選択は、前記第1の映像ブロックの符号化情報に依存する、第7項に記載の方法。

【0221】

12. 前記符号化情報は、ブロックサイズまたはブロック形状のうちの少なくとも1つを含む、第11項に記載の方法。

【0222】

13. 前記第1の映像ブロックの前記符号化情報は、AMVPモード、マージモード、アフィンモード、非アフィンAMVPモード、非アフィンマージモード、アフィンAMVP（インター）モード、アフィンマージモード、非アフィンAMVPモード、非アフィンマージモード、CPR（Current Picture Reference）モード、または非CPRモードのうちの少なくとも1つを示す、第11項に記載の方法。

【0223】

14. 前記K個の動き候補の前記選択は、i) 前記テーブルにおける複数の動き候補の数、またはii) 前記テーブルから新しい動き候補を追加する前の、動き候補リストにおける利用可能な候補の数のうちの少なくとも1つに依存する、第7項に記載の方法。

【0224】

15. より小さいブロックサイズの場合、前記選択されたK個の動き候補は、テーブルにおける最後のK個の動き候補に対応しない、第11項に記載の方法。

【0225】

16. 別のテーブルにおける動き候補は、別の順にチェックされる、第1項に記載の方法。

【0226】

17. マージまたはAMVP（Advanced Motion Vector Prediction）候補リストのうち少なくとも1つを含む動き候補リストが、最大許容候補数に達した場合、前記チェック終了判定基準を満たす、第1～16項のいずれか1項に記載の方法。

【0227】

18. チェック対象のテーブルから追加された動き候補の数が最大許容動き候補数に達した場合に、チェック終了判定基準を満たす、第1～16項のいずれか1項に記載の方法。

【0228】

10

20

30

40

50

19. チェック対象の前記最大許容動き候補数は、予め規定されるか、または信号通知される、第18項に記載の方法。

【0229】

20. チェック対象のテーブルからの前記最大許容動き候補数は、ブロック形状、ブロックサイズ、または符号化モードのうち少なくとも1つに依存する、第18項に記載の方法。

【0230】

21. i) テーブルのサイズ、または、ii) チェック対象のテーブルからの動き候補の数のうちの少なくとも1つを信号通知することをさらに含む、第1項に記載の方法。

【0231】

22. 前記候補リストにおける候補の数が、前記候補リストの最大許容候補数から閾値 (Th) を減算した値に達した場合、前記チェック終了判定基準を満たす、第1～16項のいずれか1項に記載の方法。

【0232】

23. 前記閾値 (Th) は、正の整数として予め規定される、第22項に記載の方法。

【0233】

24. ブロックによって前記閾値 (Th) が変化する、第22項に記載の方法。

【0234】

25. 前記閾値 (Th) は、VPS (Video Parameter Set)、SPS (Sequence Parameter Set)、PPS (Picture Parameter Set)、スライスヘッダ、またはタイルヘッダのうちの1つ以上において信号通知される、第22項に記載の方法。

【0235】

26. 前記閾値 (Th) は、ブロック形状、ブロックサイズ、または符号化モードのうちの少なくとも1つに依存する、第22項に記載の方法。

【0236】

27. 前記閾値 (Th) は、前記テーブルから動き候補を追加する前の利用可能な候補の数に依存する、第22項に記載の方法。

【0237】

28. 前記チェック終了判定基準は、テーブルにおける利用可能な動き候補の数、テーブルからの追加対象の最大許容動き候補、およびテーブルからの前記動き候補のチェック前における動き候補リストにおける利用可能な候補の数のうちの少なくとも1つに依存する、第1～16項のいずれか1項に記載の方法。

【0238】

29. 前記順序は、 a_0 、 $a_0 + T_0$ 、 $a_0 + T_0 + T_1$ 、 $a_0 + T_0 + T_1 + T_2$ 、 $\dots$ 、 $a_0 + T_0 + T_0 + T_1 + T_2 + \dots + T_{K-1}$ と等しいインデックスを有する K 個の動き候補を順に選択することによる、L 個の候補を含む前記テーブルからの K 個の動き候補のチェック順序を含み、 a_0 と T_i (i は $0 \dots K-1$ である) は整数値である、第1項に記載の方法。

【0239】

30. 前記テーブルにおける動き候補の最初のエントリとして、 a_0 を 0 に設定する、第29項に記載の方法。

【0240】

31. a_0 を $(K - L / K)$ に設定し、演算子「 $\div$ 」は、結果をゼロに切り捨てる整数除算として定義される、第29項に記載の方法。

【0241】

32. a_0 を、0 と L / K との間の任意の整数に設定し、演算子「 $\div$ 」は、結果をゼロに切り捨てる整数除算として定義される、第29項に記載の方法。

【0242】

33. a_0 は、前記第1の映像ブロックおよび近傍のブロックの符号化情報に依存する

10

20

30

40

50

、第 29 項に記載の方法。

【0243】

34. T_i (i は 0 と $K-1$ との間の整数である) が同じ値を有する、第 29 項に記載の方法。

【0244】

35. 前記同じ値は L/K であり、演算子「 $\div$ 」は、結果をゼロに切り捨てる整数除算として定義される、第 34 項に記載の方法。

【0245】

36. (K, L, a_0, T_i) は、 $(4, 16, 0, 4)$ 、 $(4, 12, 0, 3)$ 、 $(4, 8, 0, 1)$ 、 $(4, 16, 3, 4)$ 、 $(4, 12, 2, 3)$ または $(4, 8, 1, 2)$ のいずれか 1 つに設定される第 29 項に記載の方法。

10

【0246】

37. 前記チェック順序は、 K が L よりも小さい場合に適用される、第 29 項に記載の方法。

【0247】

38. 前記規則は、 K が閾値以上である場合、前記テーブルにおける L 個の動き候補のすべてをチェックすることを規定する、第 29 項に記載の方法。

【0248】

39. 前記閾値は L である、第 38 項に記載の方法。

【0249】

40. 前記閾値は K に依存するか、またはブロックによって変化する、第 38 項に記載の方法。

20

【0250】

41. 前記順序は、 a_0 、 $a_0 - T_0$ 、 $a_0 - T_0 - T_1$ 、 $a_0 - T_0 - T_1 - T_2$ 、 $\dots$ 、 $a_0 - T_0 - T_0 - T_1 - T_2 - \dots - T_{K-1}$ と等しいインデックスを有する K 個の動き候補を順に選択することによる、 L 個の候補を含むテーブルからの K 個の動き候補のチェック順序を含み、 a_0 と T_i (i は $0 \dots K-1$ である) は整数値である、第 1 項に記載の方法。

【0251】

42. 前記テーブルにおける動き候補の最後のエントリとして、 a_0 を $L-1$ に設定する、第 41 項に記載の方法。

30

【0252】

43. a_0 は、 $L-1 - L/K$ と $L-1$ との間の任意の整数に設定され、演算子「 $\div$ 」は、結果をゼロに切り捨てる整数除算として定義される、第 41 項に記載の方法。

【0253】

44. T_i (i は 0 と $K-1$ との間の整数である) が同じ値を有する、第 41 項に記載の方法。

【0254】

45. 前記同じ値は L/K であり、演算子「 $\div$ 」は、結果をゼロに切り捨てる整数除算として定義される、第 44 項に記載の方法。

40

【0255】

46. (K, L, a_0, T_i) は、 $(4, 16, L-1, 4)$ 、 $(4, 12, L-1, 3)$ 、 $(4, 8, L-1, 1)$ 、 $(4, 16, L-4, 4)$ 、 $(4, 12, L-3, 3)$ 、または $(4, 8, L-2, 2)$ のいずれか 1 つに設定される第 41 項に記載の方法。

【0256】

47. 前記チェック順序は、 K が L よりも小さい場合に適用される、第 41 項に記載の方法。

【0257】

48. 前記規則は、 K が閾値以上である場合、前記テーブルにおける L 個の動き候補のすべてをチェックすることを規定する、第 41 項に記載の方法。

50

【0258】

49. 前記閾値はLである、第48項に記載の方法。

【0259】

50. 前記閾値はKに依存するか、またはブロックによって変化する、第48項に記載の方法。

【0260】

51. 動き候補は、予測方向、参照ピクチャインデックス、動きベクトル値、強度補償フラグ、アフィンフラグ、動きベクトル差精度、または動きベクトル差分値のうちの少なくとも1つを含む動き情報に関連付けられる、第1～50項のいずれか1項に記載の方法。

【0261】

52. 前記チェックされた動き候補をマージ候補リストに追加することをさらに含む、第1～50項のいずれか1項に記載の方法。

【0262】

53. 前記マージ候補リストに追加された前記動き候補のチェック順は、動き候補インデックスの降順に基づく、第52項に記載の方法。

【0263】

54. 前記チェックされた動き候補の前記動きベクトルをAMVP候補リストに追加することをさらに含む、第1～50項のいずれか1項に記載の方法。

【0264】

55. 前記AMVP候補リストに追加された前記動き候補のチェック順は、動き候補インデックスの昇順に基づく、第54項に記載の方法。

【0265】

56. チェック対象の動き候補の数が4以下である、第54項に記載の方法。

【0266】

57. チェック対象の前記テーブルにおける動き候補の最大数は、Nで指標付けされた前記テーブルにおける利用可能な動き候補の数、Mで指標付けされた前記テーブルからの加算対象の最大許容動き候補、またはKで指標付けされた前記テーブルからの前記動き候補をチェックする前の候補リストにおける利用可能な候補の数のうちの、いずれか1つに依存し、N、M、Kは整数である、第1項に記載の方法。

【0267】

58. チェック対象の前記テーブルにおける動き候補の最大数を(N, M, K)の最小値に設定する、第57項に記載の方法。

【0268】

59. チェック対象の前記テーブルにおける動き候補の最大数を(N, M - K)の最小値に設定する、第57項に記載の方法。

【0269】

60. Kは、空間的または時間的に近傍のブロックから導出された、挿入された候補の数を示す、第57項に記載の方法。

【0270】

61. 前記テーブルから前記動き候補をチェックする前の候補リストにおける、K個を数えると考えられる前記利用可能な候補は、AMTVP (Alternative Temporal Motion Vector Prediction) およびSTMVP (Spatial-Temporal Motion Vector Prediction) を含むサブブロック候補を含まない、第57項に記載の方法。

【0271】

62. Mは、前記映像の符号化モードのうちの少なくとも1つに依存する、第57項に記載の方法。

【0272】

63. Mは、マージモードの場合、4、6、8、または10に設定され、Mは、AMVP (Advanced Motion Vector Prediction) モードの場合

10

20

30

40

50

合、1、2、または4に設定される、第62項に記載の方法。

【0273】

64. Mは、前記第1の映像ブロックのブロックサイズ、前記第1の映像ブロックのブロック形状、スライスタイプを含む、前記映像の符号化情報に依存する、第57項に記載の方法。

【0274】

65. 前記テーブルは、ルックアップテーブルに対応する、第1～64項のいずれか1項に記載の方法。

【0275】

66. 前記変換に基づいて、1つ以上のテーブルを更新することをさらに含む、第1～65項のいずれか1項に記載の方法。

【0276】

67. 前記1つ以上のテーブルを更新することは、前記変換を実行した後、前記第1の映像ブロックの動き情報に基づいて1つ以上のテーブルを更新することを含む、第66項に記載の方法。

【0277】

68. 前記更新されたテーブルに基づいて、前記映像の後続の映像ブロックと前記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さらに含む、第67項に記載の方法。

【0278】

69. プロセッサと、命令が記憶された非一時的メモリとを備える装置であって、前記命令は、前記プロセッサによって実行された際に、前記プロセッサに、第1～68項のいずれか1項に記載の方法を実施させる映像システムの装置。

【0279】

70. 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、第1～68項のいずれか1項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。

【0280】

以上、説明の目的で本開示の技術の特定の実施形態を説明したが、本発明の範囲から逸脱することなく様々な修正が可能であることは、理解されるであろう。従って、本開示の技術は、添付の特許請求の範囲による場合を除き、限定されない。

【0281】

本明細書に記載された開示された、およびその他の実施形態、モジュール、および機能操作の実装形態は、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施されてもよく、またはそれらの1つ以上の組み合わせで実施してもよい。開示された、およびその他の実施形態は、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、記憶装置、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、プログラマブルプロセッサ、コンピュータ、または複数のプロセッサ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、プロセッサファームウェア、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ために生成される。

【0282】

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

10

20

30

40

50

、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として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単一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。

10

【0283】

本明細書に記載されたプロセス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実行するための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プロセッサによって行うことができる。プロセスおよびロジックフローはまた、特別目的のロジック回路、例えば、FPGA(Field Programmable Gate Array)またはASIC(Application Specific Integrated Circuit)によって実行することができ、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。

20

【0284】

コンピュータプログラムの実行に適したプロセッサは、例えば、汎用および専用マイクロプロセッサの両方、並びに任意の種類のデジタルコンピュータの任意の1つ以上のプロセッサを含む。一般的に、プロセッサは、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プロセッサと、命令およびデータを記憶するための1つ以上の記憶装置とである。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、および記憶装置を含み、例えば、EPROM、EEPROM、フラッシュ記憶装置、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記憶装置を含む。プロセッサおよびメモリは、専用ロジック回路によって補完されてもよく、または専用ロジック回路に組み込まれてもよい。

30

【0285】

本特許明細書は多くの詳細を含むが、これらは、任意の発明の範囲または特許請求の範囲を限定するものと解釈されるべきではなく、むしろ、特定の発明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特徵は、1つの例において組み合わせて実装してもよい。逆に、単一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、最初にそのように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組み合わせのバリエーションに向けられてもよい。

40

【0286】

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実行されるこ

50

と、または示された全ての操作が実行さ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の構成要素の分離は、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。

【 0 2 8 7 】

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

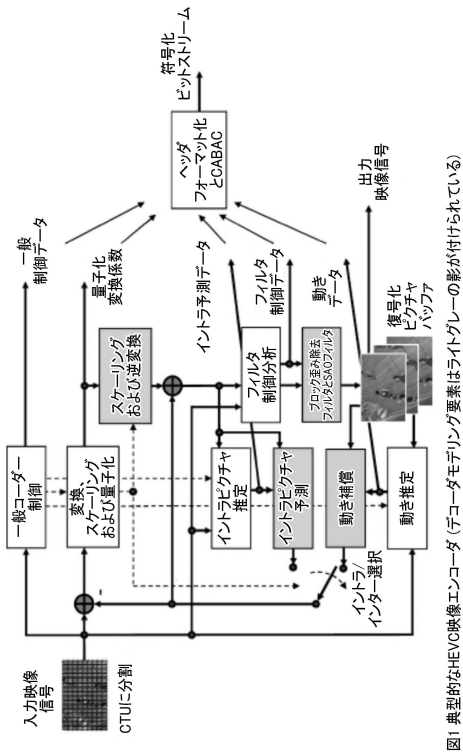
30

40

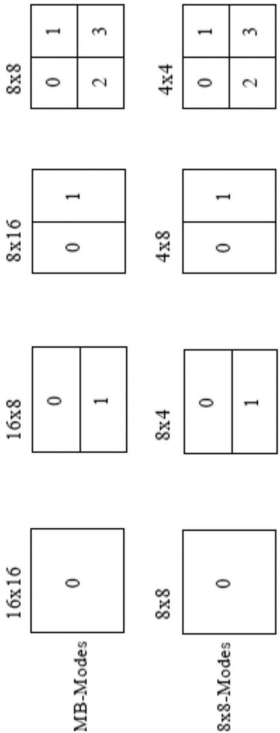
50

【図面】

【図 1】



【図 2】



【図 3】

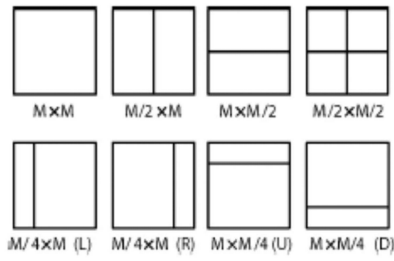
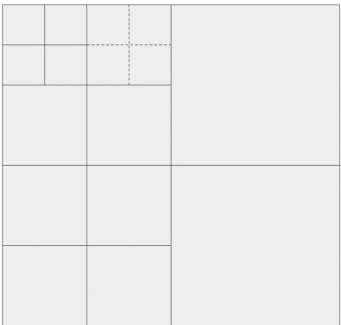
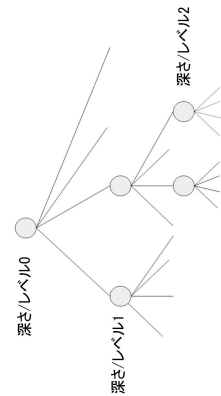
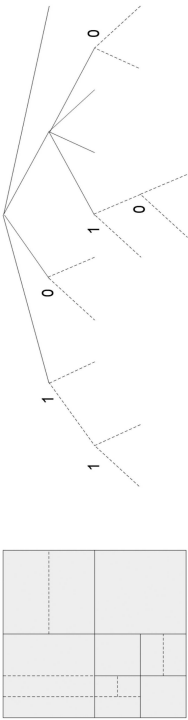


図3 特定のサイズ制限に従って、CBをPBに分割するためのモード。イントラピクチャ予測CBでは、M x MとM/2 x M/2だけがサポートされる

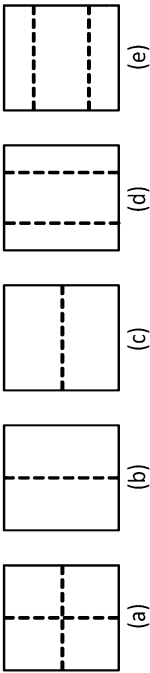
【図 4】



【図 5】



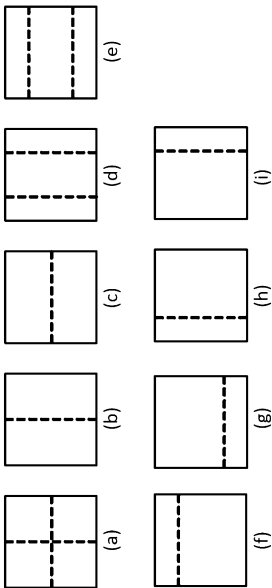
【図 6】



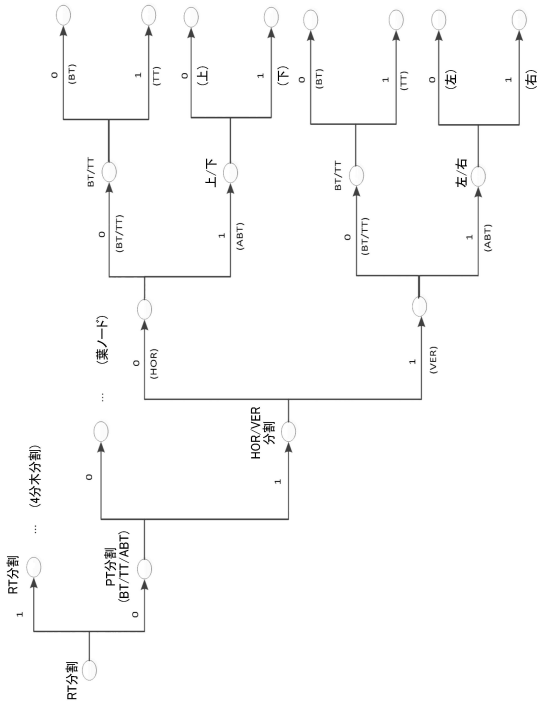
10

20

【図 7】



【図 8】

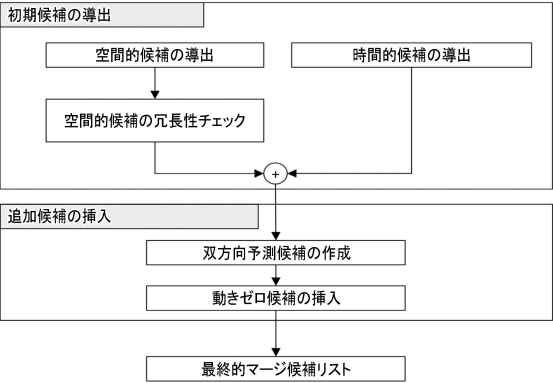


30

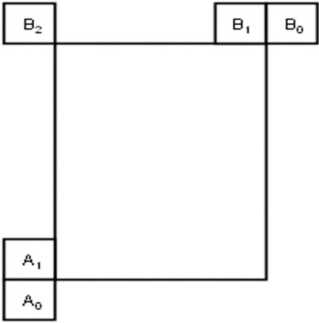
40

50

【図 9】

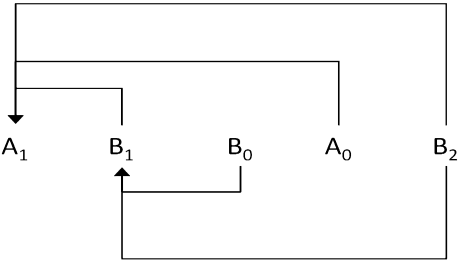


【図 10】

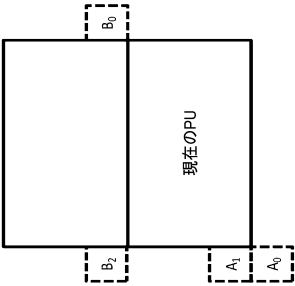


10

【図 11】

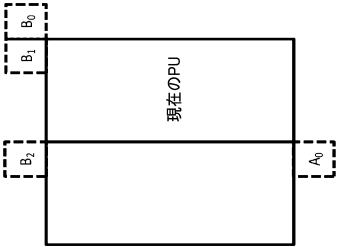


【図 12】



(b) $2N \times N$ の第20PU

20



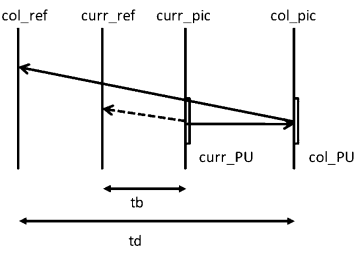
(a) $N \times 2N$ の第20PU

30

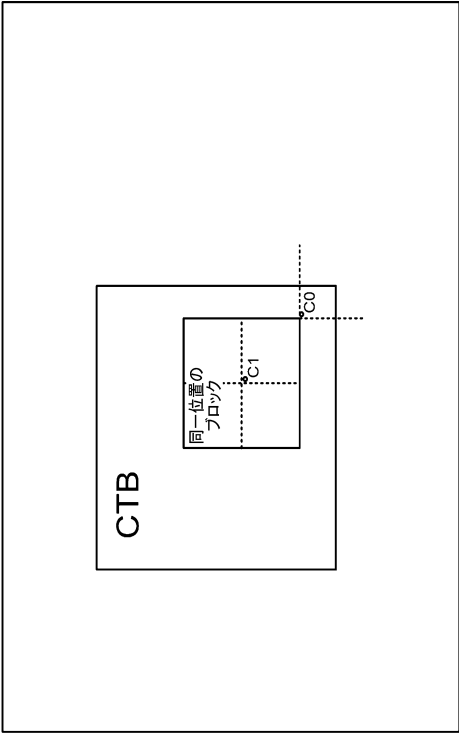
40

50

【図 1 3】



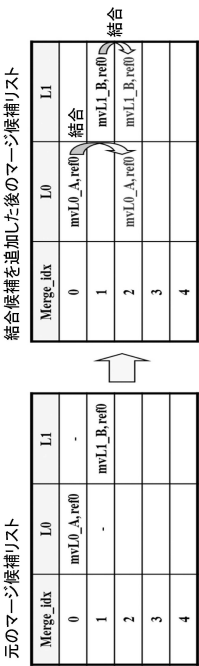
【図 1 4】



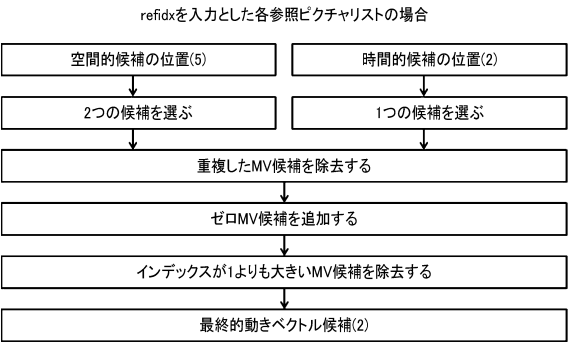
10

20

【図 1 5】



【図 1 6】

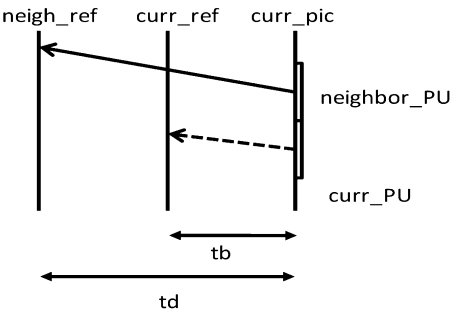


30

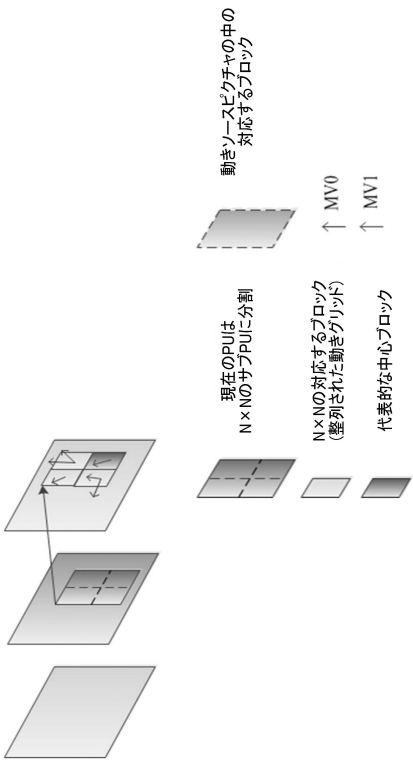
40

50

【図 17】



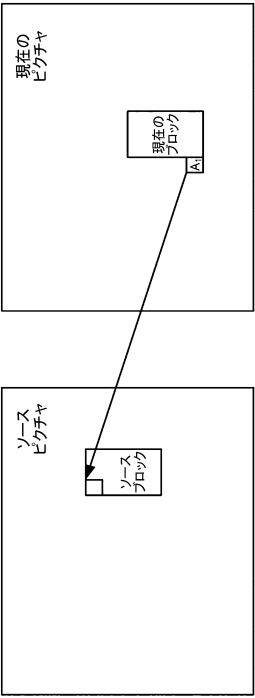
【図 18】



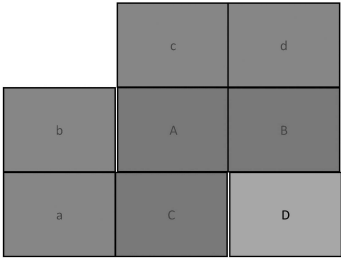
10

20

【図 19】



【図 20】

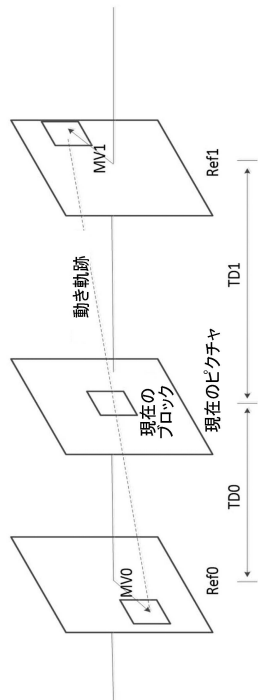


30

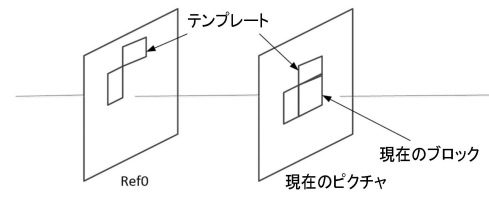
40

50

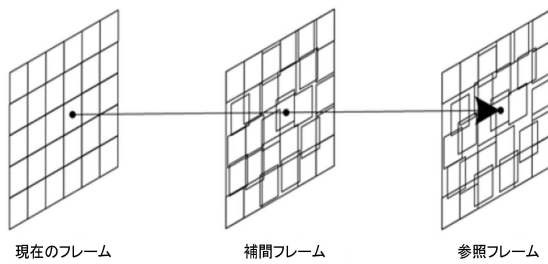
【図 2 1】



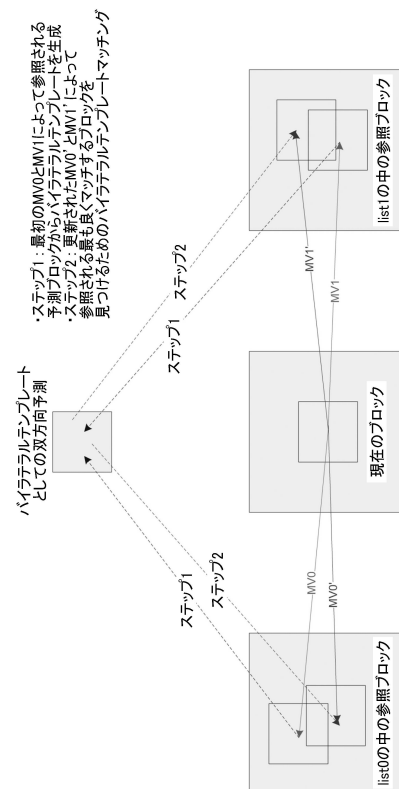
【図 22】



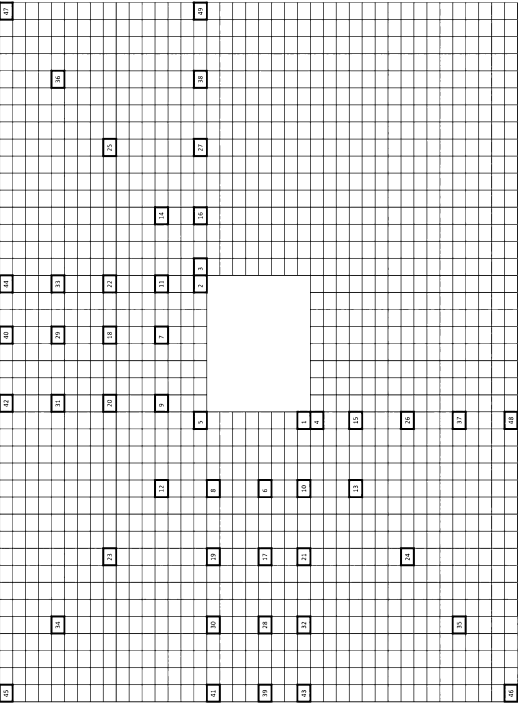
【図 23】



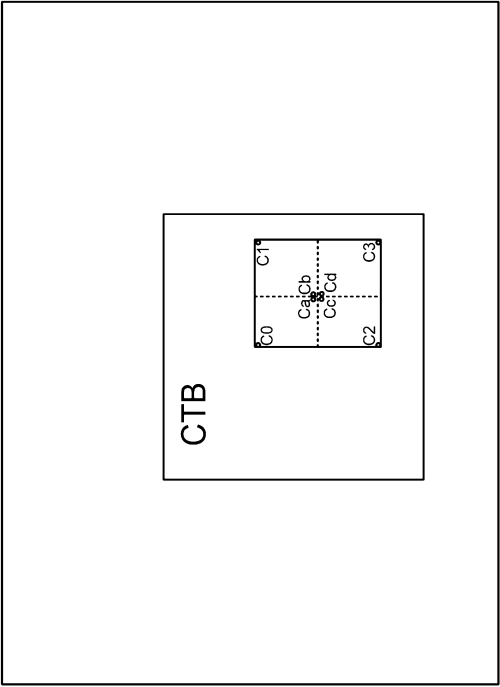
【圖 24】



【図 2 5】



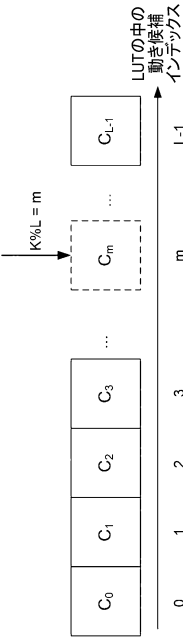
【図 2 6】



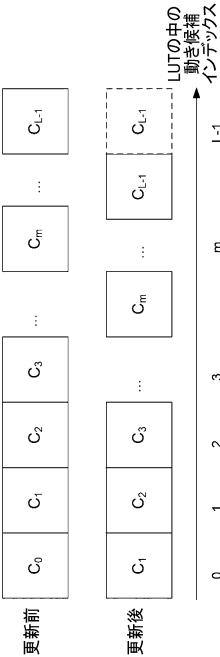
10

20

【図 2 7 A】



【図 2 7 B】

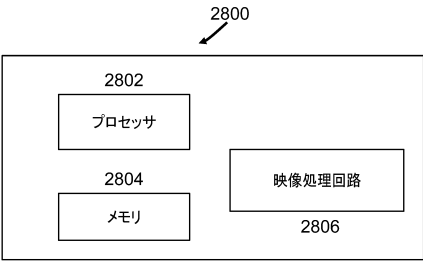


30

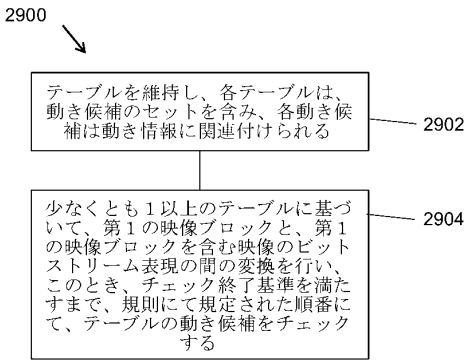
40

50

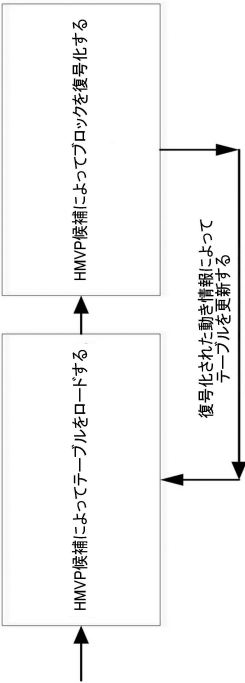
【図 28】



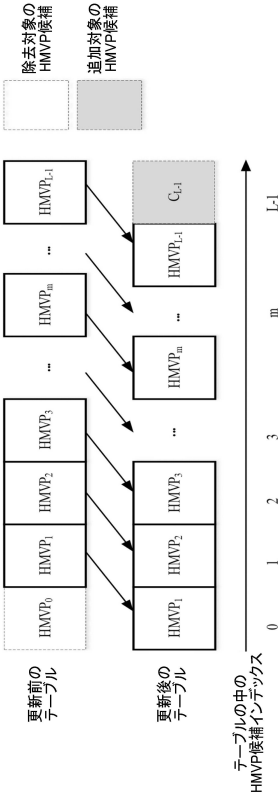
【図 29】



【図 30】



【図 31】



10

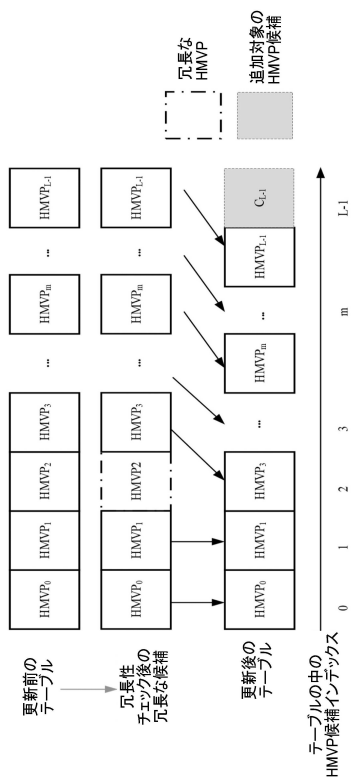
20

30

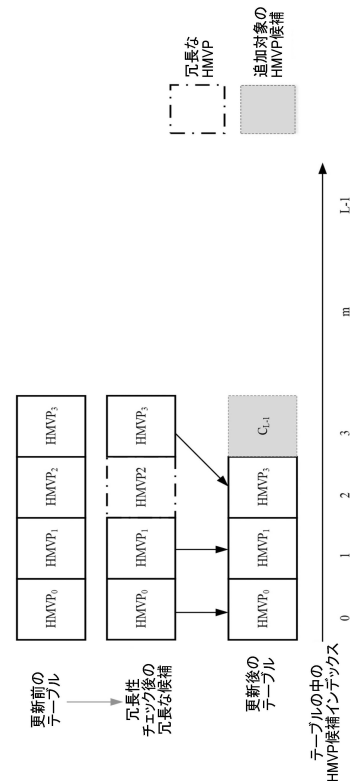
40

50

【図 3 2 A】



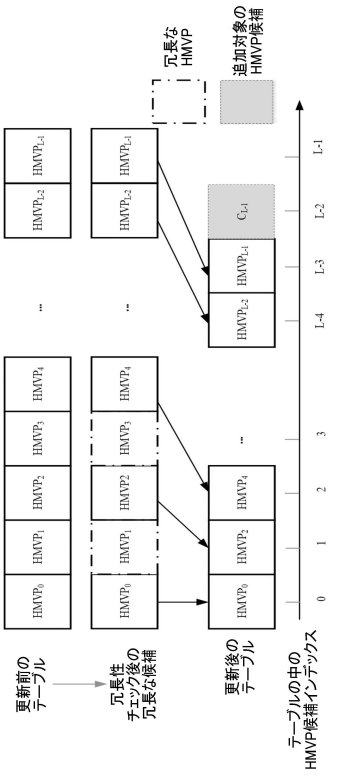
【図 3 2 B】



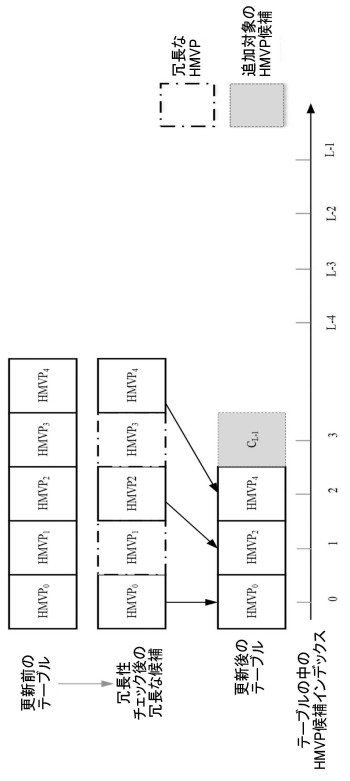
10

20

【図 3 3 A】



【図 3 3 B】



30

40

50

フロントページの続き

(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/CN2018/105193

(32)優先日 平成30年9月12日(2018.9.12)

(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/CN2018/117627

(32)優先日 平成30年11月27日(2018.11.27)

(33)優先権主張国・地域又は機関

中国(CN)

前置審査

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 ブールヴァード 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 ブールヴァード 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 베이징, 하이ディアン ディストリクト, チーチュン 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 베이징, 하이ディアン ディストリクト, チーチュン 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

審査官 鉢呂 健

(56)参考文献 国際公開第2012/172668 (WO, A1)

特表2022-504073 (JP, A)

ZHANG, Li et al., CE4-related: History-based Motion Vector Prediction, Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 11th Meeting: Ljubljana, SI, 10-18 July 2018, [JVET-K0104-v5], JVET-K0104 (version 5), ITU-T, 2018年07月18日, <URL:http://phenix.it-sudparis.eu/jvet/doc_end_user/documents/11_Ljubljana/wg11/JVET-K0104-v5.zip>: JVET-K0104_r4.docx: pp. 1-7

CHIEN, Wei-Jung et al., CE4-related: Modification on History-based Mode Vector Prediction, Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 12th Meeting: Macao, CN, 3-12 Oct. 2018, [JVET-L0401], JVET-L0401 (version 4), ITU-T, 2018年10月05日, <URL:http://phenix.it-sudparis.eu/jvet/doc_end_user/documents/12_Macao/wg11/JVET-L0401-v4.zip>: JVET-L0401_r3.docx: pp. 1-5

ZHANG, Li et al., CE4: History-based Motion Vector Prediction (Test 4.4.7), Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 12th Meeting: Macao, CN, 3-12 Oct. 2018, [JVET-L0266-v2], JVET-L0266 (version 3), ITU-T, 2018年10

月04日，<URL:http://phenix.it-sudparis.eu/jvet/doc_end_user/documents/12_Macao/wg11/JVET-L0266-v3.zip>: JVET-L0266_r1.docx: pp.1-6, JVET-L0266_spec_d1.docx: pp.1-6
(58)調査した分野 (Int.Cl., D B名)
H 0 4 N 1 9 / 0 0 - 1 9 / 9 8