

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7693632号
(P7693632)

(45)発行日 令和7年6月17日(2025.6.17)

(24)登録日 令和7年6月9日(2025.6.9)

(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 17 外国語出願 (全43頁)

<p>(21)出願番号 特願2022-195878(P2022-195878)</p> <p>(22)出願日 令和4年12月7日(2022.12.7)</p> <p>(62)分割の表示 特願2021-523113(P2021-523113)の分割</p> <p>原出願日 令和1年7月1日(2019.7.1)</p> <p>(65)公開番号 特開2023-36646(P2023-36646A)</p> <p>(43)公開日 令和5年3月14日(2023.3.14)</p> <p>審査請求日 令和4年12月27日(2022.12.27)</p> <p>(31)優先権主張番号 PCT/CN2018/093663</p> <p>(32)優先日 平成30年6月29日(2018.6.29)</p> <p>(33)優先権主張国・地域又は機関 中国(CN)</p> <p>前置審査</p>	<p>(73)特許権者 520476341 北京字節跳動網絡技術有限公司 Beijing Bytedance Network Technology Co., Ltd. 中華人民共和国100041北京市石景山区実興大街30号院3号楼2層B-0035房間 Room B-0035, 2/F, No.3 Building, No.30, Shixing Road, Shijingshan District Beijing 100041 China</p> <p>(73)特許権者 520477474 バイトダンス インコーポレイテッド 最終頁に続く</p>
---	--

(54)【発明の名称】 更新されるべき若しくは更新しない何れかのLUT

(57)【特許請求の範囲】

【請求項1】

映像データを処理する方法であって、
履歴ベースの動きベクトル予測(HMVP)テーブルである1つ又は複数のテーブルを維持することであって、前記1つ又は複数のテーブルのそれぞれのテーブルはコーディングされた1つ以上の映像ブロックから導出した1つ以上の動き候補を含み、前記テーブル内の前記動き候補の配置は、前記テーブルへ前記動き候補を追加するシーケンスに基づく、維持することと、

現在の映像ブロックのための動き候補リストを構築することであって、前記構築中に、少なくとも1つの前記動き候補を前記動き候補リストに追加するかどうかを決定するために、前記1つ又は複数のテーブルのうちの前記テーブルの少なくとも1つの動き候補を選択的に順にチェックする、構築することと、

前記動き候補リストを使用して、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とビットストリームとの変換を行うことと、

複数の映像ブロックを含む新しい領域のコーディングの開始時に前記1つ又は複数のテーブルの少なくとも1つの動き候補の数をゼロにリセットすることと、

を含み、

前記1つ又は複数のテーブルは、N個のテーブルを含み、

Nは、 $K * P$ に等しく、

Kは、前記映像データの1つのスライスのCTU行又はタイルに対応する1つのコーディングスレッド当たりのテーブルの数を表す整数であり、

Pは、前記スライスにおけるCTU行の数又はタイルの数を表す整数である、
方法。

【請求項2】

前記1つ又は複数のテーブルのうちの前記テーブルは、前記テーブルにおける許可された動き候補の数を示すサイズを有する、

請求項1に記載の方法。

【請求項3】

新しい動き候補を前記テーブルに加える前に前記テーブルにおける動き候補の数が前記テーブルの前記サイズに達している場合、前記テーブルに前記新しい動き候補を加えられることによって、前記テーブルにおける1つの候補が削除される、

請求項2に記載の方法。

10

【請求項4】

前記テーブルの前記サイズの値は、予め規定された値又は構文要素において信号通知される値のうち的一方である、

請求項2に記載の方法。

【請求項5】

前記テーブルのためのカウンタを維持することを更に含み、
前記カウンタは、前記テーブルにおける動き候補の数を表し、
前記カウンタは、前記テーブルの前記サイズ以下である、
請求項2に記載の方法。

20

【請求項6】

前記新しい領域は、新しいコーディングツリーユニット(CTU)行、新しいタイル又は新しいスライスである、
請求項1に記載の方法。

【請求項7】

前記テーブルの数は、予め規定される、
請求項1に記載の方法。

【請求項8】

前記映像データの1つのスライスのCTU行又は1つのスライスのタイルが1つのテーブルを使用し、且つPが前記スライスにおけるCTU行の数又はタイルの数を表す整数であるときに、NがPに等しい場合、前記1つ又は複数のテーブルは、N個のテーブルを含む、

請求項1に記載の方法。

30

【請求項9】

前記1つ又は複数のテーブルの数は、スライスタイプ、ピクチャの時間層インデックス、1つのスライスと最も近いイントラスライスとの間のピクチャオーダカウント(POC)距離、のうちの少なくとも1つに基づく、

請求項1に記載の方法。

40

【請求項10】

前記テーブルの中の動き候補は、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又は動きベクトル差分値、のうち少なくとも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11】

前記動き候補リストは、動きベクトル予測リスト又はマージ候補リストのうちの一つである、

請求項1に記載の方法。

【請求項12】

50

前記チェックの結果に基づいて、前記動き候補リストに1つ以上のチェック済み候補を加える、

請求項1に記載の方法。

【請求項13】

前記変換は、前記現在の映像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む、請求項1に記載の方法。

【請求項14】

前記変換は、ビットストリームから前記現在の映像ブロックを復号することを含む、請求項1のいずれか1項に記載の方法。

【請求項15】

処理装置と、命令が記憶された非一時的メモリと、を備える、映像データをコーディングする装置であって、

前記命令は、前記処理装置による実行時に、前記処理装置に、履歴ベースの動きベクトル予測(HMVP)テーブルである1つ又は複数のテーブルを維持することであって、前記1つ又は複数のテーブルのそれぞれのテーブルはコーディングされた1つ以上の映像ブロックから導出した1つ以上の動き候補を含み、前記テーブル内の前記動き候補の配置は、前記テーブルへ前記動き候補を追加するシーケンスに基づく、維持することと、

現在の映像ブロックのための動き候補リストを構築することであって、前記構築中に、少なくとも1つの前記動き候補を前記動き候補リストに追加するかどうかを決定するために、前記1つ又は複数のテーブルのうちの前記テーブルの少なくとも1つの動き候補を選択的に順にチェックする、構築することと、

前記動き候補リストを使用して、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とビットストリームとの変換を行うことと、

複数の映像ブロックを含む新しい領域のコーディングの開始時に前記1つ又は複数のテーブルの少なくとも1つの動き候補の数をゼロにリセットすることと、

を行わせ、

前記1つ又は複数のテーブルは、N個のテーブルを含み、

Nは、 $K * P$ に等しく、

Kは、前記映像データの1つのスライスのCTU行又はタイルに対応する1つのコーディングスレッド当たりのテーブルの数を表す整数であり、

Pは、前記スライスにおけるCTU行の数又はタイルの数を表す整数である、

装置。

【請求項16】

命令を記憶する非一時的な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、

履歴ベースの動きベクトル予測(HMVP)テーブルである1つ又は複数のテーブルを維持することであって、前記1つ又は複数のテーブルのそれぞれのテーブルはコーディングされた1つ以上の映像ブロックから導出した1つ以上の動き候補を含み、前記テーブル内の前記動き候補の配置は、前記テーブルへ前記動き候補を追加するシーケンスに基づく、維持することと、

現在の映像ブロックのための動き候補リストを構築することであって、前記構築中に、少なくとも1つの前記動き候補を前記動き候補リストに追加するかどうかを決定するために、前記1つ又は複数のテーブルのうちの前記テーブルの少なくとも1つの動き候補を選択的に順にチェックする、構築することと、

前記動き候補リストを使用して、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とビットストリームとの変換を行うことと、

複数の映像ブロックを含む新しい領域のコーディングの開始時に前記1つ又は複数のテ

10

20

30

40

50

ブルの少なくとも1つの動き候補の数をゼロにリセットすることと、
を行わせ、

前記1つ又は複数のテーブルは、N個のテーブルを含み、
Nは、 $K * P$ に等しく、

Kは、映像データの1つのスライスのCTU行又はタイルに対応する1つのコーディング
スレッド当たりのテーブルの数を表す整数であり、

Pは、前記スライスにおけるCTU行の数又はタイルの数を表す整数である、
 非一時的なコンピュータ可読記憶媒体。

【請求項17】

ビットストリームを記憶する方法であって、

履歴ベースの動きベクトル予測(HMVP)テーブルである1つ又は複数のテーブルを維持
することであって、前記1つ又は複数のテーブルのそれぞれのテーブルは、コーディン
グされた1つ以上の映像ブロックから導出した1つ以上の動き候補を含み、前記テー
ブル内の前記動き候補の配置は、前記テーブルへ前記動き候補を追加するシーケンスに基づ
く、維持することと、

現在の映像ブロックのための動き候補リストを構築することであって、前記構築中に、
少なくとも1つの前記動き候補を前記動き候補リストに追加するかどうかを決定するた
めに、前記1つ又は複数のテーブルのうちの前記テーブルの少なくとも1つの動き候補を選
択的に順にチェックする、構築することと、

前記動き候補リストを使用して、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、前記ビットストリームを生成することと、
前記ビットストリームを非一時的なコンピュータ可読記録媒体に記憶することと、
複数の映像ブロックを含む新しい領域のコーディングの開始時に前記1つ又は複数のテー
ブルの少なくとも1つの動き候補の数をゼロにリセットすることと、

を含み、

前記1つ又は複数のテーブルは、N個のテーブルを含み、

Nは、 $K * P$ に等しく、

Kは、映像データの1つのスライスのCTU行又はタイルに対応する1つのコーディング
スレッド当たりのテーブルの数を表す整数であり、

Pは、前記スライスにおけるCTU行の数又はタイルの数を表す整数である、

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第PCT/CN2018/093663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、国際特許出願第PCT/CN2018/093663号の全開示は、本出願の開示の一部として参照により援用される。

【0002】

本特許明細書は、映像コーディング技術、デバイスおよびシステムに関する。

【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想される。

【発明の概要】

【0004】

10

20

30

40

50

本明細書は、動きベクトルのマージリストを使用してデジタル映像を符号化および復号するための方法、システム、およびデバイスを開示する。

【 0 0 0 5 】

1つの例示的な態様において、映像復号方法が開示される。この方法は、多数の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、以前コーディングされた映像ブロックから導出された対応する動き情報に関連付けられる、維持することと、現在の映像ブロックと映像領域における現在の映像ブロックのビットストリーム表現との間で変換を行うことと、更新規則に基づいて1つ以上のテーブルを更新することと、を含む。

【 0 0 0 6 】

別の例示的な態様において、別の映像復号方法が開示される。この方法は、テーブルのセットをチェックすることを含み、各テーブルは、1つ以上の動き候補を含み、各動き候補が動き候補の動き情報に関連付けられており、1つ以上のテーブルに基づいてこの映像ブロックの動き情報を処理することと、処理によって生成された映像ブロックに基づいて1つ以上のテーブルを更新することと、を含む。

【 0 0 0 7 】

さらに別の例示的な態様において、別の映像デコーディング方法が開示される。この方法は、テーブルのセットをチェックすることを含み、各テーブルは、1つ以上の動き候補を含み、各動き候補が動き候補の動き情報に関連付けられており、ピクチャ内の映像ブロックの位置に基づいて1つ以上のテーブルを選択することと、選択された1つ以上のテーブルに基づいてこの映像ブロックの動き情報を処理することと、処理によって生成された映像ブロックに基づいて選択されたテーブルを更新することと、を含む。

【 0 0 0 8 】

さらに別の例示的な態様において、別の映像デコーディング方法が開示される。この方法は、テーブルのセットをチェックすることを含み、各テーブルは、1つ以上の動き候補を含み、各動き候補が動き候補の動き情報に関連付けられており、映像ブロックと、1つ以上のテーブル内の動き候補のうちの1つとの間の距離に基づいて1つ以上のテーブルを選択することと、選択された1つ以上のテーブルに基づいて、画像ブロックの動き情報を処理することと、処理によって生成された映像ブロックに基づいて選択されたテーブルを更新することと、を含む。

【 0 0 0 9 】

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ビデオエンコーダデバイスが開示される。

【 0 0 1 0 】

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。

【 0 0 1 1 】

さらに別の代表的な態様において、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てもよい。

【 0 0 1 2 】

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。

【 図面の簡単な説明 】

【 0 0 1 3 】

【 図 1 】 ビデオ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 . 2 6 4 映像コーディング規格におけるマクロブロックの分割を示す。

【 図 3 】 コーディングブロック (C B) を予測ブロック (P U) に分割する例を示す。

10

20

30

40

50

【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) 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。

10

【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。

【図 13】時間的マージ候補のための動きベクトルのスケールリングを示す。

【図 14】時間的マージ候補の候補位置とそのコロケーションピクチャを示す。

【図 15】結合双予測マージ候補の例を示す。

【図 16】動きベクトル予測候補の導出処理の例を示す。

【図 17】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ルリングの例を示す。

【図 18】CU の動き予測のための例示的な代替時間的動きベクトル予測 (ATMVP) を示す。

【図 19】ソースブロックおよびソースピクチャの識別の一例を図で示す。

【図 20】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。

20

【図 21】バイラテラルマッチングの例を示す。

【図 22】テンプレートマッチングの例を示す。

【図 23】フレームレートアップ変換 (FRUC) におけるユニラテラル動き推定 (ME) の例を示す。

【図 24】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 25】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 26】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 27】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 28】本明細書に記載されるビジュアルメディアのデコーディングまたはビジュアルメディアのエンコーディング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。

30

【図 29】映像ビットストリーム処理の別の例示的な方法のフローチャートである。

【図 30】映像ビットストリーム処理の別の例示的な方法のフローチャートである。

【発明を実施するための形態】

【0014】

映像の圧縮率を改善するために、研究者らは、映像をコーディングする新しい技術を絶えず求めている。

【0015】

1. 導入

40

【0016】

本明細書は、映像コーディング技術に関する。具体的には、映像コーディングにおける動き情報のコーディング (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コーディング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、将来の映像コーディング規格またはビデオコーデックにも適用可能である。

【0017】

簡単な説明

【0018】

映像コーディング規格は、主に周知の ITU-T および ISO/IEC 規格の開発によ

50

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 高度映像コーディング(AVC)とH.265/HEVC規格を共同で作成した。映像コーディング規格、H.262は、時間予測プラス変換コーディングが利用されるハイブリッド映像コーディング構造に基づく。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。

【0019】

2.1 パーティション構造

【0020】

2.1.1 H.264/AVCにおけるパーティションツリー構造

10

【0021】

以前の規格におけるコーディング層のコアは、 16×16 ブロックの輝度サンプルを含む、また、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 ブロックのクロマサンプル含むマクロブロックであった。

【0022】

イントラコーディングされたブロックは、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。

【0023】

インターコーディングされたブロックは、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)のみが許可される。

20

【0024】

2.1.2 HEVCにおけるパーティションツリー構造

【0025】

HEVCにおいて、CTUは、様々な局所的特徴に適應するように、コーディングツリー(coding tree)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、ピクチャ領域をコーディングするかどうかの決定は、CUレベルで行われる。各CUは、PU分割タイプによって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のためのコーディング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。

30

【0026】

以下、HEVCを使用したハイブリッド映像コーディングに関連する様々な特徴に焦点を当てる。

【0027】

1) コーディングツリーユニットおよびコーディングツリーブロック(CTB)構造: HEVCにおける類似した構造は、コーディングツリーユニット(CTU)であり、このコーディングツリーユニットは、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、対応するクロマ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 個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分木の様な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。

40

【0028】

2) コーディングユニット(CU)およびコーディングブロック(CB): CTUの4

50

分木の構文は、その輝度およびクロマ C B のサイズおよび位置を指定する。4 分木のルートは C T U に関連付けられる。従って、輝度 C T B のサイズは、輝度 C B に対してサポートされる最大のサイズである。C T U を輝度 C B およびクロマ 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 つのクロマ C B は、関連する構文と共に、1 つのコーディングユニット (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 つの変換ユニットのツリー (T U) とを有する。

【 0 0 2 9 】

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コーディングするかどうかの決定は、C U レベルで行われる。P U の分割構造は、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、ルマ度およびクロマ C B のサイズをさらに分割し、輝度およびクロマ予測ブロック (P B) から予測することができる。H E V C は、 64×64 個のサンプルから 4×4 個のサンプルまでの可変 P B サイズをサポートする。図 3 は、 $M \times M$ C U のための許可された P B の例を示す。

【 0 0 3 0 】

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コーディングされる。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。クロマ 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、離散正弦変換 (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。

【 0 0 3 1 】

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。

【 0 0 3 2 】

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

【 0 0 3 3 】

残差コーディングの場合、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リーフノードブロックは、変換コーディング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輝度 T B サイズおよび最小輝度 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、分割を行わないことが、非明示的に行われる。輝度 T B サイズが 4×4 である場合を除き、クロマ 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 クロマ 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。

【 0 0 3 4 】

従来の規格とは対照的に、H E V C 設計により、1 つの T B がインターピクチャ予測 C U のために複数の P B にまたがることを可能となり、4 分木構造の T B の分割の潜在的なコーディング効率の利点が最大となる。

10

20

30

40

50

【 0 0 3 5 】

2 . 1 . 2 . 2 親子ノード

【 0 0 3 6 】

C T B は、4 分木構造に基づいて分割され、そのノードはコーディングユニットである。4 分木構造における複数のノードは、リーフノードおよび非リーフノードを含む。リーフノードは、ツリー構造内に子ノードを持たない（すなわち、リーフノードはそれ以上分割されない）。非リーフノードは、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、C T B）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。複数の非リーフノードのそれぞれの非リーフノードは、ツリー構造において1つ以上の子ノードを有する。

10

【 0 0 3 7 】

2 . 1 . 3 J E Mにおけるより大きいC T Uを有する4分木+2分木ブロック構造

【 0 0 3 8 】

H E V Cを超えた将来の映像コーディング技術を探索するため、2015年には、V C E GとM P E Gが共同で共同映像探索チーム（Joint Video Exploration Team: J V E T）を設立した。それ以来、多くの新しい方法がJ V E Tによって採用され、共同探索モデル（J E M）と呼ばれる参照ソフトウェアに組み込まれてきた。

20

【 0 0 3 9 】

2 . 1 . 3 . 1 Q T B Tブロックの分割構造

【 0 0 4 0 】

H E V Cとは異なり、Q T B T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、C U、P U、T Uのコンセプトの切り離しを取り除き、C Uパーティションの形状の柔軟性を向上させる。Q T B Tブロック構造において、C U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、コーディングツリーユニット（C T U）を4分木構造で分割する。4分木のリーフノードは、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分木のリーフノードは、コーディングユニット（C U）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、Q T B Tコーディングされたブロック構造において、C U、P UおよびT Uが同じブロックサイズを有することを意味する。J E Mにおいて、C Uは、異なる色成分のコーディングされたブロック（C B）からなることがある。例えば、4 : 2 : 0クロマフォーマットのPおよびBスライスの場合、1つのC Uは1つの輝度C Bおよび2つのクロマC Bを含み、時には、単一の成分のC Bからなる。例えば、1つのC Uは、1つの輝度C Bのみを含み、またはIスライスの場合、2つのクロマC Bのみを含む。

30

【 0 0 4 1 】

Q T B T分割スキームに対して以下のパラメータを規定する。

- C T Uのサイズ：1つの4分木のルートノードのサイズ、H E V Cと同じ概念
- M i n Q T S i z e：最小許容4分木のリーフノードサイズ
- M a x B T S i z e：最大許容2分木ルートノードサイズ
- M a x B T D e p t h：最大許容2分木の深さ
- M i n B T S i z e：最小限に許容される2分木のリーフノードのサイズ

40

【 0 0 4 2 】

Q T B Tの分割構造の一例において、C T Uのサイズを、2つの対応する64 x 64ブロックのクロマサンプルを有する128 x 128の輝度サンプルとして設定し、M i n Q T S i z eを16 x 16として設定し、M a x B T S i z eを64 x 64として設定し、M i n B T S i z e（幅および高さの両方について）を4 x 4として設定し、M a x B T D e p t hを4として設定する。4分木の分割は、まずC T Uに適用され、4分木のリー

50

フノードを生成する。4分木のリーフノードのサイズは、 16×16 （即ち、 $MinQTSize$ ）から 128×128 （即ち、 CTU サイズ）までが可能である。リーフ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、リーフ4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分木のリーフノードは、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 $MinBTSize$ に等しい（すなわち、4である）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分木のリーフノードは、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。

10

【0043】

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、非リーフ）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。

20

【0044】

さらに、 $QTBT$ 方式は、輝度およびクロマが別個の $QTBT$ 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クロマ 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、クロマ CTB は別の $QTBT$ 構造によってクロマ 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つのコーディングされたブロックまたは2つのクロマ成分の1つのコーディングされたブロックからなり、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コーディングされたブロックからなることを意味する。

【0045】

$HEVC$ 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。

30

【0046】

2.1.4 VVCの3分木

【0047】

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（d）、（e）に示すように、3分木（ TT ）パーティションを2つ以上、すなわち、水平および垂直中心側の3分木を導入する。

40

【0048】

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（e）水平中心側3分木分割を示す。

【0049】

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 リーフはまた、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 リーフが基本コーディングユニットである。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

50

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。

【0050】

2.1.5 分割構造

【0051】

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、コーディングツリーユニット (CTU) を4分木構造で分割する。4分木のリーフノードは、2分木構造によってさらに分割される。

【0052】

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。

10

【0053】

FIG. 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。

【0054】

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リーフ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)。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、輝度ツリーおよびクロマツリーは、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。

20

【0055】

2.2 HEVC/H.265におけるインター予測

【0056】

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コーディングされてもよく、このようなコーディングモードは、AMVPモードと呼ばれる。

30

【0057】

1つのCUがスキップモードでコーディングされる場合、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、コーディング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としては、動きパラメータの明確な送信があり、PUごとに、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応する参照ピクチャインデックスである、動きベクトルを明確に信号通知する。

40

【0058】

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。

【0059】

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スライスのみで双予測が利用可能である。

【0060】

50

以下、H E V C 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。

【 0 0 6 1 】

2 . 2 . 1 マージモード

【 0 0 6 2 】

2 . 2 . 1 . 1 マージモードの候補の導出

【 0 0 6 3 】

マージモードを使用して P U 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、H E V C 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
 - 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

【 0 0 6 4 】

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 P U ごとに一定数の候補を想定しているため、候補数がスライスヘッダで信号通知されるマージ候補 (M a x N u m M e r g e C a n d) の最大数に達しない場合、追加の候補を生成する。候補の数は一定であるため、短縮された単項 2 値化 (T U) を使用して最良マージ候補のインデックスをエンコードする。C U のサイズが 8 に等しい場合、現在の C U のすべての P U は、 $2 N \times 2 N$ 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。

【 0 0 6 5 】

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。

【 0 0 6 6 】

2 . 2 . 1 . 2 空間的候補導出

【 0 0 6 7 】

空間的マージ候補の導出において、図 1 0 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 P U 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)、またはイントラコーディングされた場合にのみ、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、コーディング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 1 1 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、 $2 N \times 2 N$ とは異なるパーティションに関連付けられた「第 2 の P U」である。一例として、図 1 2 は、それぞれ $N \times 2 N$ および $2 N \times N$ の場合の第 2 の P U を示す。現在の P U を $N \times 2 N$ 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 つのコーディングユニットに 1 つの P U のみを有することは冗長である。同様に、現在の P U を $2 N \times N$ に分割する場合、位置 B 1 は考慮されない。

【 0 0 6 8 】

10

20

30

40

50

2.2.1.3 時間的候補導出

【0069】

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コロケーション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、コロケーション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、コロケーション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、コロケーションPUの参照ピクチャとコロケーションピクチャの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。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。

10

【0070】

参照フレームに属するコロケーション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、イントラコーディングされている場合、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。

20

【0071】

2.2.1.4 追加候補挿入

【0072】

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。時空間的マージ候補を利用して、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15は、オリジナルリスト(左側)における、 mvL_0 、 $refIdxL_0$ または mvL_1 、 $refIdxL_1$ 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。

30

【0073】

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ぬ。

40

【0074】

2.2.1.5 並列処理のための動き推定領域

【0075】

エンコーディング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。コーディング効率と処理待ち時間との間のトレードオフを緩和する

50

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。

【0076】

7.3.2.3 ピクチャパラメータセットRBS P構文

【0077】

7.3.2.3.1 一般ピクチャパラメータセットRBS P構文

【0078】

【表1】

10

pic_parameter_set_rbsp() {	Descriptor
pps_pic_parameter_set_id	ue(v)
pps_seq_parameter_set_id	ue(v)
dependent_slice_segments_enabled_flag	u(1)
...	
pps_scaling_list_data_present_flag	u(1)
if(pps_scaling_list_data_present_flag)	
scaling_list_data()	
lists_modification_present_flag	u(1)
log2_parallel_merge_level_minus2	ue(v)
slice_segment_header_extension_present_flag	u(1)
pps_extension_present_flag	u(1)
...	
rbbsp_trailing_bits()	
}	

20

30

【0079】

log2_parallel_merge_level_minus2 plus 2は、8.5.3.2.2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~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 (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、64x64ブロックに含まれたすべての予測ユニット(PU)およびコーディングユニット(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。

40

【0080】

2.2.2 AMVPモードにおける動きベクトル予測

【0081】

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-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

50

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エンコードされる。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、動きベクトル予測候補の導出処理の詳細を説明する。

【0082】

2.2.2.1 動きベクトル予測候補の導出

【0083】

図16に、動きベクトル予測候補の導出処理をまとめる。

【0084】

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。

【0085】

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。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。

【0086】

2.2.2.2 空間的動きベクトル候補

【0087】

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つの潜在的な候補のうち、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、A0、A1、スケーリングされたA0、スケーリングされたA1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、B0、B1、B2、スケーリングされたB0、スケーリングされたB1、スケーリングされたB2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）

【0088】

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、またはイントラコーディングされている場合、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。

【0089】

10

20

30

40

50

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、近傍の P U の動きベクトルをスケーリングする。主な違いは、現在の P U 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。

【 0 0 9 0 】

2 . 2 . 2 . 3 時間的動きベクトル候補

【 0 0 9 1 】

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。

10

【 0 0 9 2 】

2 . 2 . 2 . 4 A M V P 情報の信号通知

【 0 0 9 3 】

A M V P 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、M V D、および m v 予測子候補インデックスを信号通知することができる。

【 0 0 9 4 】

構文テーブル：

【 0 0 9 5 】

20

30

40

50

【表 2】

prediction_unit(x0, y0, nPbW, nPbH) {	Descriptor	
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]) {		10
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)	20
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		30
MvdL1[x0][y0][1] = 0		
} else		
mvd_coding(x0, y0, 1)		
mvp_l1_flag [x0][y0]	ae(v)	
}		
}		
}		
}		40

【 0 0 9 6 】

7 . 3 . 8 . 9 動きベクトル差構文

【 0 0 9 7 】

【表 3】

	Descriptor
mvd_coding(x0, y0, refList) {	
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

【 0 0 9 8 】

2.3 共同探索モデル (J E M) における新しいインター予測方法

【 0 0 9 9 】

2.3.1 サブCUに基づく動きベクトル予測

【 0 1 0 0 】

Q T B Tを有するJ E M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、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 (A T M V P) 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。空間的 - 時間的動きベクトル予測 (S T M V P) 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

【 0 1 0 1 】

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

【 0 1 0 2 】

2.3.1.1 代替の時間的動きベクトル予測

【 0 1 0 3 】

代替的な時間的動きベクトル予測 (A T M V P) において、動きベクトル時間的動きベクトル予測 (T M V P) 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) を取り出すことで修正される。図 1 8 に示すように、サブCUは、 $N \times N$ 個のブロックの正方形である (デフォルトでは、 N は4に設定される) 。

【 0 1 0 4 】

50

A T M V Pは、C U内のサブC U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のステップは、図18に示すように、現在のC UをサブC Uに分割し、各サブC Uに対応するブロックから各サブC Uの動きベクトルならびに参照インデックスを得る。

【0105】

第1のステップにおいて、現在のC U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、現在のC U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、A T M V Pでは、T M V P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C Uに対して右下または中心位置にある。1つの例において、第1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M V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。

10

【0106】

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。

【0107】

第2のステップにおいて、現在のC U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C Uの対応するブロックを特定する。サブC U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C U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、H E V CのT M V Pと同様に、現在のサブC Uの動きベクトルおよび参照インデックスに変換され、動きスケール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P O Cが現在のピクチャのP O C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 $M V_x$ （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C Uの動きベクトル $M V_y$ （ X が0または1に等しく、 Y が $1 - X$ に等しい）を予測する。

20

【0108】

2.3.1.2 空間的 - 時間的動きベクトル予測

30

【0109】

この方法において、サブC U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C U、A、B、C、およびDを含む 8×8 C U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。

【0110】

サブC Uの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C Uの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、またはイントラコーディングされている場合、サブC U 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C U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C Uの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ルリングする。次に、H E V Cに規定されているT M V P導出と同様の手順に従って、サブブロックAの時間的動きベクトル予測子（T M V P）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ルリングする。最後に、動き情報を検索し、スケールリングした後、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C Uの動きベクトルとする。

40

50

【 0 1 1 1 】

図 2 0 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 C U の例を示す。

【 0 1 1 2 】

2 . 3 . 1 . 3 サブ C U 動き予測モード信号通知

【 0 1 1 3 】

サブ C U 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。 A T M V P モードおよび S T M V P モードを表すように、各 C U 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 A T M V P および S T M V P 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エンコーディングロジックは、H M 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 C U について、2 つの追加のマージ候補に対して 2 回以上の R D チェックが必要となる。

10

【 0 1 1 4 】

J E M において、マージインデックスのすべてのピンは、C A B A C によってコンテキストコーディングされる。一方、H E V C においては、第 1 のピンのみがコンテキストコーディングされ、残りのピンはコンテキストバイパスコーディングされる。

【 0 1 1 5 】

2 . 3 . 2 適応型動きベクトル差解像度

【 0 1 1 6 】

H E V C 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、4 分の 1 輝度サンプルの単位で動きベクトルの差 (M V D) (動きベクトルと P U の予測動きベクトルとの差) が信号通知される。J E M において、ローカル適応型動きベクトル解像度 (L A M V R) が導入される。J E M において、M V D 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。M V D 分解能はコーディングユニット (C U) レベルで制御され、M V D 解像度フラグは、少なくとも 1 つのノンゼロ M V D モジュールを有する各 C U に対して条件付きで信号通知される。

20

【 0 1 1 7 】

少なくとも 1 つのノンゼロ M V D モジュールを有する C U の場合、1 / 4 輝度サンプル M V 精度が C U 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 M V 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 M V 精度が使用されるかまたは 4 輝度サンプル M V 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。

30

【 0 1 1 8 】

C U の第 1 の M V D 解像度フラグがゼロであるか、または C U に対してコーディングされていない (つまり、C U におけるすべての M V D がゼロである) 場合、C U に対して 1 / 4 輝度サンプル M V 解像度が使用される。C U が整数輝度サンプル M V 精度または 4 輝度サンプル M V 精度を使用する場合、C U の A M V P 候補リストにおける M V P を対応する精度に丸める。

40

【 0 1 1 9 】

エンコーダにおいて、C U レベルの R D チェックは、どの M V D 解像度を C U 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 M V D 解像度ごとに 3 回、C U レベルの R D チェックを行う。エンコーダの速度を速めるために、J E M においては、以下の符号化方式が適用される。

【 0 1 2 0 】

通常の 1 / 4 輝度サンプル M V D 解像度を有する C U の R D チェック中、現在の C U 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 M V D 解像度を有する同じ C U の R D 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

50

動き推定処理が3回重複しない。

【0121】

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。

【0122】

2.3.3 パターンマッチング動きベクトル導出

【0123】

パターンマッチング動きベクトル導出(PMMVD)モードは、フレームレートアップ変換(FRUC)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。

10

【0124】

FRUCフラグは、そのマージフラグが真である場合、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を信号通知することができ、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。

【0125】

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コストの選択に基づく。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(例えば、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、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。

20

【0126】

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、バイラテラルマッチングまたは開始点付近の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。

30

【0127】

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、 $M \times M$ 個のサブ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、予め規定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。

40

【0128】

【数1】

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

【0129】

50

図 2 1 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で、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、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、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、現在のピクチャと2つの参照ピクチャと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。

【0130】

図 2 2 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、（第2の既存のAMVP候補を取り除くことを意味する）リストサイズを2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。

【0131】

2.3.3.1 CUレベルMV候補セット

【0132】

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) 左上の近傍の動きベクトル

【0133】

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)であり、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。

【0134】

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。具体的には、現在のCUの(0, 0)、(W/2, 0)、(0, H/2)、(W/2, H/2)の位置の補間MVを加算する。

【0135】

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。

【0136】

CUレベルにおいて、AMVP CUのための15個のMVおよびマージCUに対し、13個までのMVを候補リストに加える。

【0137】

2.3.3.2 サブCUレベルMV候補セット

【0138】

10

20

30

40

50

サブCUレベルのMV候補セットは、以下からなる。

- (i) CUレベルの検索から決定されたMV、
- (i i) 左上、右上の近傍のMV、
- (i i i) 参照ピクチャから並置されたMVのスケーリングされたバージョン、
- (i v) 最大4つのATMVP候補、
- (v) 最大4つのSTMVP候補

【0139】

参照ピクチャからのスケーリングされたMV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CUの配列位置にあるMVは、開始CUレベルMVの参照に対してスケーリングされる。

10

【0140】

ATMVPおよびSTMVPの候補は、最初の4つの候補に限定される

【0141】

サブCUレベルにおいて、最大17個のMVが候補リストに追加される。

【0142】

2.3.3.3 補間MVフィールドの生成

【0143】

あるフレームをコーディングする前に、片側ME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CUレベルまたはサブCUレベルのMV候補として使用してもよい。

20

【0144】

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TD0およびTD1に基づいて(HEVCにおけるTMVPのMV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(図23に示す)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MV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。

【0145】

30

2.3.3.4 補間およびマッチングコスト

【0146】

1つの動きベクトルが1つの小数のサンプル位置を指す場合、動き補償補間が必要である。複雑性を低減するために、通常の8タップHEVC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。

【0147】

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。CU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絶対和差(SAD)である。開始MVを決定した後、サブCU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。

40

【0148】

【数2】

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

【0149】

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。

【0150】

50

F R U Cモードにおいて、M V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、M Cインター予測のために、輝度およびクロマの両方に使用される。M Vを決定した後、輝度用の8タップ補間フィルタおよびクロマ用の4タップ補間フィルタを使用して、最終的なM Cを行う。

【0151】

2.3.3.5 M Vの改良

【0152】

M V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M V検索である。J E Mでは、2つの検索パターン、即ち、無制限中心バイアス菱形検索(U C B D S)およびC UレベルおよびサブC UレベルでのM V改良のための適応クロス検索をそれぞれサポートする。C UおよびサブC UレベルのM V改善の両方のために、M Vは、1/4輝度サンプルM V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M Vの改良が行われる。C UおよびサブC UステップのためのM V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。

10

【0153】

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

【0154】

バイラテラルマッチングマージモードにおいては、双予測が常に適用される。なぜなら、2つの異なる参照ピクチャにおける現在のC U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、C Uの動き情報を導出するからである。テンプレートマッチングマージモードについては、そのような限定はない。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、l i s t 0からの単一予測、l i s t 1からの単一予測、またはC Uのための双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。

20

$cost_{Bi} \leq factor * \min(cost_0, cost_1)$ の場合

双予測を用いる。

それ以外の場合であって、 $cost_0 \leq cost_1$ の場合、

l i s t 0からの単一予測を用いる。

そうでない場合、

l i s t 1からの単一予測を用いる。

30

【0155】

ここで、 $cost_0$ はl i s t 0テンプレートマッチングのS A Dであり、 $cost_1$ はl i s t 1テンプレートマッチングのS A Dであり、 $cost_{Bi}$ は双予測テンプレートマッチングのS A Dである。 $factor$ の値が1.25である場合、選択処理が双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、C U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。

【0156】

2.3.4 デコーダ側動きベクトル改良

【0157】

双予測演算において、1つのブロック領域を予測するために、l i s t 0の動きベクトル(M V)およびl i s t 1のM Vをそれぞれ使用して構成される双予測ブロックを組み合わせ、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 M V 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M V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歪みに基づく検索を行う。

40

【0158】

D M V Rにおいて、図23に示すように、l i s t 0の最初のM V 0とl i s t 1のM V 1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

50

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。絶対差の合計(SAD)をコスト尺度として使用する。

【0159】

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。

10

【0160】

2.3.5 バイラテラルマッチングの改良を伴うマージ/スキップモードの例

【0161】

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、高度な時間的MVP（ATMVP）候補、空間的時間的MVP（STMVP）候補、および追加候補を挿入することで構築される。

20

【0162】

- ブロック1～4の空間的候補

【0163】

- ブロック1～4の外挿アフィン候補

【0164】

- ATMVP

【0165】

- STMVP

30

【0166】

- 仮想アフィン候補

【0167】

- 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）

【0168】

- 外挿アフィン候補（ブロック5）

【0169】

- 時間的候補（HEVCのように導出）

40

【0170】

- 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）

【0171】

- 結合候補

【0172】

- ゼロ候補

【0173】

なお、ICフラグは、STMVP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、双予測のものを単一予測のもの前に挿入する。

【0174】

50

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でコーディングされている場合、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。

【0175】

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

【0176】

現在のHEVC設計は、動き情報をよりよくコーディングするために、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。

【0177】

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、付加的なコーディング利得をもたらし、ハードウェア実装の複雑性を大幅に増大させる。

【0178】

4. いくつかの例

【0179】

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つ以上のルックアップテーブルを使用するLUTに基づく動きベクトル予測技術を実装し、より高いコーディング効率を有する映像符号化を提供することができる。各LUT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、LICフラグ、アフィンフラグ、動きベクトル導出(MVD)精度、および/またはMVD値の一部または全部を含んでもよい。動き情報は、動き情報がどこから来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。

【0180】

開示される技術に基づいたLUT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、様々な実装形態のために以下の例で説明される。LUTは、履歴データ（例えば、既に処理されたブロック）に基づいてエンコーディング/デコーディング処理を行うことを可能にするため、LUTに基づく動きベクトル予測は、履歴ベースの動きベクトル予測(HMVP)法と呼ぶこともできる。LUTに基づく動きベクトル予測方法において、以前にコーディングされた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、エンコーディング/デコーディング処理時に維持される。1つのブロックの符号化/復号の間、LUTにおける関連付けられた動き情報を動き候補リストに追加して、1つのブロックを符号化/復号した後、LUTを使用してもよい。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。

【0181】

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。

【0182】

以下の実施例は、一般的な概念を説明する。

【0183】

ルックアップテーブルの例

【0184】

10

20

30

40

50

例 A 1 : 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、動き情報がどこから来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。

c . ルックアップテーブルごとに 1 つのカウンタをさらに割り当ててもよい。

i . ピクチャ / スライス / LCU (CTU) 行 / タイルの符号化 / 復号の開始時に、カウンタをゼロに初期化してもよい。

ii . 一例において、前記カウンタは、CTU / CTB / CU / CB / PU / 一定の領域サイズ (例えば、 8×8 または 16×16) を符号化 / 復号した後に更新されてもよい。

iii . 一例において、1 つの候補をルックアップテーブルに加える度に、カウンタを 1 つずつ増加させる。

iv . 一例において、カウンタは、テーブルのサイズ (許容される動き候補の数) 以下であるべきである。

v . 代替的に、このカウンタを使用して、いくつの動き候補をルックアップテーブルに追加しようとしたかを示してもよい (これらのうちのいくつかはルックアップテーブルに含まれていたが、後にテーブルから削除してもよい)。この場合、カウンタはテーブルのサイズよりも大きくてもよい。

d . テーブルのサイズ (許可される動き候補の数) および / またはテーブルの数は、固定であってもよいし、適応的であってもよい。テーブルのサイズは、すべてのテーブルで同じであってもよいし、異なるテーブルで異なってもよい。

i . 代替的に、異なるサイズを異なるルックアップテーブル (例えば、1 または 2) に使用してもよい。

ii . 一例において、テーブルのサイズおよび / またはテーブルの数は、予め規定されてもよい。

iii . 一例において、テーブルのサイズおよび / またはテーブルの数は、映像パラメータセット (VPS)、シーケンスパラメータセット (SPS)、ピクチャパラメータセット (PPS)、スライスヘッダ、タイルヘッダ、コーディングツリーユニット (CTU)、コーディングツリーブロック (CTB)、コーディングユニット (CU) または予測ユニット (PU)、複数の CTU / CTB / CU / PU を含む領域で信号通知されてもよい。

iv . テーブルのサイズおよび / または数は、スライスタイプ、ピクチャの時間層インデックス、1 つのスライスと最も近いイントラスライスとの間のピクチャ順序カウント (POC) 距離に更に依存してもよい。

e . 1 つのコーディングスレッドに使用される N 個のテーブルが存在し、1 つのスライスをコーディングするために $N * P$ 個のテーブルが必要とされ、ここで、P は、LCU 行の数またはタイルの数を示す。

i . 代替的に、1 つのスライスをコーディングするために P 個のテーブルのみが必要とされてもよく、P は LCU 行の数を示し、各 LCU 行は 1 つのルックアップテーブルのみを使用し、タイルが無効とされている場合、N 個でも 1 よりも大きくてもよい。

【0185】

LUT の選択

【0186】

例 B 1 : 1 つのブロックをコーディングする場合、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 つのブロックをコーディングする間に 1 つの動き候補をチェックするとき、この動き候補を動き候補リスト (例えば、AMP、マージ候補リスト) に加えてもよい。

10

20

30

40

50

a . 代替的に、複数のルックアップテーブルからの動き候補を順にチェックしてもよい。

b . ルックアップテーブルインデックスは、CTU、CTB、CU若しくはPU、または複数のCTU/CTB/CU/PUを含む領域において信号通知されてもよい。

【0187】

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。

a . それはブロックを含むCTUアドレスに依存してもよい。ここで、アイデアを説明するために、例えばデュアルルックアップテーブル(DLUT)を挙げる。

i . ブロックがCTU行内の第1のM個のCTUのうちの一つに位置する場合、第1のルックアップテーブルをブロックのコーディングに利用してもよく、CTU行内の残りのCTUに位置するブロックの場合、第2のルックアップテーブルを利用してもよい。

ii . ブロックがCTU行内の第1のM個のCTUの一つに位置する場合、まず第1のルックアップテーブルの動き候補によってブロックをコーディングするかどうかをチェックし、第1のテーブルに十分な候補がない場合、第2のルックアップテーブルをさらに利用することができる。一方、CTU行の残りのCTUに位置するブロックに対して、第2のルックアップテーブルを利用してもよい。

iii . 代替的に、CTU行の残りのCTUに位置するブロックについて、第2のルックアップテーブルの動き候補がまずブロックのコーディングについてチェックされ、第2のテーブルに十分な候補がない場合、第1のルックアップテーブルをさらに利用してもよい。

b . それは、一つ以上のルックアップテーブルにおけるブロックの位置と一つの動き候補に関連付けられた位置との間の距離に依存してもよい。

iv . 一例において、一つの動き候補がコーディング対象のブロックまでのより小さな距離に関連付けられている場合、別の動き候補と比較して早くチェックされてもよい。

【0188】

ルックアップテーブルの使用法

【0189】

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。

a . それは、コーディング情報、ブロックサイズ、ブロック形状等にさらに依存してもよい。例えば、AMVPモードの場合、m個の動き候補のみをチェックし、マージモードの場合、n個の動き候補をチェックしてもよい(例えば、m=2、n=44)。

b . 一例において、チェック対象の動き候補の総数は、映像パラメータセット(VPS)、シーケンスパラメータセット(SPS)、ピクチャパラメータセット(PPS)、スライスヘッダ、タイルヘッダ、コーディングツリーユニット(CTU)、コーディングツリーブロック(CTB)、コーディングユニット(CU)または予測ユニット(PU)、複数のCTU/CTB/CU/PUを含む領域で信号通知されてもよい。

【0190】

例C2：一つのルックアップテーブルに含まれる一つ以上の動き候補は、一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コーディングに使用してもよい。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コーディングに使用してもよい。

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。

i . TMVP候補を挿入した後、マージ候補リストに空きがある。

ii . 空間的マージ候補導出のために特定の空間的に近傍のブロックをチェックした後、マージ候補リストに空きがある。

10

20

30

40

50

- i i i . すべての空間的マージ候補の後、マージ候補リストに空きがある。
- i v . 結合双予測マージ候補の後、マージ候補リストに空きがある。
- v . マージ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。

【0191】

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コーディングする予測モジュールとして用いられてもよい。

a . それらをAMVPモードコーディングに使用してもよい。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候補リストに空きがある。

i i . 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、空きがある。

i i i . 上位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/または左側の近傍のブロックからのAMVP候補がスケーリング無しで存在しない場合。

i v . 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。

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(MVスケーリングされた状態で)チェックする。

i i . 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。

i i i . 代替的に、マージした後、動き候補をチェックする。

【0192】

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 . 第1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、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)。

d . 代替的に、さらに、動き候補指標の降順に基づく。

e . 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。

f . 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。

g . 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。

h . 代替的に、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、終了する。

i . 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。

【0193】

例C5：1つのブロックの動き情報コーディング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。

10

20

30

40

50

【 0 1 9 4 】

例 C 6 : 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、さらにコーディング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。

a . 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、ブロックサイズ/ブロック形状に依存してもよい。一例において、より小さなブロック、例えば、 4×4 、 8×4 または 4×8 ブロックについては、ルックアップテーブルからの予測を実行することを許可されない。

b . 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、ブロックがAMVPでコーディングされているかマージモードでコーディングされているかに依存してもよい。一例において、AMVPモードの場合、ルックアップテーブルからの予測を行うことは許可されない。

c . 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、ブロックがアフィンの動きまたは他の種類の動き(例えば、並進的な動き)でコーディングされるかどうかに依存してもよい。一例において、アフィンモードの場合、ルックアップテーブルからの予測を行うことは許可されない。

【 0 1 9 5 】

例 C 7 : 以前コーディングされたフレーム/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。

a . 一例において、現在のブロックの参照ピクチャに関連付けられたルックアップテーブルのみを、現在のブロックをコーディングするために利用してもよい。

b . 一例において、現在のブロックをコーディングするために、現在のブロックの同じスライスタイプおよび/または同じ量子化パラメータを有するピクチャに関連付けられたルックアップテーブルのみを利用してもよい。

【 0 1 9 6 】

ルックアップテーブルの更新

【 0 1 9 7 】

例 D 1 : 動き情報を有するブロックをコーディングした後(すなわち、IntraBCモード、インターコーディングモード)に、1つ以上のルックアップテーブルを更新してもよい。

a . 一例において、ルックアップテーブルを更新するかどうかは、ルックアップテーブルを選択するための規則を再利用してもよい。

b . ルックアップテーブルは、コーディング情報および/またはブロック/LCUの位置に基づいて更新されてもよい。

c . ブロックが直接信号通知される動き情報(例えば、AMVPモード)でコーディングされる場合、ブロックの動き情報をルックアップテーブルに加えてもよい。

i . 代替的に、ブロックが、いかなる改良も伴わずに、空間的に近傍のブロックから直接継承された動き情報でコーディングされる場合(例えば、改良を伴わない空間的マージ候補)、ブロックの動き情報をルックアップテーブルに加えるべきではない。

ii . 代替的に、ブロックが、改良を行って、空間的に近傍のブロックから直接継承された動き情報でコーディングされる場合(DMVR、FRUCなど)、ブロックの動き情報をいかなるルックアップテーブルにも加えるべきではない。

iii . 代替的に、ブロックが、ルックアップテーブルに記憶された動き候補から直接継承された動き情報でコーディングされている場合は、ブロックの動き情報は、いかなるルックアップテーブルにも加えるべきではない。

d . ブロック内の M ($M \geq 1$)個の代表位置を選択し、この代表位置に関連付けられた動き情報を使用してルックアップテーブルを更新する。

10

20

30

40

50

i . 一例において、代表位置は、ブロック内の4つのコーナー位置（例えば、図26のC0～C3）のうちの1つとして規定される。

ii . 一例において、代表位置は、ブロック内の中心位置（例えば、図26におけるCa__Cd）として規定される。

iii . ブロックに対してサブブロック予測が許可されない場合、Mは1に設定される。

iv . サブブロック予測がブロックに対して許可される場合、Mは、1又はサブブロックの総数、又は1とサブブロックの数の間の一意の値に排他的に設定され得る。

v . 代替的に、ブロックのためにサブブロック予測を許可する場合、Mを1に設定することができ、代表的なサブブロックの選択は、に基づいて行われる。

1 . 利用される動き情報の周波数

2 . 双予測ブロックであるかどうか

3 . 参照ピクチャインデックス/参照ピクチャに基づいて

4 . 他の動きベクトルと比較した動きベクトルの差（例えば、最大MV差を選択する）

5 . 他のコーディング情報

e . M ($M > = 1$) 個の代表位置のセットを選択してルックアップテーブルを更新する場合、更なる条件をチェックした後、これらを更なる動き候補としてルックアップテーブルに加えてもよい。

i . ルックアップテーブルにおける既存の動き候補に対して、新たな動き情報のセットをプルーニングしてもよい。

ii . 一例において、新しい動き情報のセットは、ルックアップテーブルにおける既存の動き候補のいずれかまたは一部と同一であってはならない。

iii . 代替的に、新しい動き情報のセットおよび1つの既存の動き候補からの同じ参照ピクチャの場合、MVの差は、1/複数の閾値よりも小さくしてはならない。例えば、MV差の水平および/または垂直モジュールは、1ピクセルの距離よりも大きくなければならない。

iv . 代替的に、 $K > L$ である場合、新しい動き情報のセットを、最後のK個の候補または最初のK%L個の既存の動き候補によってのみプルーニングし、古い動き候補を再びアクティブにすることができるようにする。

v . 代替的に、プルーニングは適用しない。

f . M個の動き情報のセットがルックアップテーブルを更新するために用いられる場合、対応するカウンタをMだけ増加させるべきである。

g . 現在のブロックをコーディングする前に、1つの（上記の方法で）選択された動き情報のセットに対して、更新されるべきルックアップテーブルのカウンタをKとし、このブロックをコーディングした後、これを、K%Lに等しいインデックスを有する追加の動き候補として加える（ここで、Lはルックアップテーブルのサイズである）。その例を図27に示す。

i . 代替的に、それは、 $\min(K + 1, L - 1)$ に等しいインデックスを有する追加の動き候補として追加される。代替的に、 $K > = L$ である場合、第1の動き候補（インデックス=0）をルックアップテーブルから取り除き、後続のK個の候補インデックスを1だけ減らす。

h . 1つのイントラ制約ブロックをコーディングした後、ルックアップテーブルを空にしてもよい。

i . 動き情報のエントリをルックアップテーブルに追加する場合、動き情報から導出することによって、より多くの動き情報のエントリをテーブルに追加してもよい。この場合、ルックアップテーブルに関連付けられたカウンタを1より大きい数だけ増加させることができる。

i . 一例において、動き情報のエントリのMVをスケールリングし、テーブルに入れる。

10

20

30

40

50

i i . 一例において、動き情報のエントリのMVは、(dx, dy)だけ加算され、テーブルに入れられる。

i i i . 一例において、2つ以上の動き情報のエントリのMVの平均を計算し、テーブルに入れる。

【0198】

例D2: 1つのブロックが1つのピクチャ/スライス/タイルの境界に位置する場合、ルックアップテーブルの更新は常に許可されない。

【0199】

例D3: 現在のLCU行をコーディングするために、上記LCU行の動き情報を無効にしてもよい。

a . この場合、新しいスライス/タイル/LCU行の始まりにおいて、利用可能な動き候補の数を0にリセットしてもよい。

【0200】

例D4: 新しい時間層インデックスを使用してスライス/タイルをコーディングする開始時に、利用可能な動き候補の数を0にリセットすることができる。

【0201】

例D5: ルックアップテーブルは、同じ時間層インデックスを有する1つのスライス/タイル/LCUの行/スライスで連続的に更新されてもよい。

a . 代替的に、ルックアップテーブルは、各S (S ≥ 1) 個のCTU/CTB/CU/CBをコーディング/デコーディングした後、または特定の領域(例えば、8 × 8または16 × 16に等しいサイズ)をコーディング/デコーディングした後にのみ更新されてもよい。

b . 代替的に、1つのルックアップテーブルは、最大許容カウンタに達すると、更新を停止してもよい。

c . 一例において、カウンタは予め規定されてもよい。代替的に、映像パラメータセット(VPS)、シーケンスパラメータセット(PPS)、ピクチャパラメータセット(CPB)、スライスヘッダ、タイルヘッダ、コーディングツリーユニット(CTU)、コーディングツリーブロック(CTB)、コーディングユニット(CU)または予測ユニット(PU)、複数のCTU/CTB/CU/PUをカバーする領域で信号通知される。

【0202】

図28は、映像処理装置2800のブロック図である。装置2800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2800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)受信機等にも実装されてもよい。装置2800は、1つ以上の処理装置2802と、1つ以上のメモリ2804と、映像処理ハードウェア2806と、を含んでもよい。1つまたは複数の処理装置2802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)2804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2806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。

【0203】

図29は、映像デコーディング方法2900の例のフローチャートである。この方法2900は、多数のテーブル(例えばルックアップテーブル; LUT)を維持すること(2902)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、以前コーディングされた映像ブロックから導出された対応する動き情報に関連付けられ、現在の映像ブロックと映像領域における現在の映像ブロックのビットストリーム表現との間で変換を行うこと(2904)と、更新規則に基づいて1つ以上のテーブルを更新すること(2906)と、を含む。

【0204】

図30は、映像復号方法3000の例のフローチャートである。この方法3000は、テーブルのセット(例えばルックアップテーブル; LUT)をチェックすること(300

10

20

30

40

50

2) を含み、各テーブルは、1つ以上の動き候補を含み、各動き候補が動き候補の動き情報に関連付けられており、1つ以上のテーブルに基づいてこの映像ブロックの動き情報を処理すること(3004)と、処理によって生成された映像ブロックに基づいて1つ以上のテーブルを更新すること(3006)と、を含む。

【0205】

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。

【0206】

本明細書に記載された開示されたそして他の実施形態、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、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、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コーディングするように生成される。

【0207】

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として、も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。

【0208】

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処理装置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)によって実行することができ、装置はまた、特定用途のロジック回路として実装することができる。

【0209】

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、並びに任意の種類デジタルコンピュータの任意の1つ以上の処理装置

10

20

30

40

50

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。処理装置およびメモリは、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。

10

【0210】

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征是、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征是、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、特征是、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つ以上の特征是、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、サブコンビネーション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。

20

【0211】

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。

30

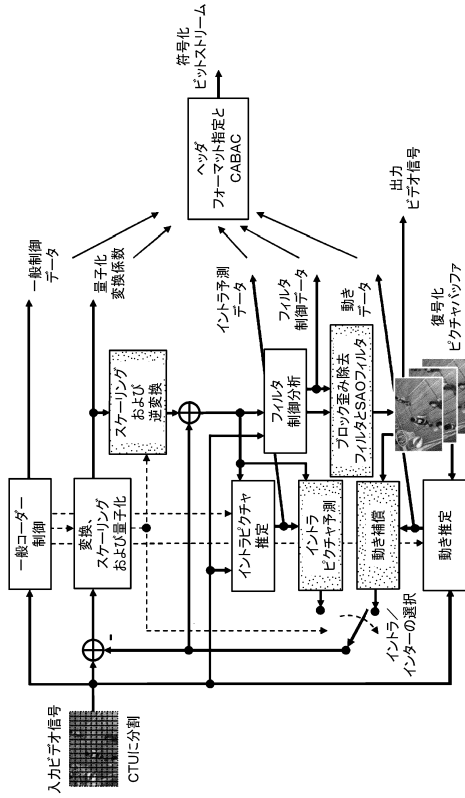
【0212】

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。

40

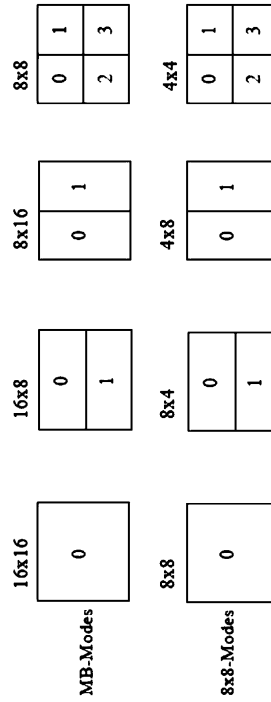
50

【図面】
【図 1】



典型的なHEVCビデオエンコーダ(デコーダモジュール要素はライトグレーの影が付けられている)。

【図 2】



10

20

【図 3】

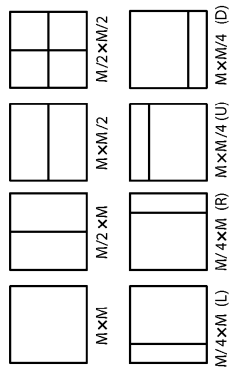
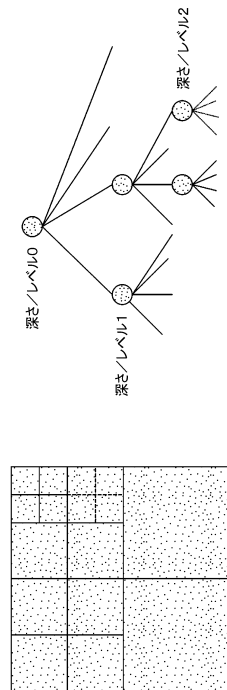


図3特定のサイズ制御に従って、CB各PBIに分類するためのモード。イントラブロック予測CBでは、MxMとM/2xM/2だけがサポートされる。

【図 4】

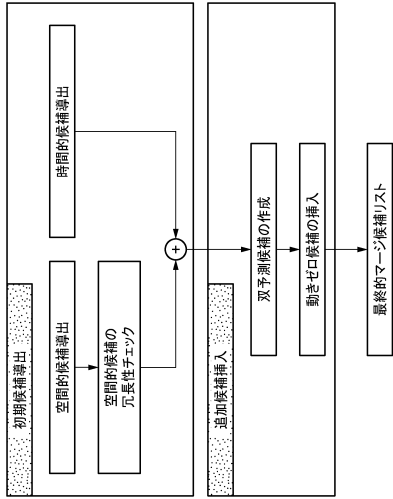


30

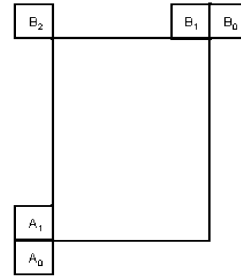
40

50

【図 9】



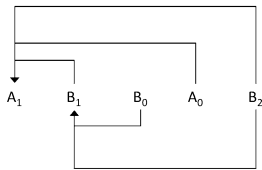
【図 10】



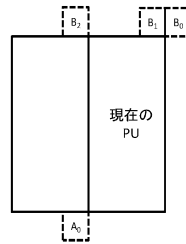
10

20

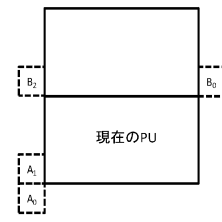
【図 11】



【図 12】



(a) $N \times 2N$ の第2のPU



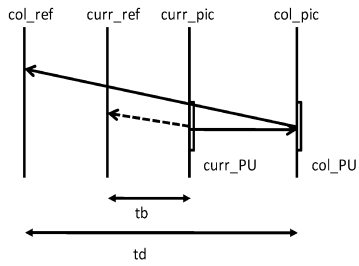
(b) $2N \times N$ の第2のPU

30

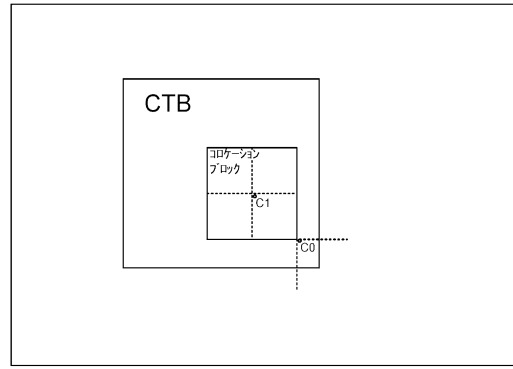
40

50

【図 13】

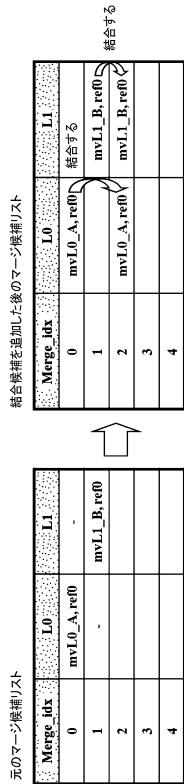


【図 14】

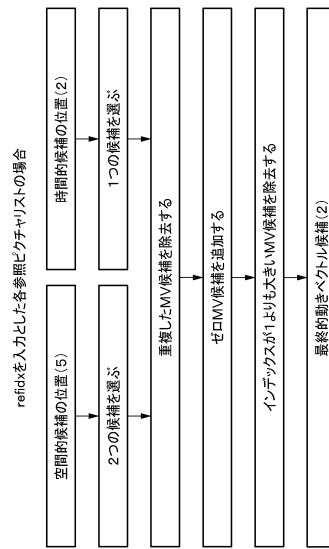


10

【図 15】



【図 16】



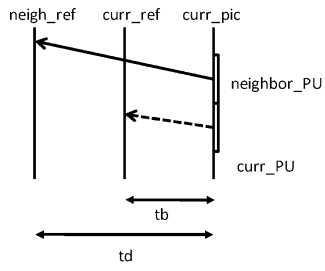
20

30

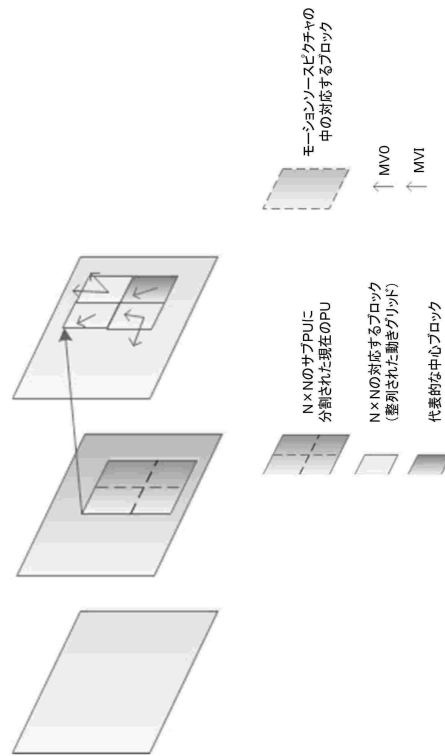
40

50

【図 17】



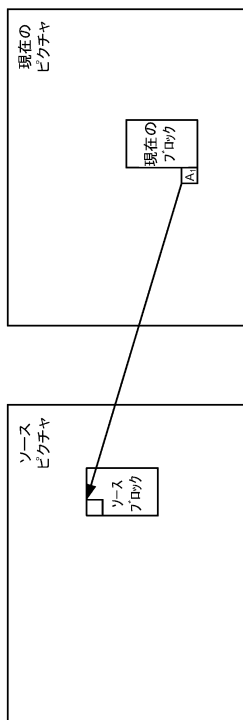
【図 18】



10

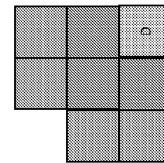
20

【図 19】



30

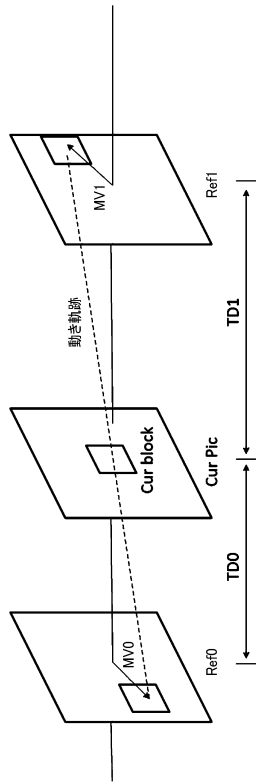
【図 20】



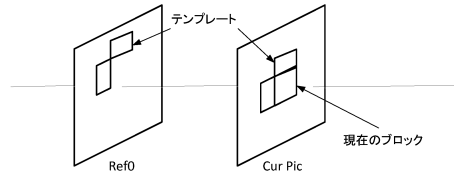
40

50

【図 2 1】



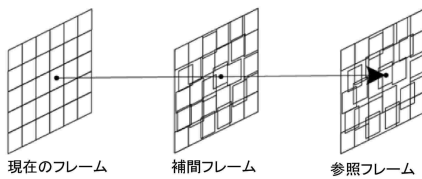
【図 2 2】



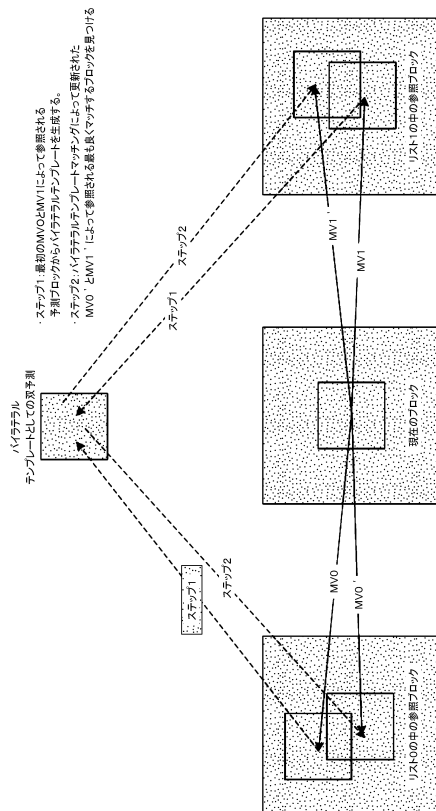
10

20

【図 2 3】



【図 2 4】

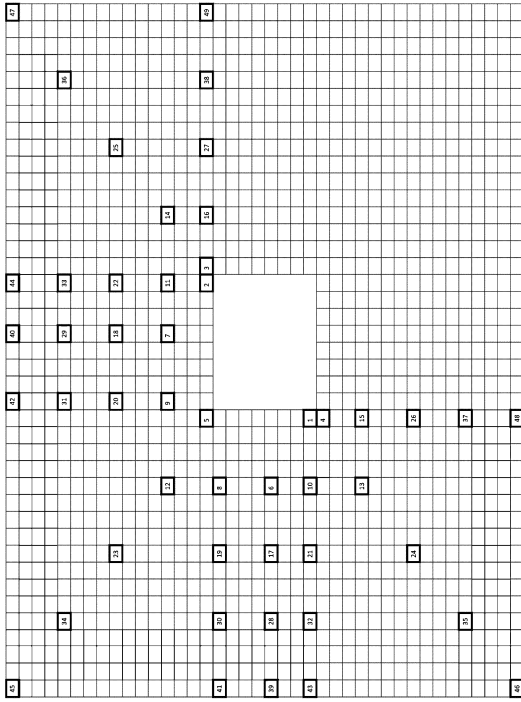


30

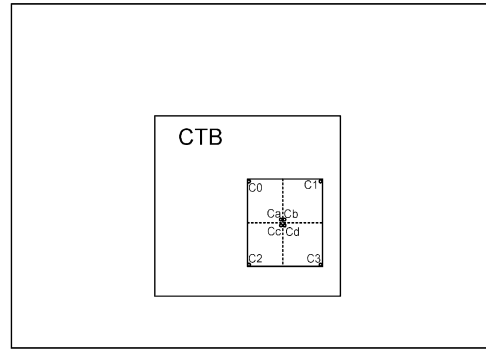
40

50

【図 25】



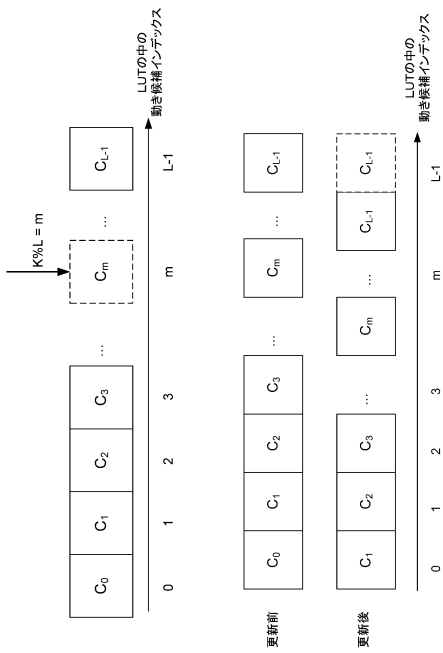
【図 26】



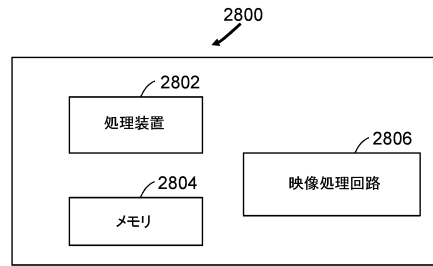
10

20

【図 27】



【図 28】



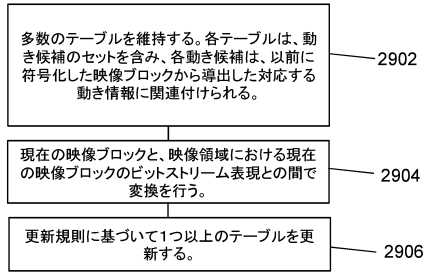
30

40

50

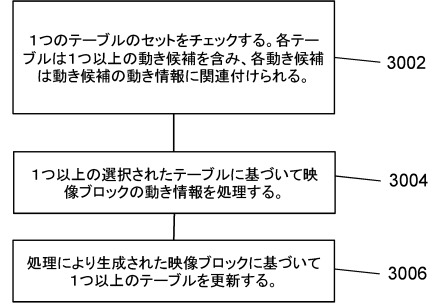
【 図 2 9 】

2900 ↓



【 図 3 0 】

3000 ↓



10

20

30

40

50

フロントページの続き

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・13712655 West Jefferson Boulevard, Sixth Floor, S
uite No. 137 Los Angeles, California 90066 Unit
ed 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 ホンピン

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 ベイジン, ハイディアン 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 西出 隆二

(56)参考文献 国際公開第2018/061522(WO, A1)

(58)調査した分野 (Int.Cl., DB名)

H04N 19/52