

(19)日本国特許庁(JP)

## (12)特許公報(B2)

(11)特許番号  
特許第7514236号  
(P7514236)

(45)発行日 令和6年7月10日(2024.7.10)

(24)登録日 令和6年7月2日(2024.7.2)

(51)国際特許分類

H 0 4 N	19/13 (2014.01)	H 0 4 N	19/13
H 0 4 N	19/52 (2014.01)	H 0 4 N	19/52
H 0 4 N	19/184 (2014.01)	H 0 4 N	19/184
H 0 4 N	19/159 (2014.01)	H 0 4 N	19/159

F I

請求項の数 9 (全83頁)

(21)出願番号 特願2021-528449(P2021-528449)  
 (86)(22)出願日 令和1年11月19日(2019.11.19)  
 (65)公表番号 特表2022-511744(P2022-511744)  
 A)  
 (43)公表日 令和4年2月1日(2022.2.1)  
 (86)国際出願番号 PCT/EP2019/081827  
 (87)国際公開番号 WO2020/126262  
 (87)国際公開日 令和2年6月25日(2020.6.25)  
 審査請求日 令和4年11月16日(2022.11.16)  
 (31)優先権主張番号 1820902.3  
 (32)優先日 平成30年12月20日(2018.12.20)  
 (33)優先権主張国・地域又は機関  
 英国(GB)

(73)特許権者 000001007  
 キヤノン株式会社  
 東京都大田区下丸子3丁目30番2号  
 (74)代理人 110003281  
 弁理士法人大塚国際特許事務所  
 ラロシュ、ギローム  
 フランス国 レンヌ・アタラント、セデ  
 ックス セッソン・セヴィニエ 3551  
 7, リュ ドウ ラ トゥッシュランペ  
 ル キヤノン リサーチ センター フラン  
 ス エス・エー・エス 内  
 ジスケ、クリストフ  
 フランス国 レンヌ・アタラント、セデ  
 ックス セッソン・セヴィニエ 3551  
 7, リュ ドウ ラ トゥッシュランペ  
 最終頁に続く

(54)【発明の名称】 ビデオ符号化及び復号

## (57)【特許請求の範囲】

## 【請求項1】

動き情報予測子に関する情報を符号化する方法であって、  
 複数の動き情報予測子候補のうちの1つを選択することと、  
 C A B A C 符号化を使用して、選択された動き情報予測子候補を特定するための情報を  
 符号化することと  
 を含み、

前記情報の1番目のビットの前記C A B A C 符号化について、プロックにおける第1領域に  
 関連する第1プロック予測子と、前記プロックにおける領域であって前記第1領域と  
 異なる領域である第2領域に関連する第2プロック予測子と、からプロック予測子が取得  
 可能であるマージモードは、インター予測モードの他のマージモードと同じコンテキスト  
 変数を使用し、

前記情報の前記1番目のビットを除く前記情報のすべてのビットが、バイパス符号化される  
 ことを特徴とする方法。

## 【請求項2】

動き情報予測子に関する情報を復号する方法であって、  
 C A B A C 復号を使用して、複数の動き情報予測子候補のうちの1つを特定するための  
 情報を復号することと、  
 復号された情報を用いて、複数の動き情報予測子候補のうちの1つを特定することと  
 を含み、

前記情報の1番目のビットの前記C A B A C復号について、ブロックにおける第1領域に関連する第1ブロック予測子と、前記ブロックにおける領域であって前記第1領域と異なる領域である第2領域に関連する第2ブロック予測子と、からブロック予測子が取得可能であるマージモードは、インター予測モードの他のマージモードと同じコンテキスト変数を使用し、

前記情報の前記1番目のビットを除く前記情報のすべてのビットが、バイパス復号されることを特徴とする方法。

**【請求項3】**

前記情報は、動き情報予測子インデックスであることを特徴とする請求項1または2に記載の方法。

10

**【請求項4】**

動き情報予測子候補が、動きベクトルを取得するための情報を含むことを特徴とする請求項1または2に記載の方法。

**【請求項5】**

ビットストリームから、前記複数の動き情報予測子候補に含まれ得る動き情報予測子候補の最大数を決定するための情報を取得することを特徴とする請求項1または2に記載の方法。

**【請求項6】**

動き情報予測子に関する情報を符号化する装置であって、  
複数の動き情報予測子候補のうちの1つを選択する手段と、  
C A B A C符号化を使用して、選択された動き情報予測子候補を特定するための情報を符号化する手段と  
を含み、

20

前記情報の1番目のビットの前記C A B A C符号化について、ブロックにおける第1領域に関連する第1ブロック予測子と、前記ブロックにおける領域であって前記第1領域と異なる領域である第2領域に関連する第2ブロック予測子と、からブロック予測子が取得可能であるマージモードは、インター予測モードの他のマージモードと同じコンテキスト変数を使用し、

前記情報の前記1番目のビットを除く前記情報のすべてのビットが、バイパス符号化されることを特徴とする装置。

30

**【請求項7】**

動き情報予測子に関する情報を復号する装置であって、  
C A B A C復号を使用して、複数の動き情報予測子候補のうちの1つを特定するための情報を復号する手段と、  
復号された情報を用いて、複数の動き情報予測子候補のうちの1つを特定する手段と  
を含み、

前記情報の1番目のビットの前記C A B A C復号について、ブロックにおける第1領域に関連する第1ブロック予測子と、前記ブロックにおける領域であって前記第1領域と異なる領域である第2領域に関連する第2ブロック予測子と、からブロック予測子が取得可能であるマージモードは、インター予測モードの他のマージモードと同じコンテキスト変数を使用し、

40

前記情報の前記1番目のビットを除く前記情報のすべてのビットが、バイパス復号されることを特徴とする装置。

**【請求項8】**

コンピュータに、請求項1に記載の方法を実行させるためのコンピュータプログラム。

**【請求項9】**

コンピュータに、請求項2に記載の方法を実行させるためのコンピュータプログラム。

**【発明の詳細な説明】**

**【技術分野】**

**【0001】**

50

本発明は、ビデオ符号化及び復号に関する。

#### 【背景技術】

##### 【0002】

最近、MPEGとITU-T Study Group 16のVCEGによって形成された共同チームであるJVET (Joint Video Experts Team) は、VVC (Versatile Video Coding) と呼ばれる新しいビデオ符号化規格の研究を開始した。VVCの目標は、既存のHEVC規格（すなわち、典型的には以前の2倍）を超える圧縮性能の著しい改善を提供し、2020年に完了することである。主なターゲットアプリケーションおよびサービスは、360度および高ダイナミックレンジ(HDR)ビデオを含むが、限定されない。総じて、JVETは、独立した試験所が実施した正式な主観的試験を用いて、32団体からのレスポンスを評価した。いくつかの提案は、HEVCを使用する場合と比較して、典型的には40%以上の圧縮効率ゲインを実証した。超高精細(UHD)ビデオ試験材料について特に有効性を示した。したがって、圧縮効率の向上は、最終標準の目標とする50%をはるかに超えることが予想される。

10

##### 【0003】

JVET探索モデル(JEM)は、すべてのHEVCツールを使用する。HEVCには存在しないさらなるツールは、動き補償を適用する場合に「アフィン動きモード」を使用することである。HEVCにおける動き補償は並進に限定されるが、実際には例えばズームイン/アウト、回転、透視動作、及び他の不規則な動作のような多くの種類の動きがある。アフィン動きモードを利用する場合、より複雑な変換がブロックに適用され、そのような動きの形成をより正確に予測しようと試みる。したがって、良好な符号化効率を達成しながら、複雑さを減らしながら、アフィン動きモードを使用することができることが望ましい。

20

##### 【0004】

HEVCに存在しない別のツールは、代替時間的動きベクトル予測(ATMVP)を使用することである。代替時間的動きベクトル予測(ATMVP)は、特定の動き補償である。時間的参照フレームからの現在のブロックについて1つの動き情報のみを考慮する代わりに、各コロケートされたブロックの各動き情報が考慮される。したがって、この時間的動きベクトル予測は、各サブブロックの関連する動き情報を用いた現在のブロックのセグメンテーションを与える。現在のVTM(VVCテストモデル)参照ソフトウェアでは、ATMVPがマージ候補のリストに挿入されたマージ候補としてシグナリングされる。SFSレベルでATMVPがイネーブルされると、マージ候補の最大数は1だけ増加される。従って、このモードがディスエーブルされているときから、5個の代わりに6個の候補が考慮される。

30

##### 【0005】

これらの、および後述する他のツールは候補のリストの中から（例えば、マージモード符号化と共に使用するためのマージ候補のリストから）どの候補が選択されたかを知るために使用されるインデックス（例えば、マージインデックス）またはフラグの符号化の符号化効率および複雑さに関する問題を引き起こしている。

40

#### 【発明の概要】

##### 【0006】

したがって、前述の問題の少なくとも1つに対する解決策が望ましい。

##### 【0007】

本発明の第1の態様によれば、動きベクトル予測子インデックスを符号化する方法であつて、

ATMVP候補を含む動きベクトル予測子候補のリストを生成し、

前記リスト内の動きベクトル予測子候補のうちの1つを選択し、

CABAC符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックス（マージインデックス）を生成し、動きベクトル予測子インデックスの1つまたは複数のビットは、バイパスCABAC符号化される

50

ことを特徴とする方法が提供される。

【0008】

一実施形態では、動きベクトル予測子インデックスの最初のビットを除くすべてのビットがバイパスC A B A C符号化される。

【0009】

本発明の第2の態様によれば、動きベクトル予測子インデックスを復号する方法であつて、

A T M V P候補を含む動きベクトル予測子候補のリストを生成し、

C A B A C復号を使用して動きベクトル予測子インデックスを復号し、動きベクトル予測子インデックスの1つまたは複数のビットは、バイパスC A B A C復号され、

復号された動きベクトル予測子インデックスを使用して、前記リスト内の動きベクトル予測子候補のうちの1つを識別する

ことを特徴とする方法が提供される。

【0010】

一実施形態では、動きベクトル予測子インデックスの最初のビットを除くすべてのビットがバイパスC A B A C復号される。

【0011】

本発明の第3の態様によれば、動きベクトル予測子インデックスを符号化する装置であつて、

A T M V P候補を含む動きベクトル予測子候補のリストを生成する手段と、

前記リスト内の動きベクトル予測子候補のうちの1つを選択する手段と、

C A B A C符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックス(マージインデックス)を生成する手段であつて、動きベクトル予測子インデックスの1つまたは複数のビットがバイパスC A B A C符号化される手段と

を備えることを特徴とする装置が提供される。

【0012】

本発明の第4の態様によれば、動きベクトル予測子インデックスを復号する装置であつて、

A T M V P候補を含む動きベクトル予測子候補のリストを生成する手段と、

C A B A C復号を使用して動きベクトル予測子インデックスを復号する手段であつて、動きベクトル予測子インデックスの1つまたは複数のビットがバイパスC A B A C復号される手段と、

復号された動きベクトル予測子インデックスを使用して、前記リスト内の動きベクトル予測子候補のうちの1つを識別する手段と

を備えることを特徴とする装置が提供される。

【0013】

本発明の第5の態様によれば、動きベクトル予測子インデックスを符号化する方法であつて、

動きベクトル予測子候補のリストを生成し、

前記リスト内の動きベクトル予測子候補のうちの1つを選択し、

C A B A C符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成し、動きベクトル予測子インデックスの2つ以上のビットが同じコンテキストを共有する

ことを特徴とする方法が提供される。

【0014】

一実施形態では、動きベクトル予測子インデックスのすべてのビットが同じコンテキストを共有する。

【0015】

本発明の第6の態様によれば、動きベクトル予測子インデックスを復号する方法であつて、

10

20

30

40

50

動きベクトル予測子候補のリストを生成し、  
 C A B A C 復号を使用して、動きベクトル予測子インデックスを復号し、動きベクトル  
 予測子インデックスの 2 つ以上のビットが同じコンテキストを共有し、  
 復号された動きベクトル予測子インデックスを使用して、前記リスト内の動きベクトル  
 予測子候補のうちの 1 つを識別する  
 ことを特徴とする方法が提供される。

## 【0016】

一実施形態では、動きベクトル予測子インデックスのすべてのビットが同じコンテキス  
 トを共有する。

## 【0017】

本発明の第 7 の態様によれば、動きベクトル予測子インデックスを符号化する装置であ  
 って、

動きベクトル予測子候補のリストを生成する手段と、  
 前記リスト内の動きベクトル予測子候補のうちの 1 つを選択する手段と、  
 C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測  
 子インデックスを生成する手段であって、動きベクトル予測子インデックスの 2 つ以上の  
 ビットが同じコンテキストを共有する手段と  
 を備えることを特徴とする装置が提供される。

## 【0018】

本発明の第 8 の態様によれば、動きベクトル予測子インデックスを復号する装置であ  
 って、

動きベクトル予測子候補のリストを生成する手段と、  
 C A B A C 復号を使用して、動きベクトル予測子インデックスを復号する手段であって  
 、動きベクトル予測子インデックスの 2 つ以上のビットが同じコンテキストを共有する手  
 段と、  
 復号された動きベクトル予測子インデックスを使用して、前記リスト内の動きベクトル  
 予測子候補のうちの 1 つを識別する手段と  
 を備えることを特徴とする装置が提供される。

## 【0019】

本発明の第 9 の態様によれば、動きベクトル予測子インデックスを符号化する方法であ  
 って、

動きベクトル予測子候補のリストを生成し、  
 前記リスト内の動きベクトル予測子候補のうちの 1 つを選択し、  
 C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測  
 子インデックスを生成し、現在のブロックの動きベクトル予測子インデックスの少なくとも  
 1 つのビットのコンテキスト変数が、現在のブロックに隣接する少なくとも 1 つのブロ  
 ックの動きベクトル予測子インデックスに依存する  
 ことを特徴とする方法が提供される。

## 【0020】

一実施形態では、動きベクトル予測子インデックスの少なくとも 1 つのビットに対する  
 コンテキスト変数が少なくとも 2 つの隣接ブロックのそれぞれの動きベクトル予測子イン  
 デックスに依存する。

## 【0021】

別の実施形態では、動きベクトル予測子インデックスの少なくとも 1 つのビットのコン  
 テキスト変数が、現在のブロックの左側の左隣接ブロックの動きベクトル予測子インデッ  
 クスと、現在のブロックの上側の上隣接ブロックの動きベクトル予測子インデックスと、  
 に依存する。

## 【0022】

別の実施形態では、左隣接ブロックは A 2 であり、上隣接ブロックは B 3 である。

## 【0023】

10

20

30

40

50

別の実施形態では、左隣接ブロックは A 1 であり、上隣接ブロックは B 1 である。

【 0 0 2 4 】

別の実施形態では、コンテキスト変数が 3 つの異なる可能な値を有する。

【 0 0 2 5 】

別の実施形態は、少なくとも 1 つの隣接ブロックの動きベクトル予測子インデックスを、現在のブロックの動きベクトル予測子インデックスのインデックス値と比較し、比較結果に応じて前記コンテキスト変数を設定することを含む。

【 0 0 2 6 】

別の実施形態は、少なくとも 1 つの隣接ブロックの動きベクトル予測子インデックスを、現在のブロックの動きベクトル予測子インデックス内の前記または 1 つの前記ビットのビット位置を表すパラメータと比較し、比較結果に応じて前記コンテキスト変数を設定することを含む。

10

【 0 0 2 7 】

さらに別の実施形態は、第 1 比較を行い、第 1 隣接ブロックの動きベクトル予測子インデックスを、現在のブロックの動きベクトル予測子インデックス内の前記または 1 つの前記ビットのビット位置を表すパラメータと比較し、第 2 比較を行い、第 2 隣接ブロックの動きベクトル予測子インデックスを、前記パラメータと比較し、第 1 および第 2 比較の結果に応じて前記コンテキスト変数を設定することを含む。

【 0 0 2 8 】

本発明の第 10 の態様によれば、動きベクトル予測子インデックスを復号する方法であつて、

20

動きベクトル予測子候補のリストを生成し、

C A B A C 復号を使用して、動きベクトル予測子インデックスを復号し、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも 1 つのビットのコンテキスト変数は、現在のブロックに隣接する少なくとも 1 つのブロックの動きベクトル予測子インデックスに依存し、

復号された動きベクトル予測子インデックスを使用して、前記リスト内の動きベクトル予測子候補のうちの 1 つを識別する

ことを特徴とする方法が提供される。

【 0 0 2 9 】

30

一実施形態では、動きベクトル予測子インデックスの少なくとも 1 つのビットに対するコンテキスト変数が、少なくとも 2 つの隣接ブロックのそれぞれの動きベクトル予測子インデックスに依存する。

【 0 0 3 0 】

別の実施形態では、動きベクトル予測子インデックスの少なくとも 1 つのビットのコンテキスト変数が、現在のブロックの左側の左隣接ブロックの動きベクトル予測子インデックスと、現在のブロックの上側の上隣接ブロックの動きベクトル予測子インデックスとに依存する。

【 0 0 3 1 】

別の実施形態では、左隣接ブロックは A 2 であり、上隣接ブロックは B 3 である。

40

【 0 0 3 2 】

別の実施形態では、左隣接ブロックは A 1 であり、上隣接ブロックは B 1 である。

【 0 0 3 3 】

別の実施形態では、コンテキスト変数が 3 つの異なる可能な値を有する。

【 0 0 3 4 】

別の実施形態は、少なくとも 1 つの隣接ブロックの動きベクトル予測子インデックスを、現在のブロックの動きベクトル予測子インデックスのインデックス値と比較し、比較結果に応じて前記コンテキスト変数を設定することを含む。

【 0 0 3 5 】

別の実施形態は、少なくとも 1 つの隣接ブロックの動きベクトル予測子インデックスを

50

、現在のブロックの動きベクトル予測子インデックス内の前記または1つの前記ビットのビット位置を表すパラメータと比較し、比較結果に応じて前記コンテキスト変数を設定することを含む。

【0036】

さらに別の実施形態は、第1比較を行い、第1隣接ブロックの動きベクトル予測子インデックスを、現在のブロックの動きベクトル予測子インデックス内の前記または1つの前記ビットのビット位置を表すパラメータと比較し、第2比較を行い、第2隣接ブロックの動きベクトル予測子インデックスを、前記パラメータと比較し、第1および第2比較の結果に応じて前記コンテキスト変数を設定することを含む。

【0037】

本発明の第11の態様によれば、動きベクトル予測子インデックスを符号化する装置であって、

動きベクトル予測子候補のリストを生成する手段と、

前記リスト内の動きベクトル予測子候補のうちの1つを選択する手段と、

C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成する手段であって、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットのコンテキスト変数が、現在のブロックに隣接する少なくとも1つのブロックの動きベクトル予測子インデックスに依存する手段と

を備えることを特徴とする装置が提供される。

【0038】

本発明の第12の態様によれば、動きベクトル予測子インデックスを復号する装置であって、

動きベクトル予測子候補のリストを生成する手段と、

C A B A C 復号を使用して、動きベクトル予測子インデックスを復号する手段であって、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットに対するコンテキスト変数は、現在のブロックに隣接する少なくとも1つのブロックの動きベクトル予測子インデックスに依存する手段と、

復号された動きベクトル予測子インデックスを使用して、前記リスト内の動きベクトル予測子候補のうちの1つを識別する手段と

を備えることを特徴とする装置が提供される。

【0039】

本発明の第13の態様によれば、動きベクトル予測子インデックスを符号化する方法であって、

動きベクトル予測子候補のリストを生成し、

前記リスト内の動きベクトル予測子候補のうちの1つを選択し、

C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成し、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットのコンテキスト変数が、前記現在のブロックのスキップフラグに依存する

ことを特徴とする方法が提供される。

【0040】

本発明の第14の態様によれば、動きベクトル予測子インデックスを符号化する方法であって、

動きベクトル予測子候補のリストを生成し、

前記リスト内の動きベクトル予測子候補のうちの1つを選択し、

C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成し、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットのコンテキスト変数は、動きベクトル予測子インデックスの復号前に利用可能な前記現在のブロックの別のパラメータまたはシンタックス要素に依存する

ことを特徴とする方法が提供される。

10

20

30

40

50

**【 0 0 4 1 】**

本発明の第 15 の態様によれば、動きベクトル予測子インデックスを符号化する方法であって、

動きベクトル予測子候補のリストを生成し、

前記リスト内の動きベクトル予測子候補のうちの 1 つを選択し、

C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成し、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも 1 つのビットのコンテキスト変数が、現在のブロック内の動きの複雑さのインジケータである前記現在のブロックの別のパラメータまたはシンタックス要素に依存することを特徴とする方法が提供される。

10

**【 0 0 4 2 】**

本発明の第 16 の態様によれば、動きベクトル予測子インデックスを復号する方法であって、

動きベクトル予測子候補のリストを生成し、

C A B A C 復号を使用して、動きベクトル予測子インデックスを復号し、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも 1 つのビットに対するコンテキスト変数は、前記現在のブロックのスキップフラグに依存し、

復号された動きベクトル予測子インデックスを使用して、前記リスト内の動きベクトル予測子候補のうちの 1 つを識別する

ことを特徴とする方法が提供される。

20

**【 0 0 4 3 】**

本発明の第 17 の態様によれば、動きベクトル予測子インデックスを復号する方法であって、

動きベクトル予測子候補のリストを生成し、

C A B A C 復号を使用して、動きベクトル予測子インデックスを復号し、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも 1 つのビットのコンテキスト変数は、動きベクトル予測子インデックスの復号前に利用可能である前記現在のブロックの別のパラメータまたはシンタックス要素に依存し、

復号された動きベクトル予測子インデックスを使用して、前記リスト内の動きベクトル予測子候補のうちの 1 つを識別する

30

ことを特徴とする方法が提供される。

**【 0 0 4 4 】**

本発明の第 18 の態様によれば、動きベクトル予測子インデックスを復号する方法であって、

動きベクトル予測子候補のリストを生成し、

C A B A C 復号を使用して、動きベクトル予測子インデックスを復号し、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも 1 つのビットのコンテキスト変数は、現在のブロックにおける動きの複雑さのインジケータである前記現在のブロックの別のパラメータまたはシンタックス要素に依存し、

復号された動きベクトル予測子インデックスを使用して、前記リスト中の動きベクトル予測子候補のうちの 1 つを識別する

40

ことを特徴とする方法が提供される。

**【 0 0 4 5 】**

本発明の第 19 の態様によれば、動きベクトル予測子インデックスを符号化する装置であって、

動きベクトル予測子候補のリストを生成する手段と、

前記リスト内の動きベクトル予測子候補のうちの 1 つを選択する手段と、

C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成する手段であって、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも 1 つのビットに対するコンテキスト変数が、前記現在のブロック

50

クのスキップフラグに依存する手段と  
を備えることを特徴とする装置が提供される。

【0046】

本発明の第20の態様によれば、動きベクトル予測子インデックスを符号化する装置であって、

動きベクトル予測子候補のリストを生成する手段と、

前記リスト内の動きベクトル予測子候補のうちの1つを選択する手段と、

C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成する手段であって、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットのコンテキスト変数は、動きベクトル予測子インデックスの復号前に利用可能である前記現在のブロックの別のパラメータまたはシンタックス要素に依存する手段と

を備えることを特徴とする装置が提供される。

【0047】

本発明の第21の態様によれば、動きベクトル予測子インデックスを符号化する装置であって、

動きベクトル予測子候補のリストを生成する手段と、

前記リスト内の動きベクトル予測子候補のうちの1つを選択する手段と、

C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成する手段であって、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットのコンテキスト変数は、現在のブロック内の動きの複雑さのインジケータである前記現在のブロックの別のパラメータまたはシンタックス要素に依存する手段と

を備えることを特徴とする装置が提供される。

【0048】

本発明の第22の態様によれば、動きベクトル予測子インデックスを復号する装置であって、

動きベクトル予測子候補のリストを生成する手段と、

C A B A C 復号を使用して、動きベクトル予測子インデックスを復号する手段であって、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットに対するコンテキスト変数は、前記現在のブロックのスキップフラグに依存する手段と、

復号された動きベクトル予測子インデックスを使用して、前記リスト内の動きベクトル予測子候補のうちの1つを識別する手段と

を備えることを特徴とする装置が提供される。

【0049】

本発明の第23の態様によれば、動きベクトル予測子インデックスを復号する装置であって、

動きベクトル予測子候補のリストを生成する手段と、

C A B A C 復号を使用して、動きベクトル予測子インデックスを復号する手段であって、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットに対するコンテキスト変数は、動きベクトル予測子インデックスの復号前に利用可能である前記現在のブロックの別のパラメータまたはシンタックス要素に依存する手段と、

復号された動きベクトル予測子インデックスを使用して、前記リスト内の動きベクトル予測子候補のうちの1つを識別する手段と

を備えることを特徴とする装置が提供される。

【0050】

本発明の第24の態様によれば、動きベクトル予測子インデックスを復号する装置であって、

動きベクトル予測子候補のリストを生成する手段と、

C A B A C 復号を使用して、動きベクトル予測子インデックスを復号する手段であって

10

20

30

40

50

、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットに対するコンテキスト変数は、現在のブロックにおける動きの複雑さのインジケータである前記現在のブロックの別のパラメータまたはシンタックス要素に依存する手段と、

復号された動きベクトル予測子インデックスを使用して、前記リスト内の動きベクトル予測子候補のうちの1つを選択する手段と

を備えることを特徴とする装置が提供される。

#### 【0051】

本発明の第25の態様によれば、動き情報予測子に関する情報を符号化する方法であって、複数の動き情報予測子候補のうちの1つを選択し、CABAC符号化を使用して、選択された動き情報予測子候補を識別するための情報を符号化し、CABAC符号化は、前記情報の少なくとも1つのビットについて、トライアングルマージモードまたは動きベクトル差分(MMVD)マージモードのマージうちの一方または両方が使用される場合に、別のインター予測モードに使用される同じコンテキスト変数を使用する、方法が提供される。

10

#### 【0052】

本発明の第26の態様によれば、動き情報予測子に関する情報を復号する方法であって、CABAC復号を使用して複数の動き情報予測子候補のうちの1つを選択し、CABAC復号は、前記情報の少なくとも1つのビットについて、トライアングルマージモードまたは動きベクトル差分(MMVD)マージモードのマージうちの1つまたは両方が使用される場合に、別のインター予測モードに使用される同じコンテキスト変数を使用する、方法が提供される。

20

#### 【0053】

本発明の第25の態様または第26の態様について、以下の特徴が、その実施形態に従って提供されてもよい。

#### 【0054】

適切には、情報の1番目のビットを除くすべてのビットがバイパスCABAC符号化されるか、またはバイパスCABAC復号される。適切には、1番目のビットがCABAC符号化またはCABAC復号される。好適には、別のインター予測モードが、マージモードまたはアフィンマージモードの一方または両方を含む。好適には、別のインター予測モードが、MHII (Multi-Hypothesis Intra Inter) マージモードを含む。好適には、別のインター予測モードのための複数の動き情報予測子候補がATMVP候補を含む。適切には、CABAC符号化またはCABAC復号がトライアングルマージモードが使用され、MMVDマージモードが使用されるとき、両方に対して同じコンテキスト変数を使用することを含む。好適には、前記情報の少なくとも1つのビットが、スキップモードが使用されるときに、CABAC符号化またはCABAC復号される。好適には、スキップモードが、マージスキップモード、アフィンマージスキップモード、トライアングルマージスキップモード、またはモーションベクトル差分(MMVD)マージスキップモードのマージうちの1つまたは複数を含む。

30

#### 【0055】

本発明の第27の態様によれば、動き情報予測子に関する情報を符号化する方法であって、複数の動き情報予測子候補のうちの1つを選択し、選択された動き情報予測子候補を識別するための情報を符号化し、情報を符号化することは、トライアングルマージモードまたは動きベクトル差分(MMVD)マージモードのマージうちの1つまたは両方が使用されるときに、前記情報の少なくとも1つのビットをバイパスCABAC符号化する、方法が提供される。

40

#### 【0056】

本発明の第28の態様によれば、動き情報予測子に関する情報を復号する方法であって、複数の動き情報予測子候補のうちの1つを選択し、情報を復号し、復号された情報を使用して複数の動き情報予測子候補のうちの1つを選択し、情報を復号することは、

50

トライアングルマージモードまたは動きベクトル差分 (MMVD) マージモードのマージのうちの 1 つまたは両方が使用される場合に、前記情報の少なくとも 1 つのビットをバイパス C A B A C 復号する、方法が提供される。

【0057】

本発明の第 27 の態様または第 28 の態様について、以下の特徴が、その実施形態に従って提供されてもよい。

【0058】

適切には、情報の 1 番目のビットを除くすべてのビットがバイパス C A B A C 符号化されるか、またはバイパス C A B A C 復号される。適切には、1 番目のビットが C A B A C 符号化または C A B A C 復号される。好適には、前記情報の全てのビットがトライアングルマージモード又は MMVD マージモードの一方又は両方が使用されるときに、バイパス C A B A C 符号化又はバイパス C A B A C 復号される。好適には、前記情報のすべてのビットがバイパス C A B A C 符号化されるか、またはバイパス C A B A C 復号される。

10

【0059】

適切には、前記情報の少なくとも 1 つのビットが、アフィンマージモードが使用されるときに、C A B A C 符号化または C A B A C 復号される。適切には、前記情報のすべてのビットが、アフィンマージモードが使用される場合を除いて、バイパス C A B A C 符号化されるか、またはバイパス C A B A C 復号される。

【0060】

好適には、前記情報の少なくとも 1 つのビットが、マージモードまたは MHII (Multi-Hypothesis Intra Inter) マージモードの一方または両方が使用されるときに、C A B A C 符号化または C A B A C 復号される。好適には、前記情報の全てのビットが、マージモード又は MHII (Multi-Hypothesis Intra Inter) マージモードの一方又は両方が使用される場合を除いて、バイパス C A B A C 符号化又はバイパス C A B A C 復号される。

20

【0061】

好適には、前記情報の少なくとも 1 つのビットが、複数の動き情報予測子候補が A TM VP 候補を含む場合に C A B A C 符号化または C A B A C 復号される。好適には、前記情報の全てのビットが、複数の動き情報予測子候補が A TM VP 候補を含む場合を除いて、バイパス C A B A C 符号化またはバイパス C A B A C 復号される。

【0062】

30

好適には、前記情報の少なくとも 1 つのビットが、スキップモードが使用されるときに、C A B A C 符号化または C A B A C 復号される。好適には、前記情報のすべてのビットが、スキップモードが使用される場合を除いて、バイパス C A B A C 符号化されるか、またはバイパス C A B A C 復号される。好適には、スキップモードが、マージスキップモード、アフィンマージスキップモード、トライアングルマージスキップモード、または動きベクトル差分 (MMVD) マージスキップモードのマージのうちの 1 つまたは複数を含む。

【0063】

本発明の第 25 の態様、第 26 の態様、第 27 の態様、または第 28 の態様について、以下の特徴が、その実施形態に従って提供され得る。

【0064】

40

好適には、少なくとも 1 つのビットが、前記情報の 1 番目のビットを含む。好適には、前記情報が、動き情報予測インデックスまたはフラグを含む。好適には、動き情報予測子候補が、動きベクトルを取得するための情報を含む。

【0065】

本発明の第 25 の態様または第 27 の態様について、以下の特徴が、その実施形態に従って提供されてもよい。

【0066】

好適には、本方法が、ビットストリームにおいて、トライアングルマージモード、MMVD マージモード、マージモード、アフィンマージモード、または MHII (Multi-Hypothesis Intra Inter) マージモードのうちの 1 つの使用を示すための情報をさらに含む。好

50

適には、本方法が、ビットストリームにおいて、複数の動き情報予測子候補に含まれ得る動き情報予測子候補の最大個数を決定するための情報をさらに含む。

【 0 0 6 7 】

本発明の第 2 6 の態様または第 2 8 の態様について、以下の特徴が、その実施形態に従つて提供されてもよい。

【 0 0 6 8 】

適切には、本方法が、ビットストリームから、トライアングルマージモード、M M V D マージモード、マージモード、アフィンマージモード、またはM H I I ( Multi-Hypothesis Intra Inter ) マージモードのうちの 1 つの使用を示すための情報を取得することをさらに含む。好適には、本方法が、ビットストリームから、複数の動き情報予測子候補に含まれ得る動き情報予測子候補の最大個数を決定するための情報を取得することをさらに含む。

10

【 0 0 6 9 】

本発明の第 2 9 の態様によれば、動き情報予測子に関する情報を符号化する装置であつて、複数の動き情報予測子候補のうちの 1 つを選択する手段と、C A B A C 符号化を使用して、選択された動き情報予測子候補を識別するための情報を符号化する手段と、を備え、C A B A C 符号化は、前記情報の少なくとも 1 つのビットについて、トライアングルマージモードまたは動きベクトル差分 ( M M V D ) マージモードのマージの一方または両方が使用される場合に、別のインター予測モードに使用される同じコンテキスト変数を使用する、装置が提供される。適切には、装置が、本発明の第 2 5 の態様または第 2 7 の態様による動き情報予測子に関する情報を符号化する方法を実行するための手段を備える。

20

【 0 0 7 0 】

本発明の第 3 0 の態様によれば、動き情報予測子に関する情報を符号化するための装置であつて、複数の動き情報予測子候補のうちの 1 つを選択する手段と、選択された動き情報予測子候補を識別するための情報を符号化する手段とを備え、情報を符号化することは、トライアングルマージモードまたは動きベクトル差分 ( M M V D ) マージモードのマージのうちの 1 つまたは両方が使用されるときに、前記情報の少なくとも 1 つのビットをバイパス C A B A C 符号化することを備える、装置が提供される。適切には、装置が、本発明の第 2 5 の態様または第 2 7 の態様による動き情報予測子に関する情報を符号化する方法を実行する手段を備える。

30

【 0 0 7 1 】

本発明の第 3 1 の態様によれば、動き情報予測子に関する情報を復号する装置であつて、C A B A C 復号を使用して、複数の動き情報予測子候補のうちの 1 つを識別するための情報を復号する手段と、復号された情報を使用して、複数の動き情報予測子候補のうちの 1 つを選択する手段とを備え、C A B A C 復号は、前記情報の少なくとも 1 つのビットについて、トライアングルマージモードまたは動きベクトル差分 ( M M V D ) マージマージモードのマージのうちの 1 つまたは両方が使用される場合に、別のインター予測モードに使用される同じコンテキスト変数を使用することを備える、装置が提供される。適切には、装置が、本発明の第 2 6 の態様または第 2 8 の態様による動き情報予測子に関する情報を復号する方法を実行する手段を備える。

30

【 0 0 7 2 】

本発明の第 3 2 の態様によれば、動き情報予測子に関する情報を復号する装置であつて、複数の動き情報予測子候補のうちの 1 つを識別するための情報を復号する手段と、復号された情報を使用して、複数の動き情報予測子候補のうちの 1 つを選択する手段とを備え、情報を復号することは、トライアングルマージモードまたは動きベクトル差分 ( M M V D ) マージモードのマージのうちの 1 つまたは両方が使用されるときに、前記情報の少なくとも 1 つのビットをバイパス C A B A C 復号することを備える装置が提供される。適切には、装置が、本発明の第 2 6 の態様または第 2 8 の態様による動き情報予測子に関する情報を復号する方法を実行する手段を備える。

40

【 0 0 7 3 】

本発明の第 3 3 の態様によれば、動きベクトル予測子インデックスを符号化する方法で

50

あって、動きベクトル予測子候補のリストを生成し、リスト内の動きベクトル予測子候補のうちの1つを選択し、C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成し、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットのコンテキスト変数が、現在のブロックのスキップフラグおよびアフィンフラグのうちの少なくとも1つのコンテキスト変数から導出される、方法が提供される。

#### 【 0 0 7 4 】

本発明の第34の態様によれば、動きベクトル予測子インデックスを復号する方法であって、動きベクトル予測子候補のリストを生成し、C A B A C 復号を使用して動きベクトル予測子インデックスを復号し、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットのコンテキスト変数が、現在のブロックのスキップフラグおよびアフィンフラグのうちの少なくとも1つのコンテキスト変数から導出され、復号された動きベクトル予測子インデックスを使用して、リスト内の動きベクトル予測子候補のうちの1つを識別する、方法が提供される。

10

#### 【 0 0 7 5 】

本発明の第35の態様によれば、動きベクトル予測子インデックスを符号化する装置であって、動きベクトル予測子候補のリストを生成する手段と、リスト内の動きベクトル予測子候補のうちの1つを選択する手段と、C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成する手段とを備え、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットのコンテキスト変数が、現在のブロックのスキップフラグおよびアフィンフラグのうちの少なくとも1つのコンテキスト変数から導出される、装置が提供される。

20

#### 【 0 0 7 6 】

本発明の第36の態様によれば、動きベクトル予測子インデックスを復号する装置であって、動きベクトル予測子候補のリストを生成する手段と、C A B A C 復号を使用して、動きベクトル予測子インデックスを復号する手段と、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットのコンテキスト変数が、現在のブロックのスキップフラグおよびアフィンフラグのうちの少なくとも1つのコンテキスト変数から導出され、復号された動きベクトル予測子インデックスを使用して、リスト内の動きベクトル予測子候補のうちの1つを識別する手段とを備える、装置が提供される。

30

#### 【 0 0 7 7 】

本発明の第37の態様によれば、動きベクトル予測子インデックスを符号化する方法であって、動きベクトル予測子候補のリストを生成し、リスト内の動きベクトル予測子候補のうちの1つを選択し、C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成し、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットのコンテキスト変数が、2つの異なる可能な値のみを有する、方法が提供される。

#### 【 0 0 7 8 】

本発明の第38の態様によれば、動きベクトル予測子インデックスを復号する方法であって、動きベクトル予測子候補のリストを生成し、C A B A C 復号を使用して、動きベクトル予測子インデックスを復号し、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットに対するコンテキスト変数が、2つの異なる可能な値のみを有し、復号された動きベクトル予測子インデックスを使用して、リスト内の動きベクトル予測子候補のうちの1つを識別する、方法が提供される。

40

#### 【 0 0 7 9 】

本発明の第39の態様によれば、動きベクトル予測子インデックスを符号化する装置であって、動きベクトル予測子候補のリストを生成する手段と、リスト内の動きベクトル予測子候補のうちの1つを選択する手段と、C A B A C 符号化を使用して、選択された動きベクトル予測子候補の動きベクトル予測子インデックスを生成する手段とを備え、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットのコンテキスト変

50

数は、2つの異なる可能な値のみを有する、装置が提供される。

【0080】

本発明の第40の態様によれば、動きベクトル予測子インデックスを復号する装置であつて、動きベクトル予測子候補のリストを生成する手段と、C A B A C復号を使用して、動きベクトル予測子インデックスを復号する手段と、ここで、現在のブロックの動きベクトル予測子インデックスの少なくとも1つのビットに対するコンテキスト変数が、2つの異なる可能な値のみを有し、復号された動きベクトル予測子インデックスを使用して、リスト内の動きベクトル予測子候補のうちの1つを識別する手段とを備える、装置が提供される。

【0081】

本発明の第41の態様によれば、動き情報予測子インデックスを符号化する方法であつて、動き情報予測子候補のリストを生成し、アフィンマージモードが使用される場合、前記リスト中の動き情報予測子候補のうちの1つをアフィンマージモード予測子として選択し、非アフィンマージモードが使用される場合、前記リスト中の動き情報予測子候補のうちの1つを非アフィンマージモード予測子として選択し、C A B A C符号化を使用して、選択された動き情報予測子候補の動き情報予測子インデックスを生成し、動き情報予測子インデックスの1つまたは複数のビットがバイパスC A B A C符号化されることを特徴とする方法が提供される。

10

【0082】

適切には、C A B A C符号化が、アフィンマージモードが使用される場合および非アフィンマージモードが使用される場合、現在のブロックの動き情報予測子インデックスの少なくとも1つのビットに対して同じコンテキスト変数を使用することを備える。あるいはC A B A C符号化が、現在のブロックの動き情報予測子インデックスの少なくとも1つのビットについて、アフィンマージモードが使用される場合は第1コンテキスト変数を使用すること、または非アフィンマージモードが使用される場合は第2コンテキスト変数を使用することを備え、方法はアフィンマージモードが使用される場合、ビットストリーム中でアフィンマージモードの使用を示すデータを含むことをさらに備える。

20

【0083】

好適には、方法が、ビットストリーム内の動き情報予測子候補の生成されたリストに含まれうる動き情報予測子候補の最大数を決定するためのデータをさらに含む。好適には、動き情報予測子インデックスの最初のビットを除く全てのビットがバイパスC A B A C符号化される。適切には、最初のビットはC A B A C符号化される。適切には、選択された動き情報予測子候補の動き情報予測子インデックスが、アフィンマージモードが使用される場合および非アフィンマージモードが使用される場合、同じシンタックス要素を使用して符号化される。

30

【0084】

本発明の第42の態様によれば、動き情報予測子インデックスを復号する方法であつて、動き情報予測子候補のリストを生成し、C A B A C復号を使用して、動き情報予測子インデックスを復号し、動き情報予測子インデックスの1つまたは複数のビットはバイパスC A B A C復号され、アフィンマージモードが使用される場合、復号された動き情報予測子インデックスを使用して、リスト内の動き情報予測子候補のうちの1つをアフィンマージモード予測子として識別し、非アフィンマージモードが使用される場合、復号された動き情報予測子インデックスを使用して、リスト内の動き情報予測子候補のうちの1つを非アフィンマージモード予測子として識別することを特徴とする方法が提供される。

40

【0085】

適切には、C A B A C復号が、アフィンマージモードが使用される場合および非アフィンマージモードが使用される場合、現在のブロックの動き情報予測子インデックスの少なくとも1つのビットに対して同じコンテキスト変数を使用することを含む。あるいは、方法が、ビットストリームから、アフィンマージモードの使用を示すデータを取得することをさらに含み、C A B A C復号は、現在のブロックの動き情報予測子インデックスの少な

50

くとも 1 つのビットについて、取得されたデータがアフィンマージモードの使用を示す場合、第 1 コンテキスト変数を使用し、取得されたデータが非アフィンマージモードの使用を示す場合、第 2 コンテキスト変数を使用する、ことを含む。

#### 【 0 0 8 6 】

適切には、方法が、ビットストリームから、アフィンマージモードの使用を示すデータを取得することをさらに含み、動き情報予測子候補の生成されたリストは、取得されたデータがアフィンマージモードの使用を示す場合、アフィンマージモード予測子候補、取得されたデータが非アフィンマージモードの使用を示す場合、非アフィンマージモード予測子候補を含む。

#### 【 0 0 8 7 】

適切には、方法が、ビットストリームから、動き情報予測子候補の生成されたリストに含まれる動き情報予測子候補の最大数を決定するためのデータを取得することをさらに含む。好適には、動き情報予測子インデックスの最初のビットを除く全てのビットが、バイパス C A B A C 復号される。適切には、最初のビットは C A B A C 復号される。適当には、動き情報予測子インデックスを復号することは、アフィンマージモードが使用される場合および非アフィンマージモードが使用される場合、ビットストリームから同じシンタックス要素を構文解析することを含む。好適には、動き情報予測子候補が動きベクトルを取得するための情報を含む。適切には、動き情報予測子候補の生成されたリストが、 A T M V P 候補を含む。適切には、動き情報予測子候補の生成されたリストが、アフィンマージモードが使用される場合および非アフィンマージモードが使用される場合、その中に含まれることができる同じ最大数の動き情報予測子候補を有する。

10

#### 【 0 0 8 8 】

本発明の第 4 3 の態様によれば、動き情報予測子インデックスを符号化する装置であって、動き情報予測子候補のリストを生成する手段と、アフィンマージモードが使用される場合、リスト内の動き情報予測子候補のうちの 1 つをアフィンマージモード予測子として選択する手段と、非アフィンマージモードが使用される場合、リスト内の動き情報予測子候補のうちの 1 つを非アフィンマージモード予測子として選択する手段と、 C A B A C 符号化を使用して、選択された動き情報予測子候補の動き情報予測子インデックスを生成する手段であって、動き情報予測子インデックスの 1 つまたは複数のビットがバイパス C A B A C 符号化される手段と、を備えることを特徴とする装置が提供される。好適には、装置が、第 4 1 の態様による動き情報予測子インデックスを符号化する方法を実行する手段を備える。

20

#### 【 0 0 8 9 】

本発明の第 4 4 の態様によれば、動き情報予測子インデックスを復号する装置であって、動き情報予測子候補のリストを生成する手段と、 C A B A C 復号を使用して、動き情報予測子インデックスを復号する手段であって、動き情報予測子インデックスの 1 つまたは複数のビットがバイパス C A B A C 復号される手段と、アフィンマージモードが使用される場合、復号された動き情報予測子インデックスを使用して、リスト内の動き情報予測子候補のうちの 1 つをアフィンマージモード予測子として識別する手段と、非アフィンマージモードが使用される場合、復号された動き情報予測子インデックスを使用して、リスト内の動き情報予測子候補のうちの 1 つを非アフィンマージモード予測子として識別する手段とを備えることを特徴とする装置が提供される。好適には、装置が第 4 2 の態様による動き情報予測子インデックスを復号する方法を実行する手段を備える。

30

#### 【 0 0 9 0 】

本発明の第 4 5 の態様によれば、アフィンマージモードの動き情報予測子インデックスを符号化する方法であって、動き情報予測子候補のリストを生成し、リスト内の動き情報予測子候補のうちの 1 つをアフィンマージモード予測子として選択し、 C A B A C 符号化を使用して、選択された動き情報予測子候補の動き情報予測子インデックスを生成し、動き情報予測子インデックスの 1 つまたは複数のビットがバイパス C A B A C 符号化されることを特徴とする方法が提供される。

40

50

## 【 0 0 9 1 】

適切には、非アフィンマージモードが使用される場合、方法は、リスト内の動き情報予測子候補のうちの 1 つを非アフィンマージモード予測子として選択することをさらに含む。適切には、C A B A C 符号化が、現在のブロックの動き情報予測子インデックスの少なくとも 1 つのビットについて、アフィンマージモードが使用される場合は第 1 コンテキスト変数を使用し、または非アフィンマージモードが使用される場合は第 2 コンテキスト変数を使用することを備え、方法は、アフィンマージモードが使用される場合、ビットストリームにアフィンマージモードの使用を示すデータを含むことをさらに備える。あるいは、C A B A C 符号化が、アフィンマージモードが使用される場合および非アフィンマージモードが使用される場合、現在のブロックの動き情報予測子インデックスの少なくとも 1 つのビットに対して同じコンテキスト変数を使用することを含む。

10

## 【 0 0 9 2 】

好適には、方法が、ビットストリーム内の動き情報予測子候補の生成されたリストに含まれうる動き情報予測子候補の最大数を決定するためのデータをさらに含む。

## 【 0 0 9 3 】

好適には、動き情報予測子インデックスの最初のビットを除く全てのビットが、バイパス C A B A C 符号化される。適切には、最初のビットは C A B A C 符号化される。適切には、選択された動き情報予測子候補の動き情報予測子インデックスが、アフィンマージモードが使用される場合および非アフィンマージモードが使用される場合、同じシンタックス要素を使用して符号化される。

20

## 【 0 0 9 4 】

本発明の第 4 6 の態様によれば、アフィンマージモードの動き情報予測子インデックスを復号する方法であって、動き情報予測子候補のリストを生成し、C A B A C 復号を使用して、動き情報予測子インデックスを復号し、動き情報予測子インデックスの 1 つまたは複数のビットがバイパス C A B A C 復号され、アフィンマージモードが使用される場合、復号された動き情報予測子インデックスを使用して、リスト内の動き情報予測子候補のうちの 1 つをアフィンマージモード予測子として識別することを特徴とする方法が提供される。

## 【 0 0 9 5 】

適切には、非アフィンマージモードが使用される場合、方法は、復号された動き情報予測子インデックスを使用して、リスト内の動き情報予測子候補のうちの 1 つを非アフィンマージモード予測子として識別することをさらに含む。適切には、方法が、ビットストリームから、アフィンマージモードの使用を示すデータを取得することをさらに含み、C A B A C 復号は、現在のブロックの動き情報予測子インデックスの少なくとも 1 つのビットについて、取得されたデータがアフィンマージモードの使用を示す場合、第 1 コンテキスト変数を使用し、取得されたデータが非アフィンマージモードの使用を示す場合、第 2 コンテキスト変数を使用する、ことを含む。あるいは、C A B A C 復号が、アフィンマージモードが使用される場合および非アフィンマージモードが使用される場合、現在のブロックの動き情報予測子インデックスの少なくとも 1 つのビットに対して同じコンテキスト変数を使用することを含む。

30

## 【 0 0 9 6 】

適切には、方法が、ビットストリームから、アフィンマージモードの使用を示すデータを取得することをさらに含み、動き情報予測子候補の生成されたリストは、取得されたデータがアフィンマージモードの使用を示す場合、アフィンマージモード予測子候補、および取得されたデータが非アフィンマージモードの使用を示す場合、非アフィンマージモード予測子候補を含む。

40

## 【 0 0 9 7 】

適切には、動き情報予測子インデックスを復号することは、アフィンマージモードが使用される場合および非アフィンマージモードが使用される場合、ビットストリームから、同じシンタックス要素を構文解析することを含む。適切には、方法がビットストリームか

50

ら、動き情報予測子候補の生成されたリストに含まれうる動き情報予測子候補の最大数を決定するためのデータを取得することをさらに含む。好適には、動き情報予測子インデックスの最初のビットを除く全てのビットがバイパスC A B A C復号される。適切には、最初のビットはC A B A C復号される。好適には、動き情報予測子候補が動きベクトルを取得するための情報を含む。適切には、動き情報予測子候補の生成されたリストがA T M V P候補を含む。適切には、動き情報予測子候補の生成されたリストがアフィンマージモードが使用される場合および非アフィンマージモードが使用される場合、その中に含まれることができる同じ最大数の動き情報予測子候補を有する。

【0098】

本発明の第47の態様によれば、アフィンマージモードの動き情報予測子インデックスを符号化する装置であって、動き情報予測子候補のリストを生成する手段と、リスト内の動き情報予測子候補のうちの1つをアフィンマージモード予測子として選択する手段と、C A B A C符号化を使用して、選択された動き情報予測子候補の動き情報予測子インデックスを生成する手段であって、動き情報予測子インデックスの1つまたは複数のビットがバイパスC A B A C符号化される手段とを備えることを特徴とする装置が提供される。好適には、装置が第45の態様による動き情報予測子インデックスを符号化する方法を実行する手段を備える。

10

【0099】

本発明の第48の態様によれば、アフィンマージモードの動き情報予測子インデックスを復号する装置であって、動き情報予測子候補のリストを生成する手段と、C A B A C復号を使用して、動き情報予測子インデックスを復号する手段であって、動き情報予測子インデックスの1つまたは複数のビットがバイパスC A B A C復号される手段と、アフィンマージモードが使用される場合、復号された動き情報予測子インデックスを使用して、リスト内の動き情報予測子候補のうちの1つをアフィンマージモード予測子として識別する手段と、を備えることを特徴とする装置が提供される。好適には、装置が第46の態様による動き情報予測子インデックスを復号する方法を実行する手段を備える。

20

【0100】

本発明のさらに別の態様は、コンピュータまたはプロセッサによって実行されると、コンピュータまたはプロセッサに前述の態様の方法のいずれかを実行させるプログラムに関する。プログラムは、それ自体で提供されてもよく、または搬送媒体上で、搬送媒体によつて、または搬送媒体内で搬送されてもよい。搬送媒体は、非一時的であってもよく、例えば、記憶媒体、特にコンピュータ可読記憶媒体であってもよい。搬送媒体はまた、一時的なもの、例えば、信号または他の伝送媒体であってもよい。信号は、インターネットを含む任意の適切なネットワークを介して送信されてもよい。

30

【0101】

本発明のさらに別の態様は、前述の装置態様のいずれかによる装置を備えるカメラに関する。一実施形態では、カメラがズーミング手段をさらに備える。一実施形態では、カメラが前記ズーミング手段が動作可能であるときを示し、ズーミング手段が動作可能であるという前記指示に応じてインター予測モードを信号で伝えるように適合される。別の実施形態では、カメラがパンニング手段をさらに備える。別の実施形態ではカメラが前記パンニング手段が動作可能であるときを示し、パンニング手段が動作可能であるという前記指示に応じて、インター予測モードを信号で伝えるように適合される。

40

【0102】

本発明のさらに別の態様によれば、上記のカメラ態様のいずれかを具現化するカメラを備えるモバイルデバイスが提供される。一実施形態では、モバイルデバイスがモバイルデバイスの向きの変化を感知するように適合された少なくとも1つの位置センサをさらに備える。一実施形態では、モバイルデバイスが前記モバイルデバイスの向きの変化を前記感知することに依存して、インター予測モードを信号で伝えるように適合される。

【0103】

本発明のさらなる特徴は、他の独立請求項および従属請求項によって特徴付けられる。

50

## 【0104】

本発明の一態様における任意の特徴は、任意の適切な組み合わせで、本発明の他の態様に適用されてもよい。特に、方法の態様は、装置の態様に適用されてもよく、逆もまた同様である。さらに、ハードウェアで実施される特徴は、ソフトウェアで実施されてもよく、その逆も可能である。したがって、本明細書でソフトウェアおよびハードウェアの特徴に対する任意の参照が解釈されるべきである。本明細書で説明されるような任意の装置の特徴もまた、方法の特徴として提供されてもよく、その逆もまた同様である。本明細書で使用されるように、ミーンズプラスファンクション特徴は、適切にプログラムされたプロセッサおよび関連するメモリなど、それらの対応する構造に関して代替的に表現されてもよい。

10

## 【0105】

また、本発明の任意の態様において説明され、定義された様々な特徴の特定の組合せは、独立して実装および／または供給および／または使用されることを理解されたい。

## 【図面の簡単な説明】

## 【0106】

ここで、例として、添付の図面を参照する。

【図1】図1は、HEVCで使用される符号化構造の説明に用いる図である。

【図2】図2は、本発明の1つまたは複数の実施形態を実施することができるデータ通信システムを概略的に示すブロック図である。

20

【図3】図3は、本発明の1つまたは複数の実施形態を実施することができる処理装置の構成要素を示すブロック図である。

【図4】図4は、本発明の実施形態による符号化方法のステップを示すフローチャートである。

【図5】図5は、本発明の実施形態による復号方法のステップを示すフローチャートである。

【図6a】図6aは、動きベクトル予測子を生成するために使用することができる空間的ブロックおよび時間的ブロックを示す。

【図6b】図6bは、動きベクトル予測子を生成するために使用することができる空間的ブロックおよび時間的ブロックを示す。

30

【図7】図7は、AMVP予測子セット導出の処理の簡略化されたステップを示す。

【図8】図8は、マージモードの動きベクトル導出処理の概略図である。

【図9】図9は、現在のブロックのセグメンテーションおよび時間的動きベクトル予測を示す。

【図10】図10(a)は、HEVCに対する、またはATMVPがSPSレベルでインープルされていない場合のマージインデックスの符号化を示す。図10(b)は、ATMVPがSPSレベルでインープルされる場合のマージインデックスの符号化を示す。

【図11】図11(a)は、単純なアフィン動きフィールドを示す。図11(b)は、より複雑なアフィン動きフィールドを示す。

【図12】図12は、符号化モードに関するいくつかのシンタックス要素の部分復号処理のフローチャートである。

40

【図13】図13は、マージ候補導出を示すフローチャートである。

【図14】図14は、本発明の第1の実施形態に係るマージインデックスの符号化を示す。

【図15】図15は、本発明の第12の実施形態における符号化モードに関するいくつかのシンタックス要素の部分復号処理のフローチャートである。

【図16】図16は、本発明の第12の実施形態におけるマージ候補のリストの生成を示すフローチャートである。

【図17】図17は、本発明の実施形態で使用するのに適したCABACエンコーダを説明する際に使用するためのブロック図である。

【図18】図18は、本発明の1つまたは複数の実施形態の実施のための通信システムの

50

概略ブロック図である。

【図 19】図 19 は、コンピューティング装置の概略ブロック図である。

【図 20】図 20 は、ネットワークカメラシステムを示す図である。

【図 21】図 21 は、スマートフォンを示す図である。

【図 22】図 22 は、第 16 の実施形態による、符号化モードに関連するいくつかのシンタックス要素の部分復号処理のフローチャートである。

【図 23】図 23 は、実施形態に係るマージモードおよびアフィンマージモードの両方のための单一インデックスシグナリング方式の使用を示すフローチャートである。

【図 24】図 24 は、実施形態に係るアフィンマージモードのアフィンマージ候補導出処理を示すフローチャートである。

【図 25】図 25 ( a ) および図 25 ( b ) は、一実施形態によるトライアングルマージモードの予測子導出処理を示す。

【図 26】図 26 は、一実施形態による、現符号化ユニットのためのインター予測モードの復号処理のフローチャートである。

【図 27】図 27 ( a ) は、一実施形態による、動きベクトル差分 ( MMVD ) マージモードのマージのためのフラグの符号化を示す。図 27 ( b ) は、一実施形態によるトライアングルマージモードのためのインデックスの符号化を示す。

【図 28】図 28 は、一実施形態による、ATMVP 候補を用いたアフィンマージモードのためのアフィンマージ候補導出処理を示すフローチャートである。

【図 29】図 29 は、第 18 の実施形態によるインター予測モードの復号処理のフローチャートである。

【図 30】図 30 ( a ) は、第 19 の実施形態による、動きベクトル差分 ( MMVD ) マージモードのマージのためのフラグの符号化を示す。図 30 ( b ) は、第 19 の実施形態によるトライアングルマージモードのためのインデックスの符号化を示す。図 30 ( c ) は、第 19 の実施形態に係るアフィンマージモードまたはマージモードのインデックスの符号化を示す。

【図 31】図 31 は、第 19 の実施形態に係るインター予測モードの復号処理のフローチャートである。

#### 【発明を実施するための形態】

##### 【0107】

以下で説明する本発明の実施形態は、CABAC を使用してインデックス / フラグ / 情報 / データの符号化および復号を改善することに関する。本発明の代替実施形態によれば、CABAC と機能的に同様の他のコンテキストベースの算術符号化方式を改善するための実装も可能であることを理解されたい。実施形態を説明する前に、ビデオ符号化および復号技術、ならびに関連するエンコーダおよびデコーダについて説明する。

##### 【0108】

この明細書では、「シグナリング」が使用、使用中止、モードの有効化または無効化（例えば、インター予測モード）または他の情報（選択に関する情報など）を表す 1 つ以上のシンタックス要素に関するビットストリーム情報を挿入（提供 / 含む / エンコード中）または抽出 / 取得（復号）することを指してもよい。

##### 【0109】

図 1 は、高効率ビデオ符号化 ( HEVC ) ビデオ規格で使用される符号化構造に関する。ビデオシーケンス 1 は、一連のデジタル画像  $i$  から構成される。このような各デジタル画像は、1 つ以上のマトリックスによって表される。行列係数はピクセルを表す。

##### 【0110】

このシーケンスの画像 2 は、スライス 3 に分割される。スライスは、場合によっては画像全体を構成する。これらのスライスは、オーバーラップしない符号化ツリーユニット ( CTU ) に分割される。符号化ツリーユニット ( CTU ) は、高効率ビデオ符号化 ( HEVC ) ビデオ規格の基本的な処理ユニットであり、概念的には、構造が、いくつかの以前のビデオ規格で使用されたマクロブロックユニットに対応する。CTU は、時には最大符

10

20

30

40

50

号化ユニット（LCU）とも呼ばれる。CTUは輝度及び彩度構成要素部分を有し、その構成要素部分の各々は、符号化ツリーブロック（CTB）と呼ばれる。これらの異なる色成分は、図1には示されていない。

#### 【0111】

CTUは一般的に、HEVC用のサイズ64ピクセル×64ピクセルであるが、VVC用にはこのサイズが128ピクセル×128ピクセルであり得る。各CTUは順番に、四分木分解を使用して、より小さい可変サイズ符号化ユニット（CU）5に反復的に分割されてもよい。

#### 【0112】

符号化ユニットは基本符号化要素であり、予測ユニット（PU）と変換ユニット（TU）と呼ばれる2種類のサブユニットから構成される。PUまたはTUの最大サイズは、CUサイズに等しい。予測ユニットは、ピクセル値の予測のためのCUの区分に対応する。6に示すように、4つの正方形PUへのパーティションと2つの長方形PUへの2つの異なるパーティションとを含む、PUへのCUの様々な異なるパーティションが可能である。変換ユニットは、DCTを使用して空間変換を行う基本ユニットである。CUは、四分木表現7に基づいてTUにパーティション化できる。したがって、スライス、タイル、CTU/LCU、CTB、CU、PU、TU、またはピクセル/サンプルのブロックは、画像部分、すなわちシーケンスの画像2の一部と呼ばれることがある。

#### 【0113】

各スライスは、1つのネットワーク抽象化層（NAL）ユニットに埋め込まれる。さらに、ビデオシーケンスの符号化パラメータは、パラメータセットと呼ばれる専用NALユニットに記憶される。HEVCおよびH.264/AVCでは、2種類のパラメータセットNALユニットが使用される：第1に、ビデオシーケンス全体の間に変更されないすべてのパラメータを収集するシーケンスパラメータセット（SPS）NALユニット。典型的には、それは符号化プロファイル、ビデオフレームのサイズ、及び他のパラメータを処理する。第2に、ピクチャパラメータセット（PPS）NALユニットは、シーケンスの1つの画像（またはフレーム）から別の画像（またはフレーム）に変更することができるパラメータを含む。HEVCには、ビットストリームの全体的な構造を記述するパラメータを含むビデオパラメータセット（VPS）NALユニットも含まれている。VPSは、HEVCで定義された新しいタイプのパラメータセットで、ビットストリームのすべてのレイヤに適用される。レイヤには、複数の時間的サブレイヤを含めることができ、すべてのバージョン1のビットストリームは、1つのレイヤに制限される。HEVCには、拡張性とマルチビューのための特定の層拡張があり、これらは後方互換性のあるバージョン1基本層を備えた複数の層を可能にする。

#### 【0114】

図2および図18は、本発明の1つまたは複数の実施形態を実施することができるデータ通信システムを示す。データ通信システムは、データ通信ネットワーク200を介して、データストリーム204（または図18のビットストリーム101）のデータパケットを受信装置、例えば、図2のクライアント端末202または図18のコンテンツ消費者100に送信するように動作可能な、送信装置、例えば、図2のサーバ201または図18のコンテンツプロバイダ150を含む。データ通信ネットワーク200は、ワイドエリアネットワーク（WAN）またはローカルエリアネットワーク（LAN）であってもよい。このようなネットワークは例えば、無線ネットワーク（Wi fi / 802.11aまたはbまたはg）、イーサネットネットワーク、インターネットネットワーク、またはいくつかの異なるネットワークから構成される混合ネットワークであってもよい。本発明の特定の実施形態では、データ通信システムは、サーバ201（または図18のコンテンツプロバイダ150）が同じデータコンテンツを複数のクライアント（またはコンテンツ消費者）に送信するデジタルテレビ放送システムであってもよい。

#### 【0115】

サーバ201（またはコンテンツプロバイダ150）によって提供されるデータストリ

10

20

30

40

50

ーム 204 (またはビットストリーム 101) は、ビデオを表すマルチメディアデータおよびオーディオデータから構成されてもよい。オーディオおよびビデオデータストリームは、本発明のいくつかの実施形態では、それぞれマイクロフォンおよびカメラを使用してサーバ 201 (またはコンテンツプロバイダ 150) によってキャプチャされ得る。いくつかの実施形態において、データストリームは、サーバ 201 (またはコンテンツプロバイダ 150) 上に記憶されてもよく、あるいは別のデータプロバイダからサーバ 201 (またはコンテンツプロバイダ 150) によって受信されてもよく、あるいはサーバ 201 (またはコンテンツプロバイダ 150) で生成されてもよい。サーバ 201 (またはコンテンツプロバイダ 150) は特に、エンコーダへの入力として提示されるデータのよりコンパクトな表現である送信のための圧縮ビットストリーム 204, 101 を提供するために、ビデオストリームおよびオーディオストリーム (例えば、図 18 の画像のオリジナルシーケンス 151) を符号化するためのエンコーダを備える。

【0116】

送信データの品質対送信データの量のより良好な比率を得るために、ビデオデータの圧縮は例えば、HEVC フォーマット、または H.264 / AVC フォーマット、または VVC フォーマットに従ってもよい。

【0117】

クライアント 202 (またはコンテンツ消費者 100) は、送信されたビットストリームを受信し、再構成されたビットストリームを復号して、表示装置上でビデオ画像 (例えば、図 18 のビデオ信号 109) を再生し、スピーカによってオーディオデータを再生する。

【0118】

図 2 または図 18 の例では、ストリーミングシナリオが考慮されているが、本発明のいくつかの実施形態では、エンコーダとデコーダとの間のデータ通信が例えば、光ディスクなどの媒体記憶デバイスを使用して実行され得ることが理解されよう。

【0119】

本発明の 1 つまたは複数の実施形態では、ビデオ画像は、画像の再構成されたピクセルに適用して最終画像内のフィルタリングされたピクセルを提供するために、補償オフセットを表すデータと共に送信されてもよい。

【0120】

図 3 は、本発明の少なくとも 1 つの実施形態を実施するように構成された処理装置 300 を概略的に示す。処理装置 300 は、マイクロコンピュータ、ワークステーション、またはライトポータブル装置などの装置とすることができます。装置 300 は、

- CPU で示されるマイクロプロセッサなどの中央処理部 311

- 本発明を実施するためのコンピュータプログラムを記憶するための ROM と表記される読み出し専用メモリ 307

- 本発明の実施形態の方法の実行可能コード、ならびに本発明の実施形態に係るデジタル画像のシーケンスを符号化する方法および / またはビットストリームを復号する方法を実現するために必要な変数およびパラメータを記録するように構成されたレジスタを記憶する RAM で示されたランダムアクセスメモリ 312

- 処理されるデジタルデータが送受信される通信ネットワーク 303 に接続された通信インターフェース 302

と接続された通信バス 313 を備える。

【0121】

オプションで、装置 300 は、以下の構成要素を含むこともできる。

【0122】

- 本発明の 1 つまたは複数の実施形態の方法を実施するためのコンピュータプログラム、および本発明の 1 つまたは複数の実施形態の実施中に使用または生成されるデータを記憶するための、ハードディスクなどのデータ記憶手段 304

- ディスク 306 用のディスクドライブ 305 であって、ディスクドライブはディスク

10

20

30

40

50

306 からデータを読み取るか、またはディスクにデータを書き込むように構成されている - キーボード 310 または他の任意のポインティング / 入力手段を用いて、データを表示し、および / またはユーザとのグラフィカルインターフェースとして機能する画面 309 装置 300 は例えば、デジタルカメラ 320 またはマイクロフォン 308 のような種々の周辺機器に接続されることが可能であり、各周辺機器はマルチメディアデータを装置 300 に供給するために、入力 / 出力カード ( 図示せず ) に接続される。

#### 【 0123 】

通信バスは、装置 300 に含まれる、またはそれに接続される種々の要素間の通信および相互運用性を提供する。バスの表現は限定されず、特に、中央処理ユニットは直接的に、または装置 300 の別の要素を用いて、装置 300 の任意の要素に命令を通信するよう 10 に動作可能である。

#### 【 0124 】

ディスク 306 は、例えばコンパクトディスク ( C D - R O M ) 、書き換え可能または 20 そわない、 Z I P ディスクまたはメモリカードなどの任意の情報媒体に置き換えることができ、一般的に言えば、マイクロコンピュータまたはマイクロプロセッサによって読み取ることができる情報記憶手段によって置き換えることができ、装置に統合または非統合され、可能であれば、リムーバブルであり、実行がデジタル画像のシーケンスを符号化する方法および / または本発明によるビットストリームを復号する方法を実行可能にする 1 つ以上のプログラムを記憶するように構成することができる。

#### 【 0125 】

実行可能コードは、読み出し専用メモリ 307 、ハードディスク 304 、または先に説明したような例えばディスク 306 のようなリムーバブルデジタル媒体のいずれかに格納することができる。変形例によれば、プログラムの実行可能コードは、ハードディスク 304 のような実行される前に装置 300 の記憶手段の 1 つに記憶されるために、インターフェース 302 を介して、通信ネットワーク 303 によって受信することができる。

#### 【 0126 】

中央処理ユニット 311 は、前述の記憶手段のうちの 1 つに記憶されている命令で、本発明によるプログラムまたはプログラムのソフトウェアコードの命令または部分の実行を制御し、指示するように構成されている。電源投入時に、例えばハードディスク 304 や 30 306 上または読み出し専用メモリ 307 中の不揮発性メモリに記憶されたプログラムまたはプログラムはランダムアクセスメモリ 312 に転送され、ランダムアクセスメモリ 213 はプログラムまたはプログラムの実行可能コード、ならびに本発明を実施するため必要な変数およびパラメータを記憶するためのレジスタを含む。

#### 【 0127 】

この実施形態では、装置は、本発明を実施するためにソフトウェアを使用するプログラム装置である。しかしながら、代替的に、本発明はハードウェア ( 例えば、特定用途向け集積回路または A S I C の形態 ) で実施されてもよい。

#### 【 0128 】

図 4 は、本発明の少なくとも 1 つの実施形態によるエンコーダのブロック図を示す。エンコーダは、接続されたモジュールによって表され、各モジュールは例えば、デバイス 300 の C P U 311 によって実行されるべきプログラム命令の形態で、本発明の 1 つまたは複数の実施形態による画像のシーケンスの画像を符号化する少なくとも 1 つの実施形態を実施する方法の少なくとも 1 つの対応するステップを実施するように適合される。

#### 【 0129 】

デジタル画像 i0 ~ in401 のオリジナルシーケンスは、エンコーダ 400 によって入力として受信される。各デジタル画像は、時には画素 ( 以下、画素と呼ぶ ) とも呼ばれる 1 組のサンプルによって表される。

#### 【 0130 】

ビットストリーム 410 は、符号化処理の実施後にエンコーダ 400 によって出力される。ビットストリーム 410 は、複数の符号化ユニットまたはスライスを備え、各スライ 50

スはスライスを符号化するために使用される符号化パラメータの符号化値を送信するためのスライスヘッダと、符号化されたビデオデータを備えるスライス本体とを備える。

#### 【0131】

入力デジタル画像  $i_0 \sim i_{n-4} 0 1$  は、モジュール 402 によってピクセルのブロックに分割される。ブロックは、画像部分に対応し、可変サイズであってもよい（例えば、 $4 \times 4$ 、 $8 \times 8$ 、 $16 \times 16$ 、 $32 \times 32$ 、 $64 \times 64$ 、 $128 \times 128$  ピクセル、およびいくつかの矩形ブロックサイズも考慮することができる）。符号化モードは、各入力ブロックに対して選択される。空間的予測符号化（イントラ予測）に基づく符号化モードと、時間的予測に基づく符号化モード（インター符号化、マージ、S K I P）の 2 つのファミリの符号化モードが提供される。可能な符号化モードがテストされる。

10

#### 【0132】

モジュール 403 は、符号化されるべき所与のブロックが符号化されるべき前記ブロックの近隣の画素から計算された予測子によって予測されるイントラ予測処理を実施する。選択されたイントラ予測子の指示、および所与のブロックとその予測子との間の差は、イントラ符号化が選択された場合に残差を提供するために符号化される。

#### 【0133】

時間的予測は、動き推定モジュール 404 および動き補償モジュール 405 によって実施される。最初に、参照画像 416 のセットの中から参照画像が選択され、符号化されるべき所与のブロックに最も近い領域（画素値類似性に関して最も近い）である参照領域または画像部分とも呼ばれる参照画像の部分が、動き推定モジュール 404 によって選択される。次いで、動き補償モジュール 405 は、選択された領域を使用して、符号化されるべきブロックを予測する。選択された参照領域と、残差ブロックとも呼ばれる所与のブロックとの間の差は、動き補償モジュール 405 によって計算される。選択された参照領域は、動きベクトルを用いて示される。

20

#### 【0134】

したがって、両方の場合（空間的予測および時間的予測）において、残差は、元のブロックが S K I P モードにないときに、元のブロックから予測子を減算することによって計算される。

#### 【0135】

モジュール 403 によって実施される I N T R A 予測では、予測方向が符号化される。モジュール 404、405、416、418、417 によって実施されるインター予測では、そのような動きベクトルを識別するための少なくとも 1 つの動きベクトルまたはデータが時間的予測のために符号化される。

30

#### 【0136】

インター予測が選択された場合、動きベクトルおよび残差ブロックに関連する情報が符号化される。ビットレートをさらに低減するために、動きが均一であると仮定すると、動きベクトルは、動きベクトル予測子に対する差によって符号化される。1 組の動き情報予測子候補からの動きベクトル予測子は、動きベクトル予測符号化モジュール 417 によって動きベクトルフィールド 418 から得られる。

40

#### 【0137】

エンコーダ 400 はさらに、レート歪み基準などの符号化コスト基準を適用することによって、符号化モードを選択するための選択モジュール 406 を備える。冗長性をさらに低減するために、変換モジュール 407 によって変換（D C T など）を残差ブロックに適用し、得られた変換データを量子化モジュール 408 によって量子化し、エントロピー符号化モジュール 409 によってエントロピー符号化する。最後に、符号化されている現在のブロックの符号化された残差ブロックは S K I P モードにないとき、ビットストリーム 410 に挿入され、モードは、ビットストリームで符号化される残差ブロックを必要とする。

#### 【0138】

また、エンコーダ 400 は、後続の画像の動き推定のための参照画像（例えば、参照画

50

像ノピクチャ 416 内のそれら)を生成するために、符号化された画像の復号を実行する。これは、ビットストリームを受信するエンコーダ及びデコーダが同じ参照フレームを有する(再構成された画像又は画像部分が使用される)ことを可能にする。逆量子化(「デクオンタイゼイション」)モジュール 411 は、量子化データの逆量子化(「デクオンタイゼイション」)を行い、続いて逆変換モジュール 412 による逆変換を行う。イントラ予測モジュール 413 は、予測情報を使用して、所与のブロックにどの予測子を使用するを決定し、動き補償モジュール 414 は、モジュール 412 によって得られた残差を、参照画像 416 のセットから得られた参照領域に実際に加算する。

#### 【0139】

その後、モジュール 415 によってポストフィルタリングが適用され、画素の再構成されたフレーム(画像または画像部分)がフィルタリングされる。本発明の実施形態では、補償オフセットが再構成画像の再構成画素の画素値に付加される S A O ループフィルタが使用される。ポストフィルタリングは、必ずしも実行される必要はないことが理解される。また、S A O ループフィルタリングに加えて、または S A O ループフィルタリングの代わりに、任意の他のタイプのポストフィルタリングを実行することもできる。

10

#### 【0140】

図 5 は、本発明の一実施形態による、エンコーダからデータを受信するために使用され得るデコーダ 60 のブロック図を示す。デコーダは、接続されたモジュールによって表され、各モジュールは例えば、デバイス 300 の C P U 311 によって実行されるプログラム命令の形成で、デコーダ 60 によって実現される方法の対応するステップを実現するように構成される。

20

#### 【0141】

デコーダ 60 は、符号化ユニット(例えば、画像部分、ブロックまたは符号化ユニットに対応するデータ)を含むビットストリーム 61 を受信し、各符号化ユニットは、符号化パラメータに関する情報を含むヘッダと、符号化されたビデオデータを含む本体とから構成される。図 4 に関して説明したように、符号化されたビデオデータは、エントロピー符号化され、動きベクトル予測子のインデックスは、所与の画像部分(例えば、ブロックまたは C U)に対して、所定のビット数で符号化される。受信された符号化ビデオデータは、モジュール 62 によってエントロピー復号される。次いで、残差データはモジュール 63 によって逆量子化され、次いで、画素値を得るためにモジュール 64 によって逆変換が適用される。

30

#### 【0142】

符号化モードを示すモードデータもエントロピー復号され、このモードに基づいて、画像データの符号化ブロック(ユニット/セット/グループ)に対して I N T R A タイプ復号または I N T E R タイプ復号が行われる。

#### 【0143】

I N T R A モードの場合、I N T R A 予測子は、ビットストリームで指定されたイントラ予測モードに基づいてイントラ予測モジュール 65 によって決定される。

#### 【0144】

モードが I N T E R である場合、エンコーダによって使用される参照領域を見つける(識別する)ために、動き予測情報がビットストリームから抽出される。動き予測情報は、参照フレームインデックスと動きベクトル残差とを含む。動きベクトル予測子は動きベクトルを得るために、動きベクトル復号モジュール 70 によって動きベクトル残差に加算される。

40

#### 【0145】

動きベクトル復号モジュール 70 は、動き予測によって符号化された画像部分(例えば、現在のブロックまたは C U)ごとに、動きベクトル復号を適用する。現在のブロックに対する動きベクトル予測子のインデックスが得られると、画像部分(例えば、現在のブロックまたは C U)に関連する動きベクトルの実際の値が復号され、モジュール 66 によって動き補償を適用するために使用され得る。復号された動きベクトルによって示される参

50

照画像部分は、参照画像 6 8 から抽出され、動き補償 6 6 が適用される。動きベクトルフィールドデータ 7 1 は、後続の復号された動きベクトルの予測に使用するために、復号された動きベクトルで更新される。

#### 【 0 1 4 6 】

最後に、復号されたブロックが得られる。適切な場合には、ポストフィルタリングモジュール 6 7 によってポストフィルタリングが適用される。復号されたビデオ信号 6 9 が最終的に得られ、デコーダ 6 0 によって与えられる。

#### 【 0 1 4 7 】

C A B A C

HEVC は、CABAC (Context based Adaptive Binary Arithmetic Coding) 、ゴロムライス符号化 (Golomb - rice Code) 、または固定長符号化 (Fixed Length Coding) と呼ばれる単純なバイナリ表現など、いくつかのタイプのエントロピー符号化を使用する。ほとんどの場合、異なるシンタックス要素を表現するためにバイナリ符号化処理が実行される。このバイナリ符号化処理も非常に具体的であり、異なるシンタックス要素に依存する。算術符号化は、それらの現在の確率に従ってシンタックス要素を表す。CABAC は、コンテキスト変数によって定義される「コンテキスト」に応じてシンタックス要素の確率を分離する算術符号化の拡張である。これは条件付き確率に相当する。コンテキスト変数は、すでに復号されている、左上のブロック (以下で詳細に説明する図 6 b の A 2 ) および左上のブロック (図 6 b の B 3 ) の現在のシンタックスの値から導き出すことができる。

10

#### 【 0 1 4 8 】

CABAC は、H.264 / AVC および H.265 / HEVC 規格の基準部分として採用されている。H.264 / AVC では、エントロピー符号化の 2 つの代替方法のうちの 1 つである。H.264 / AVC に規定された他の方法は、可変長符号の文脈適応的に切り替えられるセット、いわゆる文脈適応型可変長符号化 (CAVLC) の使用に基づく低複雑性エントロピー符号化技術である。CABAC と比較して、CAVLC は、より低い圧縮効率を犠牲にして、低減された実装コストを提供する。標準解像度または高精細度解像度の TV 信号の場合、CABAC は通常、同じ客観的なビデオ品質で CAVLC に対して 10 ~ 20 % のビットレートの節約を提供する。HEVCにおいて、CABAC は使用されるエントロピー符号化方法の一つである。多くのビットはまた、バイパス CABAC 符号化 (CABAC バイパス符号化としても表される) される。さらに、いくつかのシンタックス要素は単項コードまたは Golomb コードで符号化され、これは他のタイプのエントロピーコードである。

20

30

#### 【 0 1 4 9 】

図 17 は、CABAC エンコーダのメインブロックを示す。

#### 【 0 1 5 0 】

非バイナリ値である入力シンタックス要素は、バイナライザ 1701 によって 2 値化される。CABAC の符号化戦略は、動きベクトル差または変換係数レベル値の成分のような、ハイブリッドブロックベースのビデオ符号化器におけるシンタックス要素値の非常に効率的な符号化が、コンテキストモデリングおよび 2 進算術符号化の後続ステージのための一連の前処理ユニットとして 2 値化方式を使用することによって達成され得るという発見に基づく。一般に、2 値化方式は、シンタックス要素値の二値決定のシーケンス、いわゆるビンへのユニークなマッピングを定義し、これは、「ビット」であり得り、従って 2 値コードツリーの観点から解釈することもできる。CABAC における 2 値化方式の設計は、その構造が簡単なオンライン計算を可能にし、幾つかの適当なモデル - 確率分布に適用される少数の基本的なプロトタイプに基づいている。

40

#### 【 0 1 5 1 】

各ビンは、スイッチ 1702 の設定に従って、2 つの基本的な方法のうちの 1 つで処理することができる。スイッチが「正規 (regular)」設定にある場合、ビンはコンテキストモデル 1703 および正規符号化エンジン 1704 に供給される。スイッチが「バイパ

50

ス」設定にある場合、コンテキストモデルはバイパスされ、ビンはバイパス符号化エンジン 1705 に供給される。別のスイッチ 1706 はスイッチ 1702 と同様の「正規」および「バイパス」設定を有し、その結果、符号化エンジン 1704 および 1705 のうちの適用可能な 1 つによって符号化されたビンは、CABAC エンコーダの出力としてビットストリームを形成することができる。

#### 【0152】

他のスイッチ 1706 は、ビットストリーム中のバイパス符号化データのブロックを提供するために符号化エンジン 1705 によって符号化されたビンのいくつか（例えば、ブロックまたは符号化ユニットなどの画像部分を符号化するためのビン）をグループ化するために、およびビットストリーム中の「正規の」（または算術的に）符号化されたデータの別のブロックを提供するために符号化エンジン 1704 によって符号化されたビンのいくつか（例えば、ブロックまたは符号化ユニットを符号化するためのビン）をグループ化するために、ストレージとともに使用され得ることが理解される。バイパス符号化データと正規符号化データとのこの別個のグループ分けは、（バイパス符号化データを最初に／正規 CABAC 符号化データと並列に処理することができるから）復号処理中のスループットの改善をもたらすことができる。

#### 【0153】

各シンタックス要素値をビンのシーケンスに分解することによって、CABAC における各ビン値のさらなる処理は、関連する符号化モード決定に依存し、これは、正規モードまたはバイパスモードのいずれかとして選択することができる。後者は符号情報に関連するビン、または一様に分布していると仮定され、その結果、正規の 2 進算術符号化処理全体が単にバイパスされる、下位の有意なビンのために選択される。正規の符号化モードでは、各ビン値が正規の 2 進算術符号化エンジンを使用することによって符号化され、関連する確率モデルはコンテキストモデルリングなしに固定選択によって決定されるか、または関連するコンテキストモデルに応じて適応的に選択される。重要な設計決定として、後者の場合は一般に、最も頻繁に観測されるビンのみに適用され、他の、通常は頻繁に観測されないビンは、ジョイント、典型的にはゼロ次確率モデルを使用して処理される。このようにして、CABAC はサブシンボルレベルでの選択的なコンテキストモデルリングを可能にし、したがって、全体的なモデルリングまたは学習コストを大幅に低減してシンボル間冗長性を利用するための効率的な手段を提供する。コンテキストモデルの特定の選択に対して、4 つの基本設計タイプを CABAC で採用し、その中の 2 つを変換係数レベルのみの符号化に適用した。これらの 4 つのプロトタイプの設計は、モデル化されるソースデータの典型的な特性に関する先駆的な知識に基づいており、不必要的モデル化コストのオーバーヘッドを回避し、統計的依存性を大幅に活用するという相反する目的の間の良好な妥協点を見つける目的を反映している。

#### 【0154】

CABAC における最低レベルの処理では、各ビン値が正規またはバイパス符号化モードのいずれかで、バイナリ算術符号器に入る。後者の場合、かなり低減された複雑さを有する符号化エンジンのファーストブランチ (fast branch) が使用され、一方、前者の符号化モードの場合、所与のビン値の符号化は、ビン値と共に M コーダに渡される関連する適応確率モデルの実際の状態に依存し、この用語は、CABAC におけるテーブルベースの適応 2 進算術符号化エンジンのために選択された用語である。

#### 【0155】

次いで、対応する CABAC デコーダは、CABAC エンコーダからビットストリーム出力を受信し、それに応じてバイパス符号化データおよび正規の CABAC 符号化データを処理する。CABAC デコーダが正規の CABAC 符号化データを処理すると、コンテキストモデル（およびその確率モデル）が更新され、ビットストリームを形成するビンを正しく復号／処理（例えば、逆二値化）してシンタックス要素を得ることができる。

#### 【0156】

インター符号化

10

20

30

40

50

H E V C は 3 つの異なるインターモード、すなわち、インターモード（動き情報差をシグナリングする高度動きベクトル予測（A M V P））、「古典的」マージモード（すなわち、「非アフィンマージモード」または動き情報差をシグナリングしない「正規」マージモードとしても知られる）、および「古典的」マージスキップモード（すなわち、「非アフィンマージスキップ」モードまたは動き情報差をシグナリングせず、サンプル値の残差データもシグナリングしない「正規」マージスキップモードとしても知られる）を使用する。これらのモード間の主な違いは、ビットストリームにおけるデータシグナリングである。動きベクトル符号化の場合、現在の H E V C 規格は、規格の以前のバージョンには存在しなかった動きベクトル予測のための競合ベースのスキームを含む。これは、インターモード化モード（A M V P）またはマージモード（すなわち、「古典的 / 正規」マージモードまたは「古典的 / 正規」マージスキップモード）それぞれについて、最良の動きベクトル予測子または最良の動き情報を見つけるために、いくつかの候補がエンコーダ側のレート歪み基準と競合していることを意味する。次に、最良予測子または動き情報の最良候補に対応するインデックスまたはフラグがビットストリームに挿入される。デコーダは、予測子または候補の同じセットを導出し、復号されたインデックス / フラグに従って最良のものを使用することができる。H E V C のスクリーンコンテンツ拡張では、イントラブロックコピー（I B C）と呼ばれる新しいコーディングツールが、これらの 3 つのインターモードのいずれかとしてシグナリングされ、I B C と同等のインターモードとの差は、参照フレームが現在のものであるかどうかをチェックすることによって行われる。I B C は、現在の画像参照（C P R）としても知られている。これは、例えば、リスト L 0 の参照インデックスをチェックし、これがリスト内の最後のフレームである場合、これがイントラブロックコピーであると推定することによって実施することができる。別のやり方は、現在のフレームと参照フレームのピクチャ順序カウントを比較することであり、それらが等しい場合、これはイントラブロックコピーである。

#### 【 0 1 5 7 】

予測子および候補の導出の設計は、複雑さに不均衡な影響を与えることなく、最良の符号化効率を達成する上で重要である。H E V C では、2 つの動きベクトル導出が使用される：1 つはインターモード（Advanced Motion Vector Prediction (A M V P)）のためのものであり、1 つはマージモード（Merge derivation process - for the classical Merge mode and the classical Merge Skip mode）のためのものである。以下、これらの処理について説明する。

#### 【 0 1 5 8 】

図 6 a および図 6 b は、例えば、H E V C 符号化および復号システムの高度動きベクトル予測（A M V P）およびマージモードにおける動きベクトル予測子を生成するために使用することができる空間的ブロックおよび時間的ブロックを示し、図 7 は、A M V P 予測子セット導出の処理の簡略化されたステップを示す。

#### 【 0 1 5 9 】

2 つの空間的予測子、すなわち A M V P モードのための 2 つの空間的動きベクトルが、上隅ブロック（ブロック B 2）および左隅ブロック（ブロック A 0）を含む上ブロック（文字「B」によって示される）および左ブロック（文字「A」によって示される）の動きベクトルの中から選択され、1 つの時間的予測子が図 6 a に表されるように、コロケートされたブロックの右下ブロック（H）および中央ブロック（C e n t e r）の動きベクトルの中から選択される。

#### 【 0 1 6 0 】

以下の表 1 は、図 6 a および 6 b に示されるように、現在のブロックを基準にしてブロックを参照するときに使用される命名法の概要を示す。この命名法は、簡潔に使用されるが、特に将来の規格のバージョンでは他のラベル付けシステムが使用されてもよいことを理解されたい。

#### 【 0 1 6 1 】

10

20

30

40

50

【表1】

ブロックラベル	隣接ブロックの相対位置記述
A0	「左下」または「左隅」—現在のブロックの斜め左下
A1	「左」または「左下」—現在のブロックの下の左
A2	「左上」—現在のブロックの上の左
B0	「右上」—現在のブロックの右斜め上
B1	「上」—現在のブロックの右上
B2	「左上」または「上隅」—現在のブロックの左斜め上
B3	「上」—現在のブロックの左上
H	参照フレーム内のコロケートされたブロックの右下
中央	参照フレーム内のコロケートされたブロック内のブロック

表1

## 【0162】

「現在のブロック」は、サイズが、 $4 \times 4$ 、 $16 \times 16$ 、 $32 \times 32$ 、 $64 \times 64$ 、 $128 \times 128$ 、またはその間の任意のサイズなど、可変であることに注意する必要がある。ブロックの寸法は、好ましくは2の倍数（すなわち、 $2^n \times 2^m$  ここで、 $n$ および $m$ は正の整数である）であり、これは、バイナリ符号化を使用する場合、ビットのより効率的な使用をもたらす。現在のブロックは、正方形である必要はないが、これはしばしば、符号化の複雑さのための好ましい実施形態である。

## 【0163】

図7を参照すると、第1ステップは、左下のブロックA0およびA1の中から第1空間的予測子（C and 1、706）を選択することを目的とし、その空間的位置が図6aに示される。のために、これらのブロックは、所与の（すなわち、所定のノブリセットされた）順序で次々に選択され（700、702）、選択されたブロックごとに、所与の順序で以下の条件が評価され（704）、条件が満たされる第1ブロック（first block）は、予測子として設定される。

## 【0164】

- 同じ参照画像と同じ参照リストからの動きベクトル
- 同じ参照画像と他の参照リストからの動きベクトル
- 異なる参照画像と同じ参照リストからのスケーリングされた動きベクトル
- 異なる参照画像と他の参照リストからのスケーリングされた動きベクトル

値が見つからない場合、左予測子は使用不可と見なされる。この場合、関連ブロックがイントラ符号化されているか、またはそれらのブロックが存在しないことを示す。

## 【0165】

後のステップは、上の右ブロックB0、上のブロックB1、および左の上（上の左）ブロックB2の中から第2空間的予測子（C and 2、716）を選択することを目的とし、その空間的位置が図6aに示されている。のために、これらのブロックは、所与の順序で次々に選択され（708、710、712）、選択されたブロックごとに、所与の順序で上述の条件が評価され（714）、上述の条件が満たされる第1ブロックが予測子として設定される。

## 【0166】

再び、値が見つからない場合、上の予測子は利用不可能であると見なされる。この場合、関連ブロックがイントラ符号化されているか、またはそれらのブロックが存在しないこ

10

20

30

40

50

とを示す。

#### 【0167】

次のステップ(718)において、2つの予測子は、両方が利用可能である場合、それらが等しい場合(すなわち、同じ動きベクトル値、同じ参照リスト、同じ参照インデックス、および同じ方向タイプ)、それらのうちの1つを除去するために、互いに比較される。1つの空間的予測子のみが利用可能である場合、アルゴリズムは、後のステップにおいて時間的予測子を探す。

#### 【0168】

時間的動き予測子(C and 3, 726)は、次のように導出される。前の/参照フレームにおけるコロケートされたブロックの右下(H, 720)位置がまず、利用可能性チェックモジュール722において考慮される。それが存在しない場合、または動きベクトル予測子が利用可能でない場合、コロケートされたブロックの中央(中央、724)がチェックされるように選択される。これらの時間的位置(中央およびH)を図6aに示す。いずれにせよ、スケーリング723は、現在のフレームと参照リスト内の第1フレーム(first frame)との間の時間的距離を一致させるために、これらの候補に適用される。

#### 【0169】

次に、動き予測子値が予測子のセットに追加される。次に、予測子の数(Nb\_C and)が、予測子の最大数(Max\_C and)と比較される(728)。上述のように、AMPの導出処理が生成する必要がある動きベクトル予測子の予測子の最大数(Max\_C and)は、HEVC規格の現在のバージョンでは2である。

#### 【0170】

この最大数に達した場合、AMP予測子の最終リストまたはセット(732)が構築される。そわなければ、ゼロ予測子がリストに追加される(730)。ゼロ予測子は、(0, 0)に等しい動きベクトルである。

#### 【0171】

図7に示すように、AMP予測子の最終リストまたはセット(732)は、空間的動き予測子候補のサブセット(700~712)および時間的動き予測子候補のサブセット(720、724)から構築される。

#### 【0172】

上述のように、古典的マージモードまたは古典的マージスキップモードの動き予測子候補は、方向、リスト、参照フレームインデックス、および動きベクトル(または予測を実行するためのその任意のサブセット)というすべての必要な動き情報を表すことができる。いくつかの候補のインデックス付きリストが、マージ導出処理によって生成される。現在のHEVC設計では、両方のマージモード(すなわち、古典的マージモードおよび古典的マージスキップモード)の候補の最大数は5(4つの空間的候補および1つの時間的候補)に等しい。

#### 【0173】

図8は、マージモード(古典的マージモードおよび古典的マージスキップモード)の動きベクトル導出処理の概略図である。導出処理の第1ステップでは、5つのブロック位置が考慮される(800~808)。これらの位置は、参照番号A1、B1、B0、A0、およびB2で図6aに示される空間的位置である。後のステップにおいて、空間的動きベクトルの利用可能性がチェックされ、考慮のために多くとも5つの動きベクトルが選択/取得される(810)。予測子が存在し、ブロックがイントラ符号化されていない場合、予測子は利用可能であると見なされる。したがって、5つのブロックに対応する動きベクトルを候補として選択することは、以下の条件に従って行われる。

#### 【0174】

「左」A1動きベクトル(800)が利用可能である場合(810)、すなわち、それが存在し、このブロックがイントラ符号化されていない場合、「左」ブロックの動きベクトルが選択され、候補リスト内の第1候補として使用される(814)。

#### 【0175】

10

20

30

40

50

「上」B 1 動きベクトル (8 0 2) が利用可能である場合 (8 1 0)、候補「上」ブロック動きベクトルは、存在する場合、「左」A 1 動きベクトルと比較される (8 1 2)。B 1 動きベクトルがA 1 動きベクトルに等しい場合、B 1 は空間的候補のリストに追加されない (8 1 4)。逆に、B 1 動きベクトルがA 1 動きベクトルに等しくない場合、B 1 が空間的候補のリストに追加される (8 1 4)。

#### 【0 1 7 6】

「右上」B 0 動きベクトル (8 0 4) が利用可能である場合 (8 1 0)、「右上」の動きベクトルがB 1 動きベクトルと比較される (8 1 2)。B 0 動きベクトルがB 1 動きベクトルに等しい場合、B 0 動きベクトルは空間的候補のリストに追加されない (8 1 4)。逆に、B 0 動きベクトルがB 1 動きベクトルに等しくない場合、B 0 動きベクトルが空間的候補のリストに追加される (8 1 4)。

10

#### 【0 1 7 7】

「左下」A 0 動きベクトル (8 0 6) が利用可能である場合 (8 1 0)、「左下」の動きベクトルがA 1 動きベクトルと比較される (8 1 2)。A 0 動きベクトルがA 1 動きベクトルに等しい場合、A 0 動きベクトルは空間的候補のリストに追加されない (8 1 4)。逆に、A 0 動きベクトルがA 1 動きベクトルと等しくない場合、A 0 動きベクトルが空間的候補のリストに追加される (8 1 4)。

#### 【0 1 7 8】

空間的候補のリストが4つの候補を含まない場合、「左上」B 2 動きベクトル (8 0 8) の利用可能性がチェックされる (8 1 0)。利用可能であれば、A 1 動きベクトルおよびB 1 動きベクトルと比較される。B 2 動きベクトルがA 1 動きベクトルまたはB 1 動きベクトルに等しい場合、B 2 動きベクトルは空間的候補のリストに追加されない (8 1 4)。逆に、B 2 動きベクトルがA 1 動きベクトルまたはB 1 動きベクトルに等しくない場合、B 2 動きベクトルが空間的候補のリストに追加される (8 1 4)。

20

#### 【0 1 7 9】

この段階の最後に、空間的候補のリストは、4つまでの候補を含む。

#### 【0 1 8 0】

時間的候補については、2つの位置、コロケートされたブロックの右下の位置 (8 1 6)、図 6 a において示される H およびコロケートされたブロックの中央 (8 1 8) を使用することができる。これらの位置を図 6 a に示す。

30

#### 【0 1 8 1】

AMVP 動きベクトル導出処理の時間的動き予測子について図 7 に関連して説明したように、第1ステップは、H 位置におけるブロックの利用可能性をチェックすること (8 2 0) を目的とする。次に、それが利用可能でない場合、中央位置におけるブロックの利用可能性がチェックされる (8 2 0)。これらの位置の少なくとも1つの動きベクトルが利用可能である場合、時間的動きベクトルは、マージ動きベクトル予測子候補のリストに追加される時間的候補 (8 2 4) を作成するために、必要であれば、リスト L 0 および L 1 の両方について、インデックス 0 を有する参照フレームにスケーリングされ得る (8 2 2)。これは、リスト内の空間的候補の後に配置される。リスト L 0 および L 1 は、ゼロ、1つまたは複数の参照フレームを含む2つの参照フレームリストである。

40

#### 【0 1 8 2】

候補の数 (Nb\_Cand) が、候補の最大数より厳密に少ない場合 (8 2 6) (値がビットストリームスライスヘッダにおいてシグナリングされ、現在の HEC 設計において 5 に等しいと判断するための Max\_Cand 情報)、および現在のフレームが B タイプである場合、結合された候補が生成される (8 2 8)。結合された候補は、マージ動きベクトル予測子候補のリストの利用可能な候補に基づいて生成される。これは、主に、リスト L 0 の1つの候補の動き情報を、リスト L 1 の1つの候補の動き情報と組み合わせる (ペアリングする) ことからなる。

#### 【0 1 8 3】

候補の数 (Nb\_Cand) が候補の最大数 (Max\_Cand) よりも厳密に少ない

50

(830) 場合、マージ動きベクトル予測子候補のリストの候補の数が候補の最大数に達するまで、ゼロ動き候補が生成される(832)。

#### 【0184】

この処理の終わりに、マージ動きベクトル予測子候補のリストまたはセット(すなわち、古典的マージモードおよび古典的マージスキップモードであるマージモードの候補のリストまたはセット)が構築される(834)。図8に示すように、マージ動きベクトル予測子候補のリストまたはセットは、空間的候補のサブセット(800～808)および時間的候補のサブセット(816、818)から構築される(834)。

#### 【0185】

##### 代替時間的動きベクトル予測(ATMVP)

10

代替時間的動きベクトル予測(ATMVP)は、特別なタイプの動き補償である。時間的参照フレームからの現在のブロックについて1つの動き情報のみを考慮する代わりに、各コロケートされたブロックの各動き情報が考慮される。したがって、この時間的動きベクトル予測は、図9に示すように、各サブブロックの関連する動き情報を用いて、現在のブロックのセグメンテーションを与える。

#### 【0186】

VTMリファレンスソフトウェアでは、ATMVPがマージ候補のリスト(すなわち、古典的なマージモードおよび古典的なマージスキップモードであるマージモードの候補のリストまたはセット)に挿入されたマージ候補としてシグナリングされる。SPSレベルでATMVPがイネーブルされると、マージ候補の最大数は1だけ増加される。従って、5の代わりに6つの候補が考慮され、これは、このATMVPモードがディスエーブルされた場合であった。本発明の一実施形態によれば、ATMVPはアフィンマージ候補のリストに挿入されたアフィンマージ候補(例えば、ATMVP候補)(すなわち、以下でより詳細に説明される、アフィンマージモードのための候補の別個のリストまたはセット)としてシグナリングされ得ることが理解される。

20

#### 【0187】

さらに、この予測がSPSレベルでイネーブルされるとき、マージインデックスのすべてのビン(すなわち、マージ候補のリストから候補を識別するための識別子またはインデックスまたは情報)は、CABACによってコンテキスト符号化される。HEVC内にある間、またはATMVPがJEM内のSPSレベルでイネーブルされていないとき、第1ビン(first bin)のみがコンテキスト符号化され、残りのビンはコンテキストバイパス符号化される(すなわち、バイパスCABAC符号化される)。

30

#### 【0188】

図10(a)は、HEVC、またはJEMのSPSレベルでATMVPがイネーブルされていないときのマージインデックスの符号化を示す。これは、単項最大コードに対応する。さらに、この図10(a)では、第1番目のビット(first bit)はCABAC符号化され、他のビットはバイパスCABAC符号化される。

30

#### 【0189】

図10(b)は、ATMVPがSPSレベルでイネーブルされるときのマージインデックスの符号化を示す。すべてのビットがCABAC符号化される(1番目から5番目のビットまで)。インデックスを符号化するための各ビットは、それ自体のコンテキストを有し、言い換えれば、CABAC符号化に用いられるそれらの確率は分離されることに留意されたい。

40

#### 【0190】

##### アフィンモード

HEVCでは、動き補償予測(MCP)のために並進運動モデルのみを適用した。一方、現実世界では、ズームイン/ズームアウト、回転、パースペクティブモーション、および他の不規則なモーションなど、多くの種類のモーションがある。

#### 【0191】

JEMでは、簡易アフィン変換動き補償予測を適用し、2017年7月13日～21日

50

の T o r i n o における J V E T 会議で発表された J V E T - G 1 0 0 1 文書の抜粋に基づき、アフィンモードの一般原理を以下に説明する。この文書全体は、J E M で使用される他のアルゴリズムを説明する限り、参照により本明細書に組み込まれる。

【 0 1 9 2 】

図 1 1 ( a ) に示すように、この文書におけるブロックのアフィン動きフィールドは、2 つの制御点動きベクトルによって記述される（本発明の実施形態によれば、より多くの制御点動きベクトルを有するものなどの他のアフィンモデルを使用することもできることを理解されたい）。

【 0 1 9 3 】

ブロックの動きベクトルフィールド ( M V F ) は、以下の式によって記述される。

【 0 1 9 4 】

【 数 1 】

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

【 0 1 9 5 】

ここで、( v 0 x 、 v 0 y ) は、左上隅の制御点の動きベクトルであり、( v 1 x 、 v 1 y ) は、右上隅の制御点の動きベクトルである。また、w はブロック C u r ( 現在のブロック ) の幅である。

【 0 1 9 6 】

動き補償予測をさらに単純化するために、サブブロックベースのアフィン変換予測を適用した。サブブロックサイズ M × N は、式 2 のように導出され、ここで、M v P r e は、動きベクトル分数精度 ( J E M では 1 / 1 6 ) であり、( v 2 x , v 2 y ) は式 1 に従って計算された左上制御点の動きベクトルである。

【 0 1 9 7 】

【 数 2 】

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

【 0 1 9 8 】

式 2 によって導出された後、M および N は必要であれば、それぞれ w および h の除数になるように下方に調整されてもよい。h は、現在のブロック C u r ( 現在のブロック ) の高さである。

【 0 1 9 9 】

各 M × N サブブロックの動きベクトルを導出するために、図 1 1 ( b ) に示されるように、各サブブロックの中央サンプルの動きベクトルは、式 1 に従って計算され、1 / 1 6 分数精度に丸められる。次に、動き補償補間フィルタを適用して、導出した動きベクトルを持つ各サブブロックの予測を生成する。

【 0 2 0 0 】

アフィンモードは、インターモード ( A M V P 、 「古典的」マージ、または「古典的」マージスキップ ) のような動き補償モードである。その原理は、2 つ又は 3 つの隣接する動き情報に従って画素毎に 1 つの動き情報を生成することである。J E M では、アフィンモードが図 1 1 ( b ) に示すように、各 4 × 4 ブロックについて 1 つの動き情報を導出する（各正方形は 4 × 4 ブロックであり、図 1 1 ( b ) のブロック全体は 4 × 4 サイズのこ

10

20

30

40

50

のような正方形の 16 ブロックに分割された  $16 \times 16$  ブロックであり、各  $4 \times 4$  正方形ブロックはそれに関連する動きベクトルを有する）。本発明の実施形態では、アフィンモードが、1つの動き情報を導出することができる限り、異なるサイズまたは形状のブロックについて1つの動き情報を駆動することができることを理解されたい。

#### 【0201】

一実施形態によれば、このモードは、フラグでアフィンモードをイネーブルすることによって、AMVPモードおよびマージモード（すなわち、「非アフィンマージモード」とも呼ばれる古典的マージモード、および「非アフィンマージスキップモード」とも呼ばれる古典的マージスキップモード）に利用可能になる。このフラグはCABA符号化される。一実施形態では、コンテキストが左ブロック（図6bの位置A2）および左上ブロック（図6bの位置B3）のアフィンフラグの合計に依存する。

10

#### 【0202】

したがって、JEMでは、後の式で与えられるアフィンフラグに対して3つのコンテキスト変数（0、1、または2）が可能である。

#### 【0203】

$$C_{tx} = IsAffine(A2) + IsAffine(B3)$$

ここで、`IsAffine(block)`は、ブロックがアフィンブロックでない場合に0を返し、ブロックがアフィンである場合に1を返す関数である。

#### 【0204】

##### アフィンマージ候補導出

JEMでは、サブブロック（マージ）モードとしても知られるアフィンマージモード（またはアフィンマージスキップモード）が位置A1、B1、B0、A0、B2のブロックの中でアフィンである第1隣接ブロック（first neighbouring block）（すなわち、アフィンモードを使用して符号化される第1隣接ブロック）から現在のブロックの動き情報を導出する。これらの位置を図6aおよび6bに示す。しかしながら、アフィンパラメータがどのように導出されるかは完全には定義されておらず、本発明は例えば、アフィンマージモードのアフィンパラメータを定義することによって、少なくともこの態様を改善することを目的とし、その結果、アフィンマージ候補のためのより広い選択の選択を可能にする（すなわち、アフィンである第1隣接ブロックだけでなく、インデックスなどの識別子を用いた選択のために少なくとも1つの他の候補が利用可能である）。

20

#### 【0205】

例えば、本発明のいくつかの実施形態によれば、アフィンマージ候補（アフィンモードのための動き情報を導出／取得するための候補）のそれ自体のリストを有するアフィンマージモードと、（アフィンマージ候補のリストから1つのアフィンマージ候補を識別するための）アフィンマージインデックスとが、ブロックを符号化または復号するために使用される。

30

#### 【0206】

##### アフィンマージシグナリング

図12は、アフィンマージモードの使用をシグナリングするための符号化モードに関するいくつかのシンタックス要素の部分復号処理のフローチャートである。この図では、スキップフラグ（1201）、予測モード（1211）、マージフラグ（1203）、マージインデックス（1208）、およびアフィンフラグ（1206）を復号することができる。

40

#### 【0207】

インターライズ内のすべてのCUについて、スキップフラグが復号される（1201）。CUがスキップでない場合（1202）、予測モード（予測モード）が復号される（1211）。このシンタックス要素は、現在のCUがインターモードまたはイントラモードで符号化されている（復号される）かどうかを示す。CUがスキップである場合（1202）、その現在のモードはインターモードであることに留意されたい。CUがスキップでない場合（1202:No）、CUはAMVPモードまたはマージモードで符号化され

50

る。CUがインターの場合(1212)、マージフラグが復号される(1203)。CUがマージである場合(1204)、またはCUがスキップである場合(1202: Yes)、アフィンフラグ(1206)を復号する必要があるかどうか、すなわち(1205)で、現在のCUがアフィンモードで符号化されていた可能性があるかどうかの判定が行われるかどうかが検証/チェックされる(1205)。このフラグは現在のCUが $2N \times 2N$  CUである場合に復号され、これは、現在のVVCにおいて、CUの高さ及び幅が等しいことを意味する。さらに、少なくとも1つの隣接するCU A1またはB1またはB0またはA0またはB2は、アフィンモード(アフィンマージモードまたはアフィンモードがイネーブルされたATMVPモードのいずれか)で符号化されなければならない。最終的に、現在のCUは $4 \times 4$  CUではなく、デフォルトではCU  $4 \times 4$ はVTMリファレンスソフトウェアで無効になる。この条件(1205)が偽である場合、現在のCUがHEVCで指定されているような古典的マージモード(または古典的マージスキップモード)で符号化されていることが確実であり、マージインデックスが復号される(1208)。アフィンフラグ(1206)が1に等しく設定される場合(1207)、CUはマージアフィンCU(すなわち、アフィンマージモードで符号化されたCU)またはマージスキップアフィンCU(すなわち、アフィンマージスキップモードで符号化されたCU)であり、マージインデックス(1208)は復号される必要がない(アフィンマージモードが使用される、すなわち、CUは、アフィンである第1隣接ブロックを有するアフィンモードを使用して復号されるため)。そわない場合、現在のCUは古典的な(基本的である)マージまたはマージスキップCU(すなわち、古典的なマージまたはマージスキップモードで符号化されたCU)であり、マージ候補インデックス(1208)が復号される。

#### 【0208】

##### マージ候補導出

図13は、一実施形態に係る、マージ候補(すなわち、古典的マージモードまたは古典的マージスキップモードの候補)導出を示すフローチャートである。この導出は、図8に示すマージモードの動きベクトル導出処理(すなわち、HEVCのマージ候補リスト導出)の上に構築されたものである。HEVCと比較した主な変更は、ATMVP候補(1319、1321、1323)の追加、候補の全重複チェック(1325)、および候補の新しい順序である。ATMVP予測は、現在のCUのいくつかの動き情報を表すので、専用の候補として設定される。第1サブブロック(first sub-block)(左上)の値は、時間的候補と比較され、時間的候補はそれらが等しい場合、マージ候補のリストに追加されない(1320)。ATMVP候補は、他の空間的候補と比較されない。これは、リスト内に既にある各空間的候補と比較され(1325)、重複候補であればマージ候補リストに追加されない時間的候補とは対照的である。

#### 【0209】

空間的候補がリスト内に追加されると、それはリスト内の他の空間的候補と比較され(1312)、これはHEVCの最終バージョンの場合ではない。

#### 【0210】

現在のVTMバージョンでは、マージ候補のリストが、符号化テスト条件に対して最良の結果を提供すると判定されたときに、以下の順序で設定される。

- A1
- B1
- B0
- A0
- ATMVP
- B2
- 時間的
- 組み合わせ
- Z e r o \_ M V

空間的候補B2は、ATMVP候補の後に設定されることに留意することが重要である。

10

20

30

40

50

**【0211】**

さらに、ATMVPがスライスレベルでイネーブルされるとき、候補のリスト内の最大数は、HEVCの5ではなく6である。

**【0212】****その他のインター予測モード**

以下で説明する最初のいくつかの実施形態（第16の実施形態まで）では、その説明が（正規の）マージモードおよびアフィンマージモードのためのインデックスの符号化または復号を説明する。開発中のVVC規格の最近のバージョンでは、（正規の）マージモードおよびアフィンマージモードに加えて、追加のインター予測モードも考慮されている。現在考慮されているこのような追加のインター予測モードは、以下に説明する、Multi-hypothesis Intra Inter (MHII) マージモード、トライアングルマージモード、および動きベクトル差のマージ (MMVD) マージモードである。

10

**【0213】**

これらの第1のいくつかの実施形態の変形形態によれば、マージモードまたはアフィンマージモードに加えて、またはその代わりに、追加のインター予測モードのうちの1つまたは複数を使用することができ、追加のインター予測モードのうちの1つまたは複数のためのインデックス（またはフラグまたは情報）を、マージモードまたはアフィンマージモードのうちのいずれか1つと同じ技法を使用してシグナリング（符号化または復号）することができるなどを理解されたい。

**【0214】**

20

**MHII (Multi-Hypothesis Intra Inter) マージモード**

MHII (Multi-Hypothesis Intra Inter) マージモードは、正規のマージモードとイントラモードとを組み合わせたハイブリッドである。このモードのブロック予測子は、（正規の）マージモードブロック予測子とイントラモードブロック予測子との間の平均として得られる。得られたブロック予測子は、再構成されたブロックを得るために、現在のブロックの残差に加算される。このマージモードブロック予測子を得るために、MHII マージモードは、マージモードと同じ数の候補を使用し、同じマージ候補導出処理を使用する。したがって、MHII マージモードのためのインデックスシグナリングは、マージモードのためのインデックスシグナリングと同じ技法を使用することができる。また、このモードは、非スキップモードで符号化／復号されたブロックに対してのみ有効になる。したがって、現在のCUIがスキップモードで符号化／復号されると、MHIIは符号化／復号処理で使用できなくなる。

30

**【0215】****トライアングルマージモード**

トライアングルマージモードは、三角形形状に基づく動き補償を使用する双予測モードの一種である。図25(a)および図25(b)は、そのブロック予測子生成に使用される異なるパーティション構成を示す。ブロック予測子は、ブロック内の第1の三角形（第1のブロック予測子2501または2511）および第2の三角形（第2のブロック予測子2502または2512）から得られる。このブロック予測子の生成には、2つの異なる構成が使用される。第1のものについては、（2つのブロック予測子候補が関連付けられる）三角形部分／領域間の分割／分割が図25(a)に示されるように、左上隅から右下隅までである。第2のものでは、（2つのブロック予測子候補が関連付けられる）三角形領域間の分割／分割が図25(b)に示すように、右上隅から左下隅までである。さらに、三角形領域間の境界の周りのサンプルは重みがサンプル位置（例えば、境界からの距離）に依存する加重平均でフィルタリングされる。独立したトライアングルマージ候補リストが生成され、トライアングルマージモードのためのインデックスシグナリングは、マージモードまたはアフィンマージモードにおけるインデックスシグナリングのための技術に対して、それに応じて修正された技術を使用することができる。

40

**【0216】****動きベクトル差のマージ (MMVD) マージモード**

50

MMVD マージモードは、独立した MMVD マージ候補リストを生成する正規のマージモード候補導出の特別なタイプである。現在の CU に対する選択された MMVD マージ候補は、MMVD マージ候補の 1 つの動きベクトル成分 (mvx または mvy) にオフセット値を加算することによって得られる。オフセット値は、第 1 番目のリスト L0 または第 2 番目のリスト L1 からの動きベクトルの構成要素に、これらの参照フレームの構成 (例えば、両後方、前方または前方および後方の両方) に応じて加算される。選択された MMVD マージ候補は、インデックスを使用してシグナリングされる。オフセット値は、8 つの可能なプリセット距離 (1/4-pe1, 1/2-pe1, 1-pe1, 2-pe1, 4-pe1, 8-pe1, 16-pe1, 32-pe1) 間の距離インデックスと、x 軸または y 軸とオフセットの符号を与える方向インデックスを使用してシグナリングされる。したがって、MMVD マージモードのためのインデックスシグナリングは、マージモード、またはアフィンマージモードのためのインデックスシグナリングと同じ技法を使用することができます。

#### 【0217】

##### 実施形態

本発明の実施形態を、残りの図面を参照して説明する。実施形態は特に明記しない限り組み合わされてもよく、例えば、実施形態の特定の組合せは複雑さを増しながら符号化効率を改善してもよいが、これは特定の使用事例では許容可能であり得ることに留意されたい。

#### 【0218】

##### 第 1 の実施形態

上述のように、VTM リファレンスソフトウェアでは、ATMVP がマージ候補のリストに挿入されたマージ候補としてシグナリングされる。ATMVP は、(SPS レベルで) シーケンス全体に対してイネーブルまたはディスエーブルすることができる。ATMVP がディスエーブルされると、マージ候補の最大数は 5 である。ATMVP がイネーブルされると、マージ候補の最大数は、5 から 6 に 1 だけ増加される。

#### 【0219】

エンコーダでは、図 13 の方法を使用してマージ候補のリストが生成される。1 つのマージ候補が例えばレート歪み基準に基づいてマージ候補のリストから選択される。選択されたマージ候補は、マージインデックスと呼ばれるシンタックス要素を使用して、ビットストリーム内のデコーダにシグナリングされる。

#### 【0220】

現在の VTM リファレンスソフトウェアでは、ATMVP がイネーブルされているかディスエーブルされているかによって、マージインデックスを符号化する方法が異なる。

#### 【0221】

図 10 (a) は、ATMVP が SPS レベルでイネーブルされないときのマージインデックスの符号化を示す。5 つのマージ候補 C and 0, C and 1, C and 2, C and 3、および C and 4 は、それぞれ、0, 10, 110, 1110、および 1111 に符号化される。これは、単項最大符号化に対応する。さらに、第 1 番目のビットは、単一のコンテキストを使用して CABAC によって符号化され、他のビットはバイパス符号化される。

#### 【0222】

図 10 (b) は、ATMVP がイネーブルされたときのマージインデックスの符号化を示す。6 つのマージ候補 C and 0, C and 1, C and 2, C and 3, C and 4、および C and 5 は、それぞれ、0, 10, 110, 1110, 11110、および 1111 に符号化される。この場合、マージインデックスのすべてのビット (1 番目から 5 番目のビットまで) が CABAC によってコンテキスト符号化される。各ビットはそれ自体のコンテキストを有し、異なるビットに対して別々の確率モデルが存在する。

#### 【0223】

本発明の第 1 の実施形態では図 14 に示すように、マージ候補のリストにマージ候補と

して A T M V P が含まれている場合（例えば、S P S レベルで A T M V P がイネーブルされている場合）、マージインデックスの符号化は、マージインデックスの第 1 番目のビットのみが単一のコンテキストを使用して C A B A C によって符号化されるように修正される。コンテキストは、A T M V P が S P S レベルでイネーブルされていない場合、現在の V T M リファレンスソフトウェアと同じ方法で設定され、すなわち、他のビット（2 番目から 5 番目まで）は、バイパス符号化される。マージ候補のリストにマージ候補として A T M V P が含まれていない場合（例えば、S P S レベルで A T M V P がディスエーブルされている場合）、5 つのマージ候補が存在する。マージインデックスの第 1 番目のビットのみが、単一のコンテキストを使用して C A B A C によって符号化される。コンテキストは、A T M V P が S P S レベルでイネーブルされていない場合、現在の V T M リファレンスソフトウェアと同じ方法で設定される。他のビット（2 番目から 4 番目のビットまで）は、バイパス復号される。

#### 【 0 2 2 4 】

デコーダは、エンコーダと同じマージ候補のリストを生成する。これは、図 13 の方法を使用することによって達成することができる。A T M V P がマージ候補のリストにマージ候補として含まれていない場合（例えば、A T M V P が S P S レベルでディスエーブルされている場合）、5 つのマージ候補がある。マージインデックスの第 1 番目のビットのみが、単一のコンテキストを使用して C A B A C によって復号される。他のビット（2 番目から 4 番目のビットまで）は、バイパス復号される。現在のリファレンスソフトウェアとは対照的に、A T M V P がマージ候補のリストにマージ候補として含まれる場合（例えば、A T M V P が S P S レベルでイネーブルされる場合）、マージインデックスの復号において単一のコンテキストを使用して、マージインデックスの第 1 番目のビットのみが C A B A C によって復号される。他のビット（2 番目から 5 番目のビットまで）は、バイパス復号される。復号されたマージインデックスは、マージ候補のリストの中からエンコーダによって選択されたマージ候補を識別するために使用される。

#### 【 0 2 2 5 】

V T M 2 . 0 リファレンスソフトウェアと比較したこの実施形態の利点は、符号化効率に影響を与えることなく、マージインデックス復号およびデコーダ設計（およびエンコーダ設計）の複雑さが低減されることである。実際、この実施形態では、現在の V T M マージインデックス符号化 / 復号のために 5 ではなく、1 つの C A B A C 状態のみがマージインデックスのために必要とされる。さらに、他のビットは C A B A C バイパス符号化され、C A B A C で全てのビットを符号化するのと比較して演算の数を減らすので、最悪の場合の複雑さを減らす。

#### 【 0 2 2 6 】

##### 第 2 の実施形態

第 2 の実施形態では、マージインデックスのすべてのビットは C A B A C 符号化されるが、それらはすべて同じコンテキストを共有する。この場合、ビット間で共有される、第 1 の実施形態のような単一のコンテキストが存在し得る。その結果、マージ候補のリストにマージ候補として A T M V P が含まれている場合（例えば、S P S レベルで A T M V P がイネーブルされている場合）、V T M 2 . 0 リファレンスソフトウェアでは 5 であるのに対して、1 つのコンテキストのみが使用される。V T M 2 . 0 リファレンスソフトウェアと比較したこの実施形態の利点は、符号化効率に影響を与えることなく、マージインデックス復号およびデコーダ設計（およびエンコーダ設計）の複雑さが低減されることである。

#### 【 0 2 2 7 】

あるいは以下に第 3 から第 16 の実施形態に関連して説明するように、コンテキスト変数は、2 つ以上のコンテキストが利用可能であるが、現在のコンテキストがビットによって共有されるように、ビット間で共有されてもよい。

#### 【 0 2 2 8 】

A T M V P がディスエーブルされるとき、同じコンテキストが全てのビットに対して依

然として使用される。

【0229】

ATMVPが利用可能なモードでないか、またはディスエーブルされている場合であっても、この実施形態および以降のすべての実施形態を適用することができる。

【0230】

第2の実施形態の変形では、マージインデックスの任意の2つ以上のビットがCABA C符号化され、同じコンテキストを共有する。マージインデックスの他のビットは、バイパス符号化される。例えば、マージインデックスの最初のNビットは、CABA C符号化されてもよく、ここで、Nは2以上である。

【0231】

第3の実施形態

第1の実施形態では、マージインデックスの第1番目のビットが単一のコンテキストを使用してCABA C符号化された。

【0232】

第3の実施形態では、マージインデックスのビットのコンテキスト変数が隣接ブロックのマージインデックスの値に依存する。これにより、各コンテキストがコンテキスト変数の異なる値に対応する、ターゲットビットに複数のコンテキストが可能になる。

【0233】

隣接ブロックは、すでに復号されている任意のブロックであってもよく、その結果、そのマージインデックスは、現在のブロックが復号されている時間までにデコーダに利用可能である。例えば、隣接ブロックは、図6bに示されるブロックA0、A1、A2、B0、B1、B2、およびB3のいずれかであってもよい。

【0234】

第1の変形例では、第1番目のビットだけがこのコンテキスト変数を使用してCABA C符号化される。

【0235】

第2の変形例では、マージインデックスの最初のNビット、ここでNは2以上である、はCABA C符号化され、コンテキスト変数は、これらのNビット間で共有される。

【0236】

第3の変形例では、マージインデックスの任意のNビット、ここでNは2以上である、がCABA C符号化され、コンテキスト変数が、これらのNビット間で共有される。

【0237】

第4の変形例では、マージインデックスの最初のNビット、ここでNは2以上である、はCABA C符号化され、N個のコンテキスト変数がこれらのNビットに使用される。コンテキスト変数がK個の値を有すると仮定すると、K × N個のCABA C状態が使用される。例えば、本実施形態では、1つの隣接ブロックを用いて、コンテキスト変数は2つの値、例えば、0及び1を都合よく有することができる。換言すれば、2N個のCABA C状態が使用される。

【0238】

第5の変形例では、マージインデックスの任意のNビット、ここで、Nは2以上である、は適応PM符号化され、N個のコンテキスト変数がこれらのNビットのために使用される。

【0239】

同様の変形例は、以下に説明する第4～第16の実施形態に適用可能である。

【0240】

第4の実施形態

第4の実施形態では、マージインデックスのビットのコンテキスト変数が、2つ以上の隣接ブロックのマージインデックスのそれぞれの値に依存する。例えば、第1隣接ブロックは左ブロックA0、A1またはA2であり、第2隣接ブロック(second neighbouring block)は上ブロックB0、B1、B2またはB3である。2つ以上のマージインデック

10

20

30

40

50

ス値を組み合わせる方法は、特に限定されない。例を以下に示す。

【0241】

コンテキスト変数は便宜上、2つの隣接ブロックが存在するので、この場合、3つの異なる値、例えば、0、1、および2を有することができる。したがって、第3の実施形態に関連して説明した第4の変形例が、3つの異なる値を有するこの実施形態に適用される場合、Kは2ではなく3である。換言すれば、3N個のCABA'C状態が使用される。

【0242】

第5の実施形態

第5の実施形態では、マージインデックスのビットのコンテキスト変数が、隣接ブロックA2およびB3のマージインデックスのそれぞれの値に依存する。

10

【0243】

第6の実施形態

第6の実施形態では、マージインデックスのビットのコンテキスト変数が、隣接ブロックA1およびB1のマージインデックスのそれぞれの値に依存する。この変形例の利点は、マージ候補導出とのアラインメントである。その結果、いくつかのデコーダおよびエンコーダの実装では、メモリアクセスの削減を達成することができる。

【0244】

第7の実施形態

第7の実施形態では、現在のブロックのマージインデックスにおけるビット位置 $idx\_num$ を有するビットのコンテキスト変数が、以下の式に従って得られる。

20

【0245】

$ctxIdx = (MergeIndexLeft == idx\_num) + (MergeIndexUp == idx\_num)$

ここで、`MergeIndexLeft`は左ブロックのマージインデックス、`MergeIndexUp`は上ブロックのマージインデックス、シンボル`==`は等価シンボルである。

【0246】

例えば、6つのマージ候補がある場合、 $0 \leq idx\_num \leq 5$ である。

【0247】

左ブロックはブロックA1であり、上ブロックはブロックB1である（第6の実施形態と同様）。あるいは、左ブロックがブロックA2であり、上ブロックがブロックB3であってもよい（第5の実施形態と同様）。

30

【0248】

左ブロックのマージインデックスが`idx\_num`と等しい場合、式`(MergeIndexLeft == idx\_num)`は1に等しい。次の表は、この式`(MergeIndexLeft == idx\_num)`の結果を示す。

【0249】

40

50

【表 2】

Merge_index_left	idx_num				
	0	1	2	3	4
0	1	0	0	0	0
1	0	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	0
4	0	0	0	0	1
5	0	0	0	0	0

【0 2 5 0】

もちろん、式のテーブル (Merge\_index\_left == idx\_num) は同じである。

【0 2 5 1】

次の表は、各マージインデックス値の単項最大コードと、各ビットの相対ビット位置と、を示す。この表は、図 10 (b) に対応する。

【0 2 5 2】

【表 3】

Merge_index_left	単項最大コード				
	0	1	2	3	4
0	0				
1	1	0			
2	1	1	0		
3	1	1	1	0	
4	1	1	1	1	0
5	1	1	1	1	1

【0 2 5 3】

左ブロックがマージブロックまたはアフィンマージブロックでない（すなわち、アフィンマージモードを使用して符号化されている）場合、左ブロックは利用可能ではないと考えられる。上ブロックについても同様の条件が適用される。

【0 2 5 4】

例えば、第 1 番目のビットのみが C A B A C 符号化される場合、コンテキスト変数  $c_t \times I_d \times$  は、

左上のブロックがマージインデックスを有さない場合、または左ブロックマージインデックスが第 1 インデックス (first index) ではない（すなわち 0 ではない）場合、および上ブロックマージインデックスが第 1 インデックスではない（すなわち 0 ではない）場合は 0

左ブロックおよび上ブロックのうちの一方であって他方ではないブロックが第 1 インデックスに等しいマージインデックスを有する場合は 1

左ブロックおよび上ブロックのそれぞれについてマージインデックスが第 1 インデックスに等しい場合は 2  
に等しく設定される。

#### 【0255】

より一般的には、C A B A C 符号化された位置  $idx\_num$  のターゲットビットの場合、コンテキスト変数  $ctxIdx$  は、

左上のブロックがマージインデックスを有さない場合、または左ブロックマージインデックスが  $i$  番目のインデックスでない場合 ( $i = idx\_num$  の場合)、および上ブロックマージインデックスが  $i$  番目のインデックスでない場合は 0

左ブロックと上ブロックの一方であって他方ではないブロックが  $i$  番目のインデックスと等しいマージインデックスを有する場合は 1

左ブロックと上ブロックのそれぞれについてマージインデックスが  $i$  番目のインデックスと等しい場合は 2

に等しく設定される。ここで、 $i$  番目のインデックスは、 $i = 0$  の場合は第 1 インデックス (first index) を意味し、 $i = 1$  の場合は第 2 インデックス (second index) を、  
というように意味する。

#### 【0256】

##### 第 8 の実施形態

第 8 の実施形態では、現在のブロックのマージインデックスにおけるビット位置  $idx\_num$  を有するビットのコンテキスト変数が、以下の式に従って得られる。

#### 【0257】

$ctx = (Merge\_index\_left > idx\_num) + (Merge\_index\_up > idx\_num)$  ここで、 $Merge\_index\_left$  は左ブロックのマージインデックス、 $Merge\_index\_up$  は上ブロックのマージインデックス、シンボル  $>$  は「より大きい」を意味する。

#### 【0258】

例えば、6 つのマージ候補がある場合、 $0 \leq idx\_num \leq 5$  である。

#### 【0259】

左ブロックはブロック A 1 であり、上ブロックはブロック B 1 である (第 6 の実施形態と同様)。あるいは、左ブロックがブロック A 2 であり、上ブロックがブロック B 3 であつてもよい (第 5 の実施形態と同様)。

#### 【0260】

左ブロックのマージインデックスが  $idx\_num$  より大きい場合、式 ( $Merge\_index\_left > idx\_num$ ) は 1 に等しい。左ブロックがマージブロックまたはアフィンマージブロックでない (すなわち、アフィンマージモードを使用して符号化されている) 場合、左ブロックは利用可能ではないと考えられる。上ブロックについても同様の条件が適用される。

#### 【0261】

次の表は、この式 ( $Merge\_index\_left > idx\_num$ ) の結果を示す。

#### 【0262】

10

20

30

40

50

【表 4】

Merge_index_left	idx_num				
	0	1	2	3	4
0	0	0	0	0	0
1	1	0	0	0	0
2	1	1	0	0	0
3	1	1	1	0	0
4	1	1	1	1	0
5	1	1	1	1	1

## 【0 2 6 3】

例えば、第 1 番目のビットのみが C A B A C 符号化される場合、コンテキスト変数  $c_t \times I_d \times$  は、

左上のブロックがマージインデックスを有さない場合、または左ブロックマージインデックスが第 1 インデックス以下（すなわち 0 でない）である場合、および上ブロックマージインデックスが第 1 インデックス以下（すなわち 0 でない）である場合は 0

左ブロックと上ブロックの一方であって他方ではないブロックが第 1 インデックスより大きいマージインデックスを有する場合は 1

左ブロックと上ブロックの各々についてマージインデックスが第 1 インデックスより大きい場合は 2

に等しく設定される。

## 【0 2 6 4】

より一般的には、C A B A C 符号化された位置  $i_d \times n_u m$  のターゲットビットの場合、コンテキスト変数  $c_t \times I_d \times$  は、

左上のブロックがマージインデックスを有さない場合、または左ブロックマージインデックスが  $i$  番目のインデックスより小さい場合 ( $i = i_d \times n_u m$  の場合)、および上ブロックマージインデックスが  $i$  番目のインデックスより小さいか等しい場合は 0

左ブロックと上ブロックの一方であって他方ではないブロックが  $i$  番目のインデックスより大きいマージインデックスを有する場合は 1

左ブロックと上ブロックの各々についてマージインデックスが  $i$  番目のインデックスより大きい場合は 2

に等しく設定される。

## 【0 2 6 5】

第 8 の実施形態は、第 7 の実施形態に対してさらに符号化効率を向上させる。

## 【0 2 6 6】

## 第 9 の実施形態

第 4 ~ 第 8 の実施形態では、現ブロックのマージインデックスのビットのコンテキスト変数が 2 つ以上の隣接ブロックのマージインデックスのそれぞれの値に依存した。

## 【0 2 6 7】

第 9 の実施形態では、現在のブロックのマージインデックスのビットのコンテキスト変数が、2 つ以上の隣接ブロックのそれぞれのマージフラグに依存する。例えば、第 1 隣接ブロックは左ブロック A 0、A 1 または A 2 であり、第 2 隣接ブロックは上ブロック B 0、B 1、B 2 または B 3 である。

## 【0 2 6 8】

マージフラグは、ブロックがマージモードを使用して符号化されている場合は 1 にセッ

トされ、スキップモードやアフィンマージモードなどの他のモードが使用されている場合は 0 にセットされる。V M T 2 . 0 では、アフィンマージが基本モードまたは「古典的」マージモードとは別個のモードであることに留意されたい。アフィンマージモードは、専用のアフィンフラグを使用してシグナリングすることができる。あるいは、マージ候補のリストがアフィンマージ候補を含んでもよく、その場合、アフィンマージモードが選択され、マージインデックスを使用してシグナリングされてもよい。

#### 【 0 2 6 9 】

その後、コンテキスト変数は、

左隣接ブロックも上隣接ブロックも、1 にセットされたそのマージフラグを有さない場合には 0

左および上の隣接ブロックの一方であって他方ではない隣接ブロックが 1 に設定されたそのマージフラグを有する場合は 1

左および上の隣接ブロックの各々が 1 に設定されたそのマージフラグを有する場合は 2 に設定される。

#### 【 0 2 7 0 】

この単純な評価は、V T M 2 . 0 に対して符号化効率の改善を達成する。他の利点は、第 7 および第 8 の実施形態と比較して、隣接ブロックのマージインデックスではなくマージフラグのみがチェックされる必要があるので、より低い複雑さである。

#### 【 0 2 7 1 】

変形例では、現在のブロックのマージインデックスのビットのコンテキスト変数が単一の隣接ブロックのマージフラグに依存する。

#### 【 0 2 7 2 】

第 1 0 の実施形態

第 3 から第 9 の実施形態では、現在のブロックのマージインデックスのビットのコンテキスト変数が、1 つまたは複数の隣接ブロックのマージインデックス値またはマージフラグに依存した。

#### 【 0 2 7 3 】

第 1 0 の実施形態では、現在のブロックのマージインデックスのビットのコンテキスト変数が、現在のブロック（現在の符号化ユニット、つまり C U ）のためのスキップフラグの値に依存する。スキップフラグは、現在のブロックがマージスキップモードを使用する場合は 1 に等しく、それ以外の場合は 0 に等しくなる。

#### 【 0 2 7 4 】

スキップフラグは、現在のブロックに対してすでに復号または解析されている別の変数またはシンタックス要素の第 1 の例である。この他の変数またはシンタックス要素は、好みしくは現在のブロックにおける動き情報の複雑さのインジケータである。マージインデックス値の出現は動き情報の複雑さに依存するので、スキップフラグのような変数またはシンタックス要素は一般にマージインデックス値と相関する。

#### 【 0 2 7 5 】

より具体的には、マージスキップモードが一般に、静止シーンまたは一定の動きを伴うシーンに対して選択される。その結果、マージインデックス値は一般に、ブロック残差を含むインター予測を符号化するために使用される古典的なマージモードよりもマージスキップモードの方が低い。これは、一般に、より複雑な動きに対して生じる。しかしながら、これらのモード間の選択は、量子化及び / 又は R D 基準にも関連することが多い。

#### 【 0 2 7 6 】

この単純な評価は、V T M 2 . 0 に対して符号化効率を向上させる。また、それは、隣接ブロック又はマージインデックス値をチェックすることを伴わないので、実施するのが非常に簡単である。

#### 【 0 2 7 7 】

第 1 の変形例では、現在のブロックのマージインデックスのビットのコンテキスト変数が、単に現在のブロックのスキップフラグに等しく設定される。ビットは、第 1 番目のビ

10

20

30

40

50

ットのみにことができる。他のビットは、第1の実施形態のようにバイパス符号化される。

#### 【0278】

第2の変形例では、マージインデックスのすべてのビットがC A B A C符号化され、それらの各々はマージフラグに応じてそれ自体のコンテキスト変数を有する。これは、(6つのマージ候補に対応する)マージインデックス内に5つのC A B A C符号化ビットがあるときに10の確率状態を必要とする。

#### 【0279】

第3の変形例では、状態の数を制限するために、マージインデックスのNビットのみがC A B A C符号化され、ここで、Nは2以上、例えば最初のNビットである。これは、2Nの状態を必要とする。例えば、最初の2ビットがC A B A C符号化される場合、4つの状態が必要とされる。

#### 【0280】

一般に、スキップフラグの代わりに、現在のブロックに対して既に復号または解析されていて、現在のブロックにおけるモーション情報の複雑さのインジケータである任意の他の変数またはシンタックス要素を使用することが可能である。

#### 【0281】

##### 第11の実施形態

第11の実施形態は、図11(a)、図11(b)および図12を参照して前述したアフィンマージシグナリングに関する。

#### 【0282】

第11の実施形態では、現在のブロック(現在のC U)のマージインデックスのC A B A C符号化ビットのコンテキスト変数が、もしあれば、マージ候補のリスト内のアフィンマージ候補に依存する。このビットは、マージインデックスの第1番目のビットのみ、あるいは最初のNビット、ここでNは2以上、あるいは任意のNビット、であることがある。他のビットはバイパス符号化される。

#### 【0283】

アフィン予測は、複雑な動きを補償するために設計されている。したがって、複雑な動きの場合、マージインデックスは一般に、それほど複雑でない動きの場合よりも高い値を有する。その結果、第1アフィンマージ候補(first Affine Merge candidate)がリストのかなり下にある場合、またはアフィンマージ候補が全くない場合、現在のC Uのマージインデックスは、小さい値を有する可能性がある。

#### 【0284】

したがって、コンテキスト変数は、リスト内の少なくとも1つのアフィンマージ候補の存在および/または位置に依存することが有効である。

#### 【0285】

例えば、コンテキスト変数は、A 1がアフィンの場合は1、B 1がアフィンの場合は2、B 0がアフィンの場合は3、A 0がアフィンの場合は4、B 2がアフィンの場合は5、隣接ブロックがアフィンでない場合は0に等しく設定される。

#### 【0286】

現在のブロックのマージインデックスが復号または解析されるとき、これらの位置のマージ候補のアフィンフラグはすでにチェックされている。したがって、現在のブロックのマージインデックスのコンテキストを導出するために、それ以上のメモリアクセスは必要ない。

#### 【0287】

この実施形態は、V T M 2 . 0に対して符号化効率を向上させる。ステップ1205は、すでに隣接C Uアフィンモードをチェックすることを含むので、追加のメモリアクセスは必要とされない。

#### 【0288】

第1の変形例では、状態の数を制限するために、コンテキスト変数は、

10

20

30

40

50

隣接ブロックがアフィンでない場合、または A 1 または B 1 がアフィンである場合は 0、B 0、A 0、または B 2 がアフィンの場合は 1 に等しく設定される。

#### 【 0 2 8 9 】

第 2 の変形例では、状態の数を制限するために、コンテキスト変数は、隣接ブロックがアフィンでない場合は 0、A 1 または B 1 がアフィンの場合は 1、B 0、A 0、または B 2 がアフィンの場合は 2 に等しく設定される。

#### 【 0 2 9 0 】

第 3 の変形例では、コンテキスト変数は、A 1 がアフィンの場合は 1、B 1 がアフィンの場合は 2、B 0 がアフィンの場合は 3、A 0 または B 2 がアフィンの場合は 4、隣接ブロックがアフィンでない場合は 0 に等しく設定される。 10

#### 【 0 2 9 1 】

これらの位置は、アフィンフラグ復号がこれらの位置に依存するので、マージインデックスが復号または構文解析されるときに既にチェックされていることに留意されたい。従って、アフィンフラグの後に符号化されるマージインデックスコンテキストを導出するために、追加のメモリアクセスは必要ない。

#### 【 0 2 9 2 】

##### 第 1 2 の実施形態

第 1 2 の実施形態では、アフィンモードをシグナリングすることは、候補動き予測子としてアフィンモードを挿入することを含む。 20

#### 【 0 2 9 3 】

第 1 2 の実施形態の一例では、アフィンマージ（およびアフィンマージスキップ）がマージ候補として（すなわち、古典的マージモードまたは古典的マージスキップモードと共に使用するためのマージ候補のうちの 1 つとして）シグナリングされる。この場合、図 1 2 のモジュール 1 2 0 5、1 2 0 6、および 1 2 0 7 は除去される。さらに、マージモードの符号化効率に影響を与えないように、マージ候補の最大可能数がインクリメントされる。例えば、現在の VTM バージョンでは、この値は 6 に等しく設定され、したがって、この実施形態を VTM の現在のバージョンに適用する場合、値は 7 になる。

#### 【 0 2 9 4 】

利点は、復号する必要があるシンタックス要素が少ないため、マージモードのシンタックス要素の設計簡素化である。状況によっては、符号化効率の改善 / 变化が観察され得る。 30

#### 【 0 2 9 5 】

この例を実施するための 2 つの可能性を以下に説明する。

#### 【 0 2 9 6 】

アフィンマージ候補のマージインデックスは、他のマージ MV の値が何であれ、リスト内で常に同じ位置を有する。候補動き予測子の位置は、選択される可能性を示し、したがって、それがリストの上に配置される場合（より低いインデックス値）、その動きベクトル予測子が選択される可能性がより高い。

#### 【 0 2 9 7 】

第 1 の例では、アフィンマージ候補のマージインデックスが、マージ候補のリスト内で常に同じ位置を有する。これは、固定された「マージ  $i \ d \ x$ 」値を有することを意味する。例えば、アフィンマージモードは、最も確率の高いコンテンツではない複雑な動きを表すべきであるので、この値は 5 に等しく設定することができる。この実施形態の追加の利点は、現在のブロックが構文解析される（シンタックス要素の復号 / 読み出しだけでなく、データ自体を復号する）とき、現在のブロックをアフィンブロックとして設定できることである。その結果、この値を使用して、AMVP に使用されるアフィンフラグの CAB AC コンテキストを決定することができる。したがって、条件付き確率は、このアフィンフラグについて改善されるべきであり、符号化効率は、より良好であるべきである。 40

#### 【 0 2 9 8 】

第 2 の例では、アフィンマージ候補が他のマージ候補と共に導出される。この例では、

新しいアフィンマージ候補がマージ候補のリストに追加される（古典的マージモードまたは古典的マージスキップモードの場合）。図16は、この例を示す。図13と比較すると、アフィンマージ候補は、A1、B1、B0、A0、およびB2(1917)からの第1アフィン隣接ブロック(first affine neighbouring block)である。図12の1205と同じ条件が有効である場合(1927)、アフィンパラメータを用いて生成された動きベクトルフィールドが生成され、アフィンマージ候補が得られる(1929)。最初のマージ候補のリストは、ATMVP、時間的およびアフィンマージ候補の使用に従って、4、5、6、または7の候補を有することができる。

#### 【0299】

すべてのこれらの候補の間の順序は、より可能性の高い候補が、動きベクトル候補のカットを行う可能性がより高いことを保証するために、最初に処理されるべきであるため、重要であり、好ましい順序は、以下の通りである。

#### 【0300】

A1、B1、B0、A0、アフィンマージ、ATMVP、B2、時間的、組み合わせ、Zero\_MV

アフィンマージ候補は、ATMVP候補の前であるが、4つの主要な隣接ブロックの後に配置されることに留意することが重要である。ATMVP候補の前にアフィンマージ候補を設定することの利点は、ATMVPおよび時間的予測子候補の後にそれを設定することと比較して、符号化効率が増加することである。この符号化効率の向上は、GOP (group of pictures) 構造と、GOP内の各ピクチャのQP (Quantization Parameter) 設定と、に依存する。しかし、最も使用されるGOPおよびQP設定では、この順序は符号化効率の増加をもたらす。

#### 【0301】

この解決策のさらなる利点は、シンタックスおよび導出処理の両方のための、古典的なマージおよび古典的なマージスキップモード（すなわち、ATMVPまたはアフィンマージ候補などの追加の候補を有するマージモード）のクリーンな設計である。さらに、アフィンマージ候補のマージインデックスは、マージ候補のリスト内の以前の候補の利用可能性または値（重複チェック）に従って変更することができる。その結果、効率的な信号化を得ることができる。

#### 【0302】

さらなる例では、アフィンマージ候補のためのマージインデックスが、1つまたはいくつかの条件に従って可変である。

#### 【0303】

例えば、アフィンマージ候補に関連付けられたリスト内のマージインデックスまたは位置は、基準に従って変化する。原理は、アフィンマージ候補が選択される確率が高い場合には、アフィンマージ候補に対応するマージインデックスに対して低い値を設定することである（選択される確率が低い場合には、より高い値を設定する）。

#### 【0304】

第12の実施形態では、アフィンマージ候補がマージインデックス値を有する。マージインデックスの符号化効率を改善するために、マージインデックスのビットのコンテキスト変数を、隣接ブロックおよび/または現在のブロックのアフィンフラグに依存させることが有効である。

#### 【0305】

例えば、コンテキスト変数は、以下の式を使用して決定することができる。

#### 【0306】

$c_t \times I_d x = IsAffine(A1) + IsAffine(B1) + IsAffine(B0) + IsAffine(A0) + IsAffine(B2)$

結果として得られるコンテキスト値は、値0、1、2、3、4、または5を有することができる。

#### 【0307】

10

20

30

40

50

アフィンフラグは、符号化効率を高める。

#### 【0308】

第1の変形例では、より少ない隣接ブロックを含むために、 $c t \times I d x = I s A f f i n e (A1) + I s A f f i n e (B1)$  である。結果として得られるコンテキスト値は、値0、1、または2を有することができる。

#### 【0309】

また、第2の変形例では、より少ない隣接ブロックを含むために、 $c t \times I d x = I s A f f i n e (A2) + I s A f f i n e (B3)$  である。この場合も、結果として得られるコンテキスト値は、値0、1、または2を有することができる。

#### 【0310】

第3の変形例では、隣接ブロックを含まないようにするために、 $c t \times I d x = I s A f f i n e$  (現在のブロック) である。結果として得られるコンテキスト値は、値0または1を有することができる。

#### 【0311】

図15は、第3の変形例による符号化モードに関連するいくつかのシンタックス要素の部分復号処理のフローチャートである。この図では、スキップフラグ(1601)、予測モード(1611)、マージフラグ(1603)、マージインデックス(1608)、およびアフィンフラグ(1606)を復号することができる。このフローチャートは、先に説明した図12のフローチャートと同様であるため、詳細な説明は省略する。差は、マージインデックスのコンテキスト変数を取得するときに、マージインデックスの前に復号されるアフィンフラグを使用できるように、マージインデックス復号処理がアフィンフラグを考慮することである。これは、 $VTM2.0$  ではそうではない。 $VTM2.0$  では、現在のブロックのアフィンフラグが常に同じ値「0」を有するので、マージインデックスのためのコンテキスト変数を取得するために使用することができない。

#### 【0312】

##### 第13の実施形態

第10の実施形態では、現在のブロックのマージインデックスのビットのためのコンテキスト変数が、現在のブロック(現在の符号化ユニット、つまりCU)のためのスキップフラグの値に依存する。第13の実施形態では、スキップフラグ値を直接使用してマージインデックスのターゲットビットのコンテキスト変数を導出する代わりに、ターゲットビットのコンテキスト変数が、現在のCUのスキップフラグを符号化するために使用されるコンテキスト変数から導出される。これは、スキップフラグ自体がCABAC符号化され、したがってコンテキスト変数を有するために可能である。好ましくは、現在のCUのマージインデックスのターゲットビットに対するコンテキスト変数が、現在のCUのスキップフラグを符号化するために使用されるコンテキスト変数に等しく設定される(からコピーされる)。ターゲットビットは、第1番目のビットのみにすることができる。他のビットは、第1の実施形態のようにバイパス符号化されてもよい。

#### 【0313】

現在のCUのスキップフラグのコンテキスト変数は、 $VTM2.0$  に規定された方法で導出される。 $VTM2.0$  リファレンスソフトウェアと比較したこの実施形態の利点は、符号化効率に影響を与えることなく、マージインデックス復号およびデコーダ設計(およびエンコーダ設計)の複雑さが低減されることである。実際に、この実施形態では、現在の $VTM$ マージインデックス符号化(符号化/復号)のための5の代わりに、マージインデックスを符号化するために、最低でも1つのCABAC状態のみが必要とされる。さらに、他のビットはCABACバイパス符号化され、CABACで全てのビットを符号化するのと比較して演算の数を減らすので、最悪の場合の複雑さを減らす。

#### 【0314】

##### 第14の実施形態

第13の実施形態では、ターゲットビットのコンテキスト変数/値が、現在のCUのスキップフラグのコンテキスト変数から導出された。第14の実施形態では、ターゲットビ

10

20

30

40

50

ットのコンテキスト値が、現在の C U のアフィンフラグのコンテキスト変数から導出される。

【 0 3 1 5 】

これは、アフィンフラグ自体が C A B A C 符号化され、したがってコンテキスト変数を有するために可能である。好ましくは、現在の C U のマージインデックスのターゲットビットに対するコンテキスト変数が、現在の C U のアフィンフラグに対するコンテキスト変数に等しく設定される（からコピーされる）。ターゲットビットは、第 1 番目のビットのみにすることができる。他のビットは、第 1 の実施形態のようにバイパス符号化される。

【 0 3 1 6 】

現在の C U のアフィンフラグのコンテキスト変数は、V T M 2 . 0 に規定された方法で導出される。

10

【 0 3 1 7 】

V T M 2 . 0 リファレンスソフトウェアと比較したこの実施形態の利点は、符号化効率に影響を与えることなく、マージインデックス復号およびデコーダ設計（およびエンコーダ設計）の複雑さが低減されることである。実際に、この実施形態では、現在の V T M マージインデックス符号化（符号化 / 復号）のために 5 ではなく、最小で 1 つの C A B A C 状態のみがマージインデックスのために必要とされる。さらに、他のビットは C A B A C バイパス符号化され、C A B A C で全てのビットを符号化するのと比較して演算の数を減らすので、最悪の場合の複雑さを減らす。

【 0 3 1 8 】

20

第 1 5 の実施形態

前述の実施形態のいくつかでは、コンテキスト変数が 2 を超える値、例えば、3 つの値 0、1、および 2 を有していた。しかしながら、複雑さを低減し、処理される状態の数を低減するために、許可されるコンテキスト変数値の数を 2、例えば、0 および 1 に制限することが可能である。これは、例えば、値 2 を有する任意の初期コンテキスト変数を 1 に変更することによって達成することができる。実際には、この単純化が符号化効率に影響を及ぼさないか、または限定された影響しか及ぼさない。

【 0 3 1 9 】

実施形態および他の実施形態の組合せ

前述の実施形態のうちの任意の 2 つ以上が組み合わされてもよい。

30

【 0 3 2 0 】

前述の説明は、マージインデックスの符号化および復号に焦点を当ててきた。例えば、第 1 の実施形態は、A T M V P 候補（古典的マージモードまたは古典的マージスキップモード、すなわち非アフィンマージモードまたは非アフィンマージスキップモードの場合）を含むマージ候補のリストを生成することと、リスト内のマージ候補のうちの 1 つを選択することと、C A B A C 符号化を使用して、選択されたマージ候補のマージインデックスを生成することと、を含み、マージインデックスの 1 つまたは複数のビットは、バイパス C A B A C 符号化される。原則として、本発明は、動き情報予測子候補のリスト（例えば、アフィンマージ候補または動きベクトル予測子（M V P）候補のリスト）を生成することと、リスト内の動き情報予測子候補（例えば、M V P 候補）のうちの 1 つを選択することと、リスト内の選択された動き情報予測子候補（例えば、選択されたアフィンマージ候補または現在のブロックの動きベクトルを予測するための選択された M V P 候補）の識別子またはインデックスを生成することとを含むマージモード（例えば、アフィンマージモード）以外のモードに適用することができる。したがって、本発明はマージモード（すなわち、古典的マージモードおよび古典的マージスキップモード）に限定されず、符号化または復号されるインデックスは、マージインデックスに限定されない。例えば、V V C の開発において、前述の実施形態の技術は、H E V C の A M V P モード、または V V C におけるその同等のモード、またはアフィンマージモードなど、マージモード以外のモードに適用され得る（または拡張され得る）ことが考えられる。添付の特許請求の範囲は、相応に解釈されるべきである。

40

50

## 【0321】

前述したように、前述の実施形態では、アフィンマージモード（アフィンマージまたはアフィンマージスキップモード）および／または1つまたは複数のアフィンパラメータの1つまたは複数の動き情報候補（たとえば、動きベクトル）は、空間的に隣接するブロック（たとえば、位置A1、B1、B0、A0、B2）または時間的に関連するブロック（たとえば、コロケートされたブロックを有する「中央」ブロック、または「H」などのその空間的隣接）の間でアフィン符号化される第1隣接ブロックから取得される。これらの位置は、図6aおよび図6bに示されている。1つまたは複数の動き情報および／または現在のブロック（または現在のCUなどの、現在符号化／復号されているサンプル／ピクセル値のグループ）と隣接ブロック（現在のブロックに空間的に隣接するかまたは時間的に関連する）との間のアフィンパラメータのこの取得（例えば、導出するまたは共有するまたは「マージする」）を可能にするために、マージ候補（すなわち、古典的マージモード候補）のリストに、1つまたは複数のアフィンマージ候補が追加され、その結果、選択されたマージ候補（たとえば、HEVC内の「merge\_idx」などのシンタックス要素またはその機能的に同等のシンタックス要素を使用して、マージインデックスを使用してシグナリングされる）がアフィンマージ候補である場合、現在のCU／ブロックは、アフィンマージ候補とともにアフィンマージモードを使用して符号化／復号される。

10

## 【0322】

上述のように、アフィンマージモードの1つまたは複数の動き情報および／またはアフィンパラメータを取得する（例えば、導出するまたは共有する）ためのそのような1つまたは複数のアフィンマージ候補は、アフィンマージ候補の別個のリスト（またはセット）（古典的マージモードに使用されるマージ候補のリストと同じであっても異なっていてもよい）を使用してシグナリングすることもできる。

20

## 【0323】

本発明の一実施形態によれば、前述の実施形態の技術がアフィンマージモードに適用される場合、アフィンマージ候補のリストは図8に示され、それに関連して説明された古典的マージモードのための動きベクトル導出処理と同じ技術を使用して、または図13に示され、それに関連して説明されたマージ候補導出処理と同じ技術を使用して生成することができる。（アフィンマージモードまたはアフィンマージスキップモードのための）アフィンマージ候補のこのリストおよび（古典的マージモードまたは古典的マージスキップモードのための）マージ候補のリストを生成／コンパイルするために同じ技術を共有することの利点は、別個の技術を有する場合と比較して、符号化／復号処理の複雑さが低減されることである。

30

## 【0324】

同様の利点を達成するために、他の実施形態によれば、同様の技法が、（複数の候補から）選択された動き情報予測子をシグナリングすることを必要とする他のインター予測モードに適用されることを理解されたい。

## 【0325】

別の実施形態によれば、図24に関連して以下に示す別個の技術を使用して、アフィンマージ候補のリストを生成／コンパイルすることができる。

40

## 【0326】

図24は、アフィンマージモード（アフィンマージモードおよびアフィンマージスキップモード）のためのアフィンマージ候補導出処理を示すフローチャートである。導出処理の第1ステップでは、空間的アフィンマージ候補2413を取得／導出するため、5つのブロック位置が考慮される（2401～2405）。これらの位置は、参照番号A1、B1、B0、A0、およびB2で図6a（および図6b）に示される空間的位置である。次のステップでは、空間的動きベクトルの利用可能性がチェックされ、各位置A1、B1、B0、A0、およびB2に関連付けられたインターモード符号化ブロックの各々が（例えば、アフィンマージ、アフィンマージスキップ、またはアフィンAMVPモードのいずれか1つを使用して）アフィンモードで符号化されるかどうかが判定される（2410）。

50

。せいぜい 5 つの動きベクトル（すなわち、空間的アフィンマージ候補）が選択／取得／導出される。予測子が存在する（例えば、その位置に関連付けられた動きベクトルを取得／導出するための情報がある）場合、およびブロックがイントラ符号化されていない場合、およびブロックがアフィンである場合（すなわち、アフィンモードを使用して符号化されている）には、予測子が利用可能であるとみなされる。

#### 【0327】

次に、各利用可能なブロック位置について、アフィン動き情報が導出／取得される（2411）（2410）。この導出は、ブロック位置のアフィンモデル（および、例えば、図11（a）および図11（b）に関連して説明したそのアフィンモデルパラメータ）に基づいて、現在のブロックに対して実行される。次に、プルーニング処理（2412）を適用して、リストに以前に追加された互いに同じアフィン動き補償を与える（または同じアフィンモデルパラメータを有する）候補を除去する。

10

#### 【0328】

この段階の最後に、空間的アフィンマージ候補のリストは、5つまでの候補を含む。

#### 【0329】

候補の数（Nb\_Cand）が候補の最大数よりも厳密に少ない場合（2426）（ここではMax\_Cand はビットストリームスライスヘッダでシグナリングされる値で、アフィンマージモードの場合は 5 に等しくなるが、インプリメンテーションによって異なる／可変になることがある）。

20

#### 【0330】

次に、構築されたアフィンマージ候補（すなわち、例えば、HEVCにおける結合された双予測マージ候補と同様の役割を果たす、ターゲット番号に近づくだけでなく、何らかのダイバーシティを提供するために生成される追加のアフィンマージ候補）が生成される（2428）。これらの構築されたアフィンマージ候補は、現在のブロックの隣接する空間的および時間的位置に関連する動きベクトルに基づいている。まず、アフィンモデルを生成するための動き情報を生成するために、制御点が定義される（2418、2419、2420、2421）。これらの制御点のうち 2 つは、例えば図11（a）と図11（b）の v0 と v1 に対応する。これらの 4 つの制御点は、現在のブロックの四隅に対応している。

20

#### 【0331】

制御点の左上（2418）の動き情報は、それが存在し、このブロックがインターモード（2414）で符号化されている場合には、位置 B2（2405）におけるブロック位置の動き情報から（例えば、それと等しくすることによって）得られる。そわない場合、制御点左上（2418）の動き情報は、それが存在し、このブロックがインターモード（2414）で符号化されており、そのようなケースではない場合には、（図6b に示されるように）位置 B3（2406）におけるブロック位置の動き情報から（例えば、それと等しくすることによって）取得され、制御点左上（2418）の動き情報は、それが存在し、このブロックがインターモード（2414）で符号化されている場合には、（図6b に示されるように）位置 A2（2407）におけるブロック位置の動き情報から（例えば、等しい）取得される。この制御点に利用可能なブロックがない場合、それは利用不可能（利用不可能）であると考えられる。

30

#### 【0332】

制御点右上（2419）の動き情報は、それが存在し、このブロックがインターモード（2415）で符号化されている場合には、位置 B1（2402）にあるブロック位置の動き情報から得られる（例えば、それに等しい）。そわない場合、制御点右上（2419）の動き情報は、それが存在し、このブロックがインターモード（2415）で符号化されている場合には、位置 B0（2403）におけるブロック位置の動き情報から（例えば、等しい）取得される。この制御点に利用可能なブロックがない場合、それは利用不可能（利用不可能）であると考えられる。

40

#### 【0333】

50

制御点左下 (2420) の動き情報は、それが存在し、このブロックがインター モード (2416) で符号化されている場合には、位置 A1 (2401) にあるブロック位置の動き情報から得られる（例えば、等しい）。そわなければ、制御点左下 (2420) の動き情報は、それが存在し、このブロックがインター モード (2416) で符号化されている場合には、位置 A0 (2404) にあるブロック位置の動き情報から得られる（例えば、等しい）。この制御点に利用可能なブロックがない場合、それは利用不可能（利用不可能）であると考えられる。

#### 【0334】

制御点右下 (2421) の動き情報は、それが存在し、このブロックがインター モード (2417) で符号化されている場合には、時間的候補の動き情報、例えば、位置 H (2408) におけるコロケートされたブロック位置（図 6a に示すように）、から得られる（例えば、等しい）。この制御点に利用可能なブロックがない場合、それは利用不可能（利用不可能）であると考えられる。

10

#### 【0335】

これらの制御点に基づいて、最大 10 個の構築されたアフィンマージ候補を生成することができる（2428）。これらの候補は、4 つ、3 つ、または 2 つの制御点を有するアフィンモデルに基づいて生成される。例えば、第 1 の構築されたアフィンマージ候補は、4 つの制御点を使用して生成されてもよい。次に、以下の 4 つの構築されたアフィンマージ候補は、3 つの制御点の 4 つの異なるセット（すなわち、4 つの利用可能な制御点のうちの 3 つを含むセットの 4 つの異なる可能な組み合わせ）を使用して生成することができる 4 つの可能性である。次いで、他の構築されたアフィンマージ候補は、2 つの制御点の異なるセット（すなわち、4 つの制御点のうちの 2 つを含むセットの異なる可能な組み合わせ）を使用して生成されたものである。

20

#### 【0336】

候補の数 (Nb\_Cand) が、これらの追加の（構築された）アフィンマージ候補を追加した後に、候補の最大数 (Max\_Cand) よりも厳密に少ないままである場合 (2430)、アフィンマージ候補のリスト内の候補の数が目標数（例えば、候補の最大数）に達するまで、ゼロ動きベクトル候補（または、適用可能な場合には結合された双予測マージ候補）などの他の追加の仮想動き情報候補が追加 / 生成される（2432）。

30

#### 【0337】

この処理の最後に、アフィンマージモード候補のリストまたはセット（すなわち、アフィンマージモードおよびアフィンマージスキップモードであるアフィンマージモードの候補のリストまたはセット）が生成 / 構築される（2434）。図 24 に示すように、アフィンマージ（動きベクトル予測子）候補のリストまたはセットは、空間的候補のサブセット (2401 ~ 2407) および時間的候補 (2408) から構築 / 生成される（2434）。本発明の実施形態によれば、アフィンマージ候補のリスト / セットを生成するために、利用可能性、ブルーニング処理、または潜在的候補の数 / タイプをチェックするための異なる順序を有する他のアフィンマージ候補導出処理（例えば、ATMVP 候補も、図 13 または図 16 のマージ候補リスト導出処理と同様の方法で追加することができる）を使用することもできることを理解されたい。

40

#### 【0338】

以下の実施形態は、アフィンマージ候補のリスト（またはセット）をどのように使用して、選択されたアフィンマージ候補（マージモードに使用されるマージインデックス、または特にアフィンマージモードで使用される別個のアフィンマージインデックスを使用してシグナリングすることができる）をシグナリングする（例えば、符号化または復号する）ことができるかを示す。

#### 【0339】

以下の実施形態では、マージモード（すなわち、後に定義されるアフィンマージモード以外のマージモード、言い換えれば、古典的非アフィンマージモードまたは古典的非アフィンマージスキップモード）は、空間的に隣接するブロックまたは時間的に関連するブロ

50

ックのいずれかの動き情報が現在のブロックについて取得される（または現在のブロックについて導出される、または現在のブロックと共有される）場合のマージモードのタイプであり、マージモード予測子候補（すなわち、マージ候補）は、マージモードで現在のブロックが動き情報を取得／導出することができる、1つまたは複数の空間的に隣接するブロックまたは時間的に関連するブロックに関する情報であり、マージモード予測子は、選択されたマージモード予測子候補であり、その情報は、現在のブロックの動き情報を予測するとき、およびマージモード（例えば符号化または復号）処理にてシグナリングする間に使用され、マージモード予測子候補のリスト（またはセット）からマージモード予測子を識別するインデックス（例えばマージインデックス）がシグナリングされ、アフィンマージモードは、現在のブロックの動き情報および／またはアフィンモード処理（またはアフィン動きモデル処理）のためのアフィンパラメータが、この取得された／導出された／共有された動き情報が使用できるように、空間的に隣接しているブロックまたは時間的に関連するブロックのいずれかの動き情報が現在のブロックについて取得される（現在のブロックについて導出される、または現在のブロックと共有される）場合におけるマージモードのタイプであり、アフィンマージモード予測子候補（すなわちアフィンマージ候補）は、アフィンマージモードにおいて現在のブロックが動き情報を取得／導出できる、1つまたは複数の空間的に隣接しているブロックまたは時間的に関連するブロックに関する情報であり、アフィンマージモード予測子は、選択されたアフィンマージモード予測子候補であり、その情報は、現在のブロックの動き情報を予測するとき、およびアフィンマージモード（例えば符号化または復号）処理にてシグナリングする間に、アフィン動きモデルにおいて使用可能であり、アフィンマージモード予測子候補のリスト（またはセット）からアフィンマージモード予測子を識別するインデックス（例えばアフィンマージインデックス）がシグナリングされる。以下の実施形態では、アフィンマージモードが候補のリスト／セット（「アフィンマージリスト」または「サブブロックマージリスト」としても知られる）から1つのアフィンマージモード予測子候補を識別するためのそれ自体のアフィンマージインデックス（変数である識別子）を有するマージモードであり、それに関連する単一のインデックス値を有するのとは対照的に、アフィンマージインデックスは、その特定のアフィンマージモード予測子候補を識別するためにシグナリングされることが理解される。

#### 【0340】

以下の実施形態では、「マージモード」は、HEVC/JEM/VTMにおける古典的なマージスキップモードまたは古典的なマージモードまたは任意の機能的に同等のモードのいずれか1つを指し、上記のような動き情報の取得（例えば、導出または共有）およびマージインデックスのシグナリングが上記モードで使用されることを条件とする。「アフィンマージモード」は、アフィンマージモードまたはアフィンマージスキップモード（存在する場合、そのような取得／導出を使用する）のいずれか一方、あるいは同一の特徴が上記モードで使用されることを条件とする、任意の他の機能的に同等のモードも指すことを理解されたい。

#### 【0341】

##### 第16の実施形態

第16の実施形態では、アフィンマージ候補のリストからアフィンマージモード予測子（候補）を識別するための動き情報予測子インデックスが、CABAC符号化を使用してシグナリングされ、動き情報予測子インデックスの1つまたは複数のビットは、バイパスCABAC符号化される。

#### 【0342】

実施形態の第1の変形例によれば、エンコーダにおいて、アフィンマージモードのための動き情報予測子インデックスは、動き情報予測子候補のリストを生成し、リスト内の動き情報予測子候補のうちの1つをアフィンマージモード予測子として選択し、CABAC符号化を使用して、選択された動き情報予測子候補のための動き情報予測子インデックスを生成する、ことによって符号化され、動き情報予測子インデックスの1つまたは複数の

10

20

30

40

50

ビットは、バイパス C A B A C 符号化される。次に、この選択された動き情報予測子候補に対するインデックスを示すデータが、ビットストリームに含まれる。次に、デコーダは、このデータを含むビットストリームから、動き情報予測子候補のリストを生成し、C A B A C 復号を使用して動き情報予測子インデックスを復号し、動き情報予測子インデックスの 1 つまたは複数のビットはバイパス C A B A C 復号され、アフィンマージモードが使用されるときに、復号された動き情報予測子インデックスを使用して、リスト内の動き情報予測子候補のうちの 1 つをアフィンマージモード予測子として識別する、ことによって、アフィンマージモードのための動き情報予測子インデックスを復号する。

#### 【 0 3 4 3 】

第 1 の変形例のさらなる変形例によれば、マージモードが使用されるときに、デコーダが復号された動き情報予測子インデックス（例えば、マージインデックス）を使用して、リスト内の動き情報予測子候補のうちの 1 つをマージモード予測子として識別することができるよう、リスト内の動き情報予測子候補のうちの 1 つまたは複数も、マージモードが使用されるときに、マージモード予測子として選択可能である。このさらなる変形例では、アフィンマージインデックスが、アフィンマージモード予測子（候補）をシグナリングするために使用され、アフィンマージインデックスをシグナリングすることは、第 1 ~ 第 1 5 の実施形態のいずれか 1 つによるマージインデックスシグナリング、または現在の V T M または H E V C で使用されるマージインデックスシグナリングに類似するインデックスシグナリングを使用して実装される。

#### 【 0 3 4 4 】

この変形例では、マージモードが使用される場合、マージインデックスをシグナリングすることは、は第 1 ~ 第 1 5 の実施形態のいずれか 1 つによるマージインデックスシグナリング、または現在の V T M または H E V C で使用されるマージインデックスシグナリングを使用して実施することができる。この変形例では、アフィンマージインデックスをシグナリングすることと、マージインデックスをシグナリングすることと、は異なるインデックスシグナリングスキームを使用することができる。この変形例の利点は、アフィンマージモードとマージモードの両方に対して効率的なインデックス符号化 / シグナリングを使用することによって、より良好な符号化効率を達成することである。さらに、この変形例では、マージインデックス（H E V C の " M e r g e \_ i d x [ ] [ ] " またはその機能的に同等なものなど）と、アフィンマージインデックス（" A \_ M e r g e \_ i d x [ ] [ ] " など）に別個のシンタックス要素を使用できる。これにより、マージインデックスとアフィンマージインデックスを個別にシグナリング（符号化 / 復号）できるようになる。

#### 【 0 3 4 5 】

さらに別のさらなる変形例によれば、マージモードが使用され、リスト内の動き情報予測子候補のうちの 1 つもマージモード予測子として選択可能である場合、C A B A C 符号化は、両方のモードについて、すなわちアフィンマージモードが使用される場合、およびマージモードが使用される場合に、現在のブロックの動き情報予測子インデックス（たとえば、マージインデックスまたはアフィンマージインデックス）の少なくとも 1 つのビットについて同じコンテキスト変数を使用し、その結果、アフィンマージインデックスおよびマージインデックスの少なくとも 1 つのビットは、同じコンテキスト変数を共有する。次に、デコーダは、マージモードが使用されるとき、復号された動き情報予測子インデックスを使用して、リスト内の動き情報予測子候補のうちの 1 つをマージモード予測子として識別し、C A B A C 復号は、両方のモードについて、すなわち、アフィンマージモードが使用されるとき、およびマージモードが使用されるときに、現在のブロックの動き情報予測子インデックスの少なくとも 1 つのビットについて同じコンテキスト変数を使用する。

#### 【 0 3 4 6 】

実施形態の第 2 の変形例によれば、エンコーダにおいて、動き情報予測子インデックスは、動き情報予測子候補のリストを生成し、アフィンマージモードが使用されるとき、リスト内の動き情報予測子候補のうちの 1 つをアフィンマージモード予測子として選択し、マージモードが使用されるとき、リスト内の動き情報予測子候補のうちの 1 つをマージモ

10

20

30

40

50

ード予測子として選択し、C A B A C 符号化を使用して、選択された動き情報予測子候補のための動き情報予測子インデックスを生成する、ことによって符号化され、動き情報予測子インデックスの1つまたは複数のビットは、バイパスC A B A C 符号化される。次に、この選択された動き情報予測子候補に対するインデックスを示すデータが、ビットストリームに含まれる。次に、デコーダは、ビットストリームから、動き情報予測子インデックスを、動き情報予測子候補のリストを生成し、C A B A C 復号を使用して動き情報予測子インデックスを復号し、動き情報予測子インデックスの1つまたは複数のビットはバイパスC A B A C 復号され、アフィンマージモードが使用される場合、復号された動き情報予測子インデックスを使用して、リスト内の動き情報予測子候補のうちの1つをアフィンマージモード予測子として識別し、マージモードが使用される場合、復号された動き情報予測子インデックスを使用して、リスト内の動き情報予測子候補のうちの1つをマージモード予測子として識別する、ことによって復号する。

#### 【 0 3 4 7 】

第2の変形例のさらなる変形例によれば、アフィンマージインデックスシグナリングおよびマージインデックスシグナリングは、第1～第15の実施形態のいずれか1つによる同じインデックスシグナリング方式、または現在のV T MまたはH E V Cで使用されるマージインデックスシグナリングを使用する。このさらなる変形例の利点は、実施中の単純な設計であり、これはまた、より少ない複雑さにつながり得る。この変形例では、アフィンマージモードが使用されるとき、エンコーダのC A B A C 符号化は、現在のブロックの動き情報予測子インデックス（アフィンマージインデックス）の少なくとも1つのビットのためのコンテキスト変数を使用することを含み、コンテキスト変数は、マージモードが使用されるときには、動き情報予測子インデックス（マージインデックス）の少なくとも1つのビットのための別のコンテキスト変数から分離可能であり、アフィンマージモードの使用を示すデータは、アフィンマージモードおよびマージモードのためのコンテキスト変数が、C A B A C 復号処理のために区別される（明確に識別される）ことができるよう、ビットストリームに含まれる。次に、デコーダは、ビットストリームから、ビットストリームにおけるアフィンマージモードの使用を示すためのデータを取得し、アフィンマージモードが使用されるとき、C A B A C 復号は、このデータを使用して、アフィンマージインデックスおよびマージインデックスのためのコンテキスト変数間で区別する。さらに、デコーダでは、アフィンマージモードの使用を示すためのデータが、取得されたデータがアフィンマージモードの使用を示すときに、アフィンマージモード予測子候補のリスト（またはセット）を生成したり、取得されたデータがマージモードの使用を示すときには、マージモード予測子候補のリスト（またはセット）を生成したりするためにも使用できる。

#### 【 0 3 4 8 】

この変形例は、マージインデックスおよびアフィンマージインデックスの両方が同じインデックスシグナリングスキームを使用してシグナリングされることを可能にし、一方、マージインデックスおよびアフィンマージインデックスは依然として、（例えば、別個のコンテキスト変数を使用することによって）互いに独立して符号化／復号される。

#### 【 0 3 4 9 】

同じインデックスシグナリングスキームを使用する1つの方法は、アフィンマージインデックスとマージインデックスの両方に同じシンタックス要素を使用することであり、つまり、アフィンマージモードが使用されている場合とマージモードが使用されている場合、選択された動き情報予測子候補の動き情報予測子インデックスは、どちらの場合も同じシンタックス要素を使用して符号化される。次に、デコーダでは、アフィンマージモードまたはマージモードを使用して現在のブロックが符号化された（かつ復号されている）かどうかにかかわらず、ビットストリームから同じシンタックス要素を解析することによって、動き情報予測子インデックスが復号される。

#### 【 0 3 5 0 】

図22は、第16の実施形態のこの変形例による、符号化モード（すなわち、同じイン

10

20

30

40

50

デックスシグナリングスキーム) に関するいくつかのシンタックス要素の部分復号処理を示す。この図は、アフィンマージモード (2257: Yes) のためのアフィンマージインデックス (2255 - 「マージ  $i d \times$  アフィン」) と、同じインデックスシグナリングスキームを有するマージモード (2257: No) のためのマージインデックス (2258 - 「マージ  $i d \times$ 」) と、のシグナリングを示す。いくつかの変形例では、アフィンマージ候補リストが、現在の VTM のマージ候補リストのように ATMP 候補を含むことができる。アフィンマージインデックスの符号化は、図 10 (a)、図 10 (b)、または図 14 に示すように、マージモードのマージインデックスの符号化と同様である。いくつかの変形例では、アフィンマージ候補導出で ATMP マージ候補が定義されていない場合、アフィンマージ候補リスト内の候補の最大数がマージ候補リスト内の候補の最大数と一致する。ATMP が最大 5 つの他の候補 (つまり、合計 6 つの候補) でマージモードに対してイネーブルになっている場合、アフィンマージインデックスは図 10 (b) で説明されているように符号化される。したがって、アフィンマージインデックスの各ビットは、それ自体のコンテキストを有する。マージインデックスシグナリングのビットに使用されるすべてのコンテキスト変数は、アフィンマージインデックスシグナリングのビットに使用されるコンテキスト変数とは無関係である。

#### 【0351】

さらなる変形例によれば、マージインデックスおよびアフィンマージインデックスシグナリングによって共有されるこの同じインデックスシグナリングスキームは、第 1 の実施形態と同様に、第 1 ビンのみについて CABAC 符号化を使用する。すなわち、動き情報予測子インデックスの第 1 番目のビットを除く全てのビットは、バイパス CABAC 符号化される。第 16 の実施形態のこのさらなる変形例では、ATMP がマージ候補のリストまたはアフィンマージ候補のリストのうちの 1 つに候補として含まれる場合 (たとえば、ATMP が SPS レベルでイネーブルされる場合)、各インデックス (すなわち、マージインデックスまたはアフィンマージインデックス) の符号化は、インデックスの第 1 番目のビットのみが、図 14 に示されるように单一のコンテキスト変数を使用して CABAC によって符号化されるように修正される。この单一のコンテキストは、ATMP が SPS レベルでイネーブルされない場合に、現在の VTM リファレンスソフトウェアと同じ方法で設定される。他のビット (リストに 5 つの候補のみ存在する場合は、2 番目のビットから 5 番目のビットまたは 4 番目のビット) は、バイパス符号化される。マージ候補リストに ATMP が候補として含まれていない場合 (たとえば、SPS レベルで ATMP がディスエーブルされている場合)、利用可能な 5 つのマージ候補と 5 つのアフィンマージ候補がある。マージモードのためのマージインデックスの第 1 番目のビットのみが、第 1 の单一のコンテキスト変数を使用して CABAC によって符号化される。そして、アフィンマージモードのためのアフィンマージインデックスの第 1 番目のビットのみが、第 2 の单一のコンテキスト変数を使用して CABAC によって符号化される。これらの第 1 と第 2 のコンテキスト変数は、マージインデックスとアフィンマージインデックスの両方で ATMP が SPS レベルでイネーブルされていない場合、現在の VTM リファレンスソフトウェアと同じ方法で設定される。他のビット (2 番目から 4 番目のビットまで) は、バイパス復号される。

#### 【0352】

デコーダは、エンコーダと同じマージ候補のリストおよび同じアフィンマージ候補のリストを生成する。これは、たとえば、図 24 の方法を使用することによって達成される。マージモードおよびアフィンマージモードの両方に同じインデックスシグナリングスキームが使用されるが、アフィンフラグ (2256) を使用して、現在復号されているデータがマージインデックスまたはアフィンマージインデックスに対するものであるかどうかを判定し、その結果、第 1 および第 2 コンテキスト変数が CABAC 復号処理のために互いに分離可能 (または区別可能) である。すなわち、アフィンフラグ (2256) は、インデックス復号処理中に使用され (すなわち、ステップ 2257 で使用され)、「マージ  $i d \times 2258$ 」または「マージ  $i d \times$  アフィン 2255」を復号するかどうかを決定する。

10

20

30

40

50

ATMVPがマージ候補のリストに候補として含まれていない場合（たとえば、ATMVPがSPSレベルでディスエーブルされている場合）、候補のリストの両方に5つのマージ候補がある（マージモードとアフィンマージモードについて）。マージインデックスの第1番目のビットのみが、第1の單一コンテキスト変数を使用してCABACによって復号される。そして、アフィンマージインデックスの第1番目のビットのみが、第2の單一コンテキスト変数を使用してCABACによって復号される。他のすべてのビット（2番目から4番目のビットまで）は、バイパス復号される。現在のリファレンスソフトウェアとは対照的に、ATMVPがマージ候補のリストに候補として含まれる場合（たとえば、ATMVPがSPSレベルでイネーブルされる場合）、マージインデックスの復号において第1の單一コンテキスト変数を使用し、およびアフィンマージインデックスの復号において第2の單一コンテキスト変数を使用して、マージインデックスの第1番目のビットのみがCABACによって復号される。他のビット（2番目から5番目のビットまたは4番目のビット）は、バイパス復号される。復号されたインデックスは次に、対応する候補のリスト（すなわち、マージ候補またはアフィンマージ候補）からエンコーダによって選択された候補を識別するために使用される。

#### 【0353】

この変形例の利点は、マージインデックスおよびアフィンマージインデックスの両方にに対して同じインデックスシグナリングスキームを使用することにより、符号化効率に著しい影響を与えることなく、これら2つの異なるモードを実施するためのインデックス復号およびデコーダ設計（およびエンコーダ設計）の複雑さが低減されることである。実際、この変数では、マージインデックスのすべてのビットおよびアフィンマージインデックスのすべてのビットがCABAC符号化／復号される場合の9または10の代わりに、2つのCABAC状態（第1および第2の單一コンテキスト変数のそれぞれに1つ）のみがインデックスシグナリングに必要とされる。さらに、他のすべてのビット（第1番目のビットは別として）がCABACバイパス符号化されるため、最悪の場合の複雑さが低減され、CABACによるすべてのビットの符号化と比較して、CABAC符号化／復号処理中に必要とされる動作の数が低減される。

#### 【0354】

さらに別のさらなる変形例によれば、CABAC符号化または復号は、アフィンマージモードが使用されるときとマージモードが使用されるときの両方のために、現在のブロックの動き情報予測子インデックスの少なくとも1つのビットに対して同じコンテキスト変数を使用する。このさらなる変形例では、マージインデックスの第1番目のビットとアフィンマージインデックスの第1番目のビットに使用されるコンテキスト変数が、どのインデックスが符号化または復号されているかに依存せず、つまり、（前の変形例からの）第1および第2の單一コンテキスト変数は区別／分離されず、1つと同じ單一コンテキスト変数である。したがって、前の変形例とは対照的に、マージインデックスおよびアフィンマージインデックスは、CABAC処理中に1つのコンテキスト変数を共有する。図23に示されるように、インデックスシグナリングスキームは、マージインデックスとアフィンマージインデックスの両方について同じであり、すなわち、1つのタイプのインデックス「マージidx(2308)」のみが、両方のモードについて符号化または復号される。CABACデコーダに関する限り、マージインデックスおよびアフィンマージインデックスの両方に同じシンタックス要素が使用され、コンテキスト変数を考慮する際にそれらを区別する必要はない。したがって、図22のステップ（2257）のように、現在のブロックがアフィンマージモードで符号化される（復号される）かどうかを判定するためにアフィンフラグ（2306）を使用する必要はなく、1つのインデックス（「マージidx」）だけが復号を必要とするので、図23のステップ2306の後に分岐はない。アフィンフラグは、アフィンマージモードで、すなわちCABACデコーダがインデックスを復号した後の予測処理中に（「マージidx」）、動き情報予測を実行するために使用される。さらに、このインデックスの第1番目のビット（すなわち、マージインデックスおよびアフィンマージインデックス）のみが、1つの單一コンテキストを使用してCABAC

10

20

30

40

50

Cによって符号化され、他のビットは、第1の実施形態について説明したようにバイパス符号化される。したがって、このさらなる変形例では、マージインデックスとアフィンマージインデックスの第1番目のビットの1つのコンテキスト変数が、マージインデックスとアフィンマージインデックスシグナリングの両方によって共有される。候補のリストのサイズがマージインデックスとアフィンマージインデックスとで異なる場合、各ケースの関連するインデックスをシグナリングするための最大ビット数も異なる可能性があり、つまり、それらは互いに独立している。したがって、バイパス符号化ビットの数は、必要に応じて、アフィンフラグ(2306)の値に従って、例えば、ビットストリームからの関連するインデックスのためのデータの構文解析を可能にするように、適宜調整することができる。

10

#### 【0355】

この変形例の利点は、符号化効率に著しい影響を及ぼすことなく、マージインデックスおよびアフィンマージインデックス復号処理およびデコーダ設計(およびエンコーダ設計)の複雑さが低減されることである。実際、このさらなる変形例では、マージインデックスおよびアフィンマージインデックスの両方をシグナリングするときに、前の変形例または9または10のCABAC状態の代わりに、1つのCABAC状態のみが必要とされる。さらに、他のすべてのビット(第1番目のビットは別として)がCABACバイパス符号化されるため、最悪の場合の複雑さが低減され、CABACによるすべてのビットの符号化と比較して、CABAC符号化/復号処理中に必要とされる動作の数が低減される。

20

#### 【0356】

この実施形態の前述の変形例では、アフィンマージインデックスシグナリングおよびマージインデックスシグナリングが、コンテンツの数を削減する、および/または第1～第15の実施形態のいずれかで説明したように、1つまたは複数のコンテキストを共有することができる。この利点は、これらのインデックスを符号化または復号するのに必要なコンテキストの数が減少することによる複雑さの減少である。

#### 【0357】

この実施形態の前述の変形例では、動き情報予測子候補が、方向、リストのID、参照フレームインデックス、および動きベクトルのうちの1つまたは複数を取得する(または導出する)ための情報を備える。好ましくは、動き情報予測子候補が、動きベクトル予測子候補を得るために情報を含む。好ましい変形例では、動き情報予測子インデックス(例えば、アフィンマージインデックス)は、アフィンマージモード予測子候補をシグナリングするために使用され、アフィンマージインデックスシグナリングは、第1～第15の実施形態のいずれか1つによるマージインデックスシグナリング、または現在のVTMまたはHEVCで使用されるマージインデックスシグナリング(マージ候補としてアフィンマージモードの動き情報予測子候補を伴う)に類似するインデックスシグナリングを使用して実装される。

30

#### 【0358】

本実施形態の前述の変形例では、動き情報予測子候補の生成されたリストが、第1の実施形態のように、または他の前述の第2から第15の実施形態のいくつかの変形例のように、ATMVP候補を含む。ATMVP候補は、マージ候補リストおよびアフィンマージ候補リストの一方または両方に含まれてもよい。あるいは、動き情報予測子候補の生成されたリストが、ATMVP候補を含まない。

40

#### 【0359】

本実施形態の前述の変形例では、マージインデックスおよびアフィンマージインデックスの候補のリストに含めることができる候補の最大数は固定である。マージインデックスとアフィンマージインデックスの候補のリストに含めることができる候補の最大数は同じであってもよい。そして、動き情報予測子候補の生成されたリストに含まれ得る動き情報予測子候補の最大数(またはターゲット数)を決定する(または示す)ためのデータが、エンコーダによってビットストリームに含まれ、デコーダはビットストリームから、動き情報予測子候補の生成されたリストに含まれ得る動き情報予測子候補の最大数(またはタ

50

ーゲット数)を決定するためのデータを取得する。これにより、マージインデックスまたはアフィンマージインデックスを復号するためのデータをビットストリームから解析できる。最大数(またはターゲット数)を決定する(または示す)ためのこのデータは、復号されたときの最大数(またはターゲット数)そのものであってもよく、またはデコーダが他のパラメータ/シンタックス要素、例えば、HEVCで使用される「five\_minus\_max\_num\_merge\_cand」または「MaxNumMergeCandidate-1」またはその機能的に同等のパラメータと関連して、この最大/ターゲット数を決定することを可能にしてもよい。

#### 【0360】

あるいは、マージインデックスおよびアフィンマージインデックスの候補のリスト内の候補の最大数(またはターゲット数)が変化しうる若しくは異なりうる場合(ATMVP候補または他の任意の候補の使用が一方のリストに対してイネーブルまたはディスエーブルされているが、他方のリストに対してはされていないから、または、リストが異なる候補リスト生成/導出処理を使用しているから、など)、アフィンマージモードが使用される場合、およびマージモードが使用されている場合に、動き情報予測子候補の生成されたリストに含まれ得る動き情報予測子候補の最大数(またはターゲット数)が、別々に決定可能であり、エンコーダがビットストリームに最大数/ターゲット数を決定するためのデータを含める。次に、デコーダはビットストリームから、最大/ターゲット数を決定するためのデータを取得し、取得したデータを使用して、動き情報予測子インデックスを解析または復号する。次に、アフィンフラグを使用して、例えば、マージインデックスの解析または復号と、アフィンマージインデックスと、を切り替えることができる。

10

#### 【0361】

前述のように、追加のインター予測モード(MHIIMージモード、トライアングルマージモード、およびMMVDマージモードなど)のうちの1つまたは複数を、マージモードまたはアフィンマージモードに加えて、またはその代わりに使用することができ、追加のインター予測モードのうちの1つまたは複数のためのインデックス(またはフラグまたは情報)をシグナリング(符号化または復号)することができる。以下の実施形態は、追加のインター予測モードのための情報(インデックスなど)のシグナリングに関する。

20

#### 【0362】

##### 第17の実施形態

30

すべてのインター予測モード(マージモード、アフィンマージモード、MHIIMージモード、トライアングルマージモード、MMVDマージモードを含む)のシグナリング

これらの複数のインター予測「マージ」モードは、第17の実施形態に従って、それらの関連するシンタックス(要素)と共にビットストリームで提供されるデータを使用してシグナリングされる。図26は、本発明の一実施形態による、現在のCU(画像部分またはブロック)のためのインター予測モードのための復号処理を示す。図12(および1201のスキップフラグ)に関連して説明されているように、第1番目のCUスキップフラグがビットストリーム(2601)から抽出される。CUがスキップ(2602)でない場合、すなわち、現在のCUがスキップモードで処理されない場合、predモードフラグ(2603)および/またはマージフラグ(2606)が復号され、現在のCUがマージCUであるかどうかが判断される。現在のCUがマージスキップ(2602)またはマージCU(2607)で処理される場合、MMVD\_Skip\_FlagまたはMMVD\_Merge\_Flagが復号される(2608)。このフラグが1に等しい場合(2609)、現在のCUはMMVDマージモードを使用して(すなわち、MMVDマージモードで、またはMMVDマージモードで)復号され、その結果、MMVDマージインデックスが復号され(2610)、続いてMMVD距離インデックス(2611)およびMMVD方向インデックス(2612)が復号される。CUがMMVDマージCU(2609)でない場合、マージサブロックフラグが復号される(2613)。このフラグは、前の説明では「アフィンフラグ」としても示されている。現在のCUがアフィンマージモード(「サブロックマージ」モードとも知られている)(2614)で処理される場合、マ

40

50

ージサブブロックインデックス（すなわち、アフィンマージインデックス）が復号される（2615）。現在のCUがアフィンマージモード（2614）で処理されず、スキップモード（2616）でも処理されない場合、MHIIMergeFlagが復号される（2620）。このブロックがMHIIMergeモード（2621）で処理される場合、正規のマージインデックス（2619）が、MHIIMergeモードのための関連するINTL予測モード（2622）で復号される。MHIIMergeモードは非スキップ「マージ」モードのためだけに利用可能であり、スキップモードのためには利用可能でないことに留意されたい。MHIIMergeFlagが0（2621）に等しい場合、またはスキップモード（2616）&アフィンマージモード（2614）で現在のCUを処理しない場合、トライアングルマージフラグが復号される（2617）。このCUがトライアングルマージモードで処理される場合（2618）、トライアングルマージインデックスが復号される（2623）。現在のCUがトライアングルマージモードで処理されない場合（2618）、現在のCUは正規のマージモードCUであり、マージインデックスが復号される。

### 【0363】

#### 各マージ候補のシグナリング

##### MMVDマージフラグ / インデックスシグナリング

第17の実施形態の第1の変形例では、MMVDマージモードでの使用 / 選択のために2つの初期候補のみが利用可能である。しかしながら、距離インデックスのための8つの可能な値と方向インデックスのための4つの可能な値もビットストリームと共にシグナリングされる場合、デコーダにおけるMMVDマージモードで使用するための潜在的な候補の数は64（2つの候補 × 8の距離インデックス × 4の方向インデックス）であり、各潜在的な候補は初期候補が異なる場合には別の（すなわち、一意的である）とは異なる。これらの64の潜在的な候補はエンコーダ側でMMVDマージモードのために評価 / 比較されることができ、次いで、選択された初期候補のためのMMVDマージインデックス（2610）は、単項最大コードでシグナリングされる。初期候補は2つのみ使用されるため、このMMVDマージインデックス（2610）はフラグに対応する。図27（a）は、1つのコンテキスト変数を使用してCABAC符号化されるこのフラグの符号化を示す。別の変形例では、異なる数の初期候補、距離インデックス値、および / または方向インデックス値を代わりに使用することができ、MMVDマージインデックスのシグナリングがそれに応じて適応される（例えば、少なくとも1つのビットが1つのコンテキスト変数を使用してCABAC符号化される）ことを理解されたい。

### 【0364】

#### トライアングルマージインデックスシグナリング

第17の実施形態の第1の変形例では、トライアングルマージインデックスが他のインター予測モードのためのインデックスシグナリングと比較した場合、異なるようにシグナリングされる。トライアングルマージモードでは、5つの初期候補と2つの可能なタイプの三角形との組合せに対応する候補の40の可能な置換が利用可能である（図25（a）および図25（b）を参照、2つの可能な第1のブロック予測子（2501または2511）および第2のブロック予測子（2502または2512）は各タイプの三角形についてのものである）。図27（b）は、トライアングルマージモード、すなわちこれらの候補のシグナリングのためのインデックスの符号化を示す。第1番目のビット（すなわち、第1番目のビン）は、1つのコンテキストでCABAC復号される。この第1番目のビットが0に等しい場合、第2番目のビット（すなわち、第2番目のビン）がCABACバイパス復号される。この第2番目のビットが0に等しい場合、インデックスはリスト内の第1番目の候補、つまりインデックス0（Cand0）に対応する。それ以外の場合（第2番目のビットが1に等しい場合）、インデックスはリスト内の第2番目の候補、すなわちインデックス1（Cand1）に対応する。第1番目のビットが1に等しい場合、Expontential-Golombコードがビットストリームから抽出され、Exponentia l-Golombコードがリスト内の選択された候補のインデックスを表す。つまり、インデックス2（Cand2）からインデックス39（Cand39）までから選

10

20

30

40

50

択される。

【0365】

別の変形例では、異なる数の初期候補が代わりに使用されてもよく、トライアングルマージインデックスのシグナリングがそれに応じて適応される（例えば、少なくとも1つのビットが1つのコンテキスト変数を使用してCABAC符号化される）ことが理解される。

【0366】

アフィンマージリストのATMVP

第17の実施形態の第2の変形例では、ATMVPがアフィンマージ候補リスト（すなわち、アフィンマージモードでは、-「サブプロックマージ」モードとしても知られる）の候補として利用可能である。図28は、この追加のATMVP候補（2848）を有するアフィンマージリスト導出のリストを示す。この図は図24（先に説明した）と同様であるが、この追加のATMVP候補（2848）がリストに追加されているので、詳細な説明の繰り返しはここでは省略する。別の変形例では、異なる数の初期候補が代わりに使用されてもよく、トライアングルマージインデックスのシグナリングがそれに応じて適応される（例えば、少なくとも1ビットが1つのコンテキスト変数を使用してCABAC符号化される）ことが理解される。

10

【0367】

別の変形例ではATMVP候補が別のインター予測モードの候補のリストに追加されてもよく、そのインデックスのシグナリングはそれに応じて適応される（例えば、少なくとも1つのビットは1つのコンテキスト変数を使用してCABAC符号化される）ことが理解される。

20

【0368】

図26は別の変形例によれば、すべてのインター予測モード（すなわち、マージモード、アフィンマージモード、MHIIMージモード、トライアングルマージモード、およびMMVDマージモード）のシグナリングの完全な概要を提供するが、インター予測モードのサブセットのみが代わりに使用されてもよい、ことも理解される。

【0369】

第18の実施形態

第18の実施形態によれば、トライアングルマージモードまたはMMVDマージモードの一方または両方は符号化または復号処理で使用するために利用可能であり、これらのインター予測モードの一方または両方はそのインデックス/フラグをシグナリングするときに、別のインター予測モードとコンテキスト変数（CABAC符号化とともに使用される）を共有する。

30

【0370】

この実施形態または以下の実施形態のさらなる変形例では、インター予測モードの1つまたは複数がそのインデックス/フラグをシグナリングするときに2つ以上のコンテキスト変数を使用することができることが理解される（例えば、アフィンマージモードはATMVP候補もそのアフィンマージインデックス符号化/復号処理のためにリストに含めることができるかどうかに応じて、4つまたは5つのコンテキスト変数を使用することができる）。

40

【0371】

例えば、この実施形態または以下の実施形態の変形例が実施される前に、すべてのインター予測モードのためのインデックス/フラグのすべてのビットをシグナリングするためのコンテキスト変数の総数は7：（正規）マージ=1（図10（a）に示すように）；アフィンマージ=4（図10（b）に示すように、ただし、1つのより少ない候補、例えば、ATMVP候補なし）；トライアングル=MMVD=1；およびMHI（使用可能な場合）=0（正規のマージと共有）であっても良い。次に、変形例を実装することにより、全てのインター予測モードのためのインデックス/フラグの全てのビットをシグナリングするためのコンテキスト変数の総数は5：（正規）マージ=1（図10（a）に示されるように）；アフィンマージ=4（図10（b）に示されるが、1つより少ない候補、例えばA

50

T M V P 候補なし) ; およびトライアングル = M M V D = M H I I (使用可能な場合) = 0 (正規マージと共有) に低減され得る。

【 0 3 7 2 】

別の例では、この変形例が実装される前に、すべてのインター予測モードのためのインデックス / フラグのすべてのビットをシグナリングするためのコンテキスト変数の総数は 4 : (正規) マージ = アフィンマージ = トライアングル = M M V D = 1 (図 10 (a) に示すように) ; および M H I I (使用可能な場合) = 0 (正規マージと共有) とすることができます。次に、この変形例を実施することによって、すべてのインター予測モードのためのインデックス / フラグのすべてのビットをシグナリングするためのコンテキスト変数の総数は 2 : (正規) マージ = アフィンマージ = 1 (図 10 (a) に示すように) ; およびトライアングル = M M V D = M H I I (使用可能な場合) = 0 (正規マージと共有) に低減される。

10

【 0 3 7 3 】

以下の説明を簡単にするために、1つのコンテキスト変数 (例えば、第1番目のビットのみ) を共有することまたは共有しないことを説明することに留意されたい。これは以下の説明では、コンテキスト変数を使用して各インター予測モードについて第1番目のビットのみをシグナリングする単純な場合をしばしば見ることを意味し、このコンテキスト変数は 1 (別個の / 独立のコンテキスト変数が使用される) または 0 (このビットがバイパス C A B A C 符号化されるか、または別のインター予測モードと同じコンテキスト変数を共有するので、別個の / 独立のコンテキスト変数は存在しない) のいずれかである。この実施形態および以下の実施形態の異なる変形例はそれに限定されず、他のビットのコンテキスト変数、または実際にはすべてのビットは同じ方法で共有 / 非共有 / バイパス C A B A C 符号化されてもよいことが理解される。

20

【 0 3 7 4 】

第 18 の実施形態の第 1 の変形例では、符号化または復号処理で使用するために利用可能なすべてのインター予測モードが少なくともいくつかの C A B A C コンテキストを共有する。

【 0 3 7 5 】

この変形例では、インデックス符号化およびインター予測モードのためのその関連パラメータ (例えば、(初期) 候補の数) は可能な限り / 互換性がある限り、同じまたは類似するように設定され得る。例えば、シグナリングを簡単にするために、アフィンマージモードおよびマージモードの候補の数をそれぞれ 5 および 6 に設定し、M M V D マージモードの初期候補の数を 2 に設定し、トライアングルマージモードの最大候補数は 40 である。また、トライアングルマージインデックスは、他のインター予測モードのように単項最大コードを使用してシグナリングされない。このトライアングルマージモードの場合、第1番目のビットのみのコンテキスト変数 (トライアングルマージインデックスの場合) を他のインター予測モードと共有することができる。この変形例の利点は、エンコーダとデコーダの設計が簡単になることである。

30

【 0 3 7 6 】

さらなる変形例では、すべてのマージインター予測モードのインデックスのための C A B A C コンテンツが共有される。これは、全てのインデックスの第1番目のビットに対して 1 つの C A B A C コンテキスト変数のみが必要とされることを意味する。さらに別の変形例では、インデックスが C A B A C 符号化されるべき 2 つ以上のビットを含む場合、追加ビット (第1番目のビットから離れたすべての C A B A C 符号化ビット) の符号化は別個の部分として (すなわち、C A B A C 符号化処理に関する限り、別のシンタックス要素のためであるかのように) 扱われ、2 つ以上のインデックスが C A B A C 符号化されるべき 2 つ以上のビットを有する場合、1 つの同じコンテキスト変数がこれらの C A B A C 符号化された「追加」ビットに対して共有される。この変形例の利点は、C A B A C コンテキストの量が減少することである。これは、変形例を実施するビデオコーデックによって処理されるシーケンスの大部分に対する符号化効率に著しく影響を与えることなく、エン

40

50

コーダおよびデコーダ側で記憶される必要があるコンテキスト状態に対する記憶要件を低減する。

#### 【 0 3 7 7 】

図 29 は、インター予測モードのためのさらなる変形例の復号処理を示す。この図は図 26 と同様であるが、この変形例の実装を含む。この図では現在の C U が MMVD マージモードで処理されると、その MMVD マージインデックスは正規のマージモードのマージインデックスと同じインデックス（つまり “マージインデックス” (2919)）として復号される。一方、MMVD マージモードでは正規のマージモードの場合とは異なり、選択できるのは 2 つの初期候補のみであり、6 ではない。可能性は 2 つしかないため、MMVD マージモードで使用されるこの「共有」インデックスは基本的にフラグである。同じインデックスが共有されるので、CABA C コンテキスト変数は、MMVD マージモードにおけるこのフラグと、マージモードにおけるマージインデックスの第 1 番目のビットとに対して同じである。次に、現在の C U が MMVD マージモードで処理されるべきであると判定された場合 (2925)、距離インデックス (2911) および方向インデックス (2912) が復号される。現在の C U がアフィンマージモード (2914) で処理されると判断された場合、そのアフィンマージインデックスは正規のマージモードのマージインデックスと同じインデックス（つまり “マージインデックス” (2919)）として復号される。一方、アフィンマージモードでは、正規のマージモードの場合とは異なり、最大候補数（つまり最大インデックス数）は 5 であり、6 ではない。現在の C U がトライアングルマージモード (2918) で処理されるべきであると判定された場合、第 1 番目のビットが共有インデックスとして復号され (2919)、その結果、同じ CABA C コンテキスト変数が正規のマージモードと共有される。この C U がトライアングルマージモード (2926) で処理されると、トライアングルマージインデックスに関連する残りのビットが復号される (2923)。

10

20

30

#### 【 0 3 7 8 】

したがって、例えば、CABA C 符号化処理中に、これらのインデックス / フラグを処理するとき、各インター予測モードのためのインデックス / フラグの第 1 番目のビットに使用される別個の（独立した）コンテキスト変数の数は：

（正規）マージ = 1 ;

MHI I = アフィンマージ = トライアングル = MMVD = 0 （正規マージと共有）

である。

30

#### 【 0 3 7 9 】

第 2 の変形例では、トライアングルマージモードまたは MMVD マージモードの一方または両方が使用される（すなわち、現在の C U の動き情報予測子選択に関する情報が関連するインター予測モードで処理 / 符号化 / 復号される）とき、その / それらのインデックスシグナリングはマージモードのインデックスシグナリングとコンテキスト変数を共有する。この変形例では、トライアングルマージインデックスの CABA C コンテキストおよび / または MMVD マージインデックス / フラグの CABA C コンテキストが（（正規の）マージモードの）マージインデックスの同じ CABA C コンテキストを共有する。これは、少なくともこれらの 3 つのモードに必要な CABA C 状態は 1 つだけであることを意味する。

40

#### 【 0 3 8 0 】

第 2 の変形例のさらなる変形例では、トライアングルマージインデックスの CABA C コンテキストおよび / または MMVD マージインデックス / フラグの CABA C コンテキストがマージインデックスの同じ第 1 番目の CABA C コンテキスト変数、例えば、マージインデックスの第 1 番目のビットの同じコンテキスト変数を共有する。

#### 【 0 3 8 1 】

したがって、例えば、CABA C 符号化処理中に、これらのインデックス / フラグを処理するとき、インデックス / フラグの第 1 番目のビットに使用される別個の（独立した）コンテキスト変数の数は：

50

(正規) マージ = 1 ;  
 M H I I (使用可能な場合) = アフィンマージ (使用可能な場合) = 0 (正規マージと共有) または実装によっては 1 ;  
 トライアングル = M M V D = 0 (正規マージと共有)  
 である。

#### 【0382】

第 2 の変形例のさらに別の変形例では、2つ以上のコンテキスト変数がトライアングルマージインデックス C A B A C 符号化 / 復号に使用されるか、または2つ以上のコンテキスト変数が M M V D マージインデックス C A B A C 符号化 / 復号に使用される場合、それらはすべて、マージインデックス C A B A C 符号化 / 復号に使用される2つ以上の C A B A C コンテキスト変数と共有されるか、または互換性がある場合はいつでも少なくとも部分的に共有され得る。

10

#### 【0383】

この第 2 の変形例の利点は、記憶される必要があるコンテキストの量が減少し、その結果、エンコーダおよびデコーダ側で記憶される必要がある状態の量が、それらを実施するビデオコーデックによって処理されるシーケンスの大部分の符号化効率に著しく影響を及ぼすことなく減少することである。

#### 【0384】

第 3 の変形例では、トライアングルマージモードまたは M M V D マージモードの一方または両方が使用される（すなわち、現在の C U の動き情報予測子選択に関する情報が関連するインター予測モードで処理 / 符号化 / 復号される）とき、その / それらのインデックスシグナリングは、アフィンマージモードのインデックスシグナリングとコンテキスト変数を共有する。この変形例では、トライアングルマージインデックスの C A B A C コンテキストおよび / または M M V D マージインデックス / フラグの C A B A C コンテキストが（アフィンマージモードのための）アフィンマージインデックスの同じ C A B A C コンテキストを共有する。

20

#### 【0385】

第 3 の変形例のさらなる変形例では、トライアングルマージインデックスの C A B A C コンテキストおよび / または M M V D マージインデックス / フラグの C A B A C コンテキストが、アフィンマージインデックスの同じ第 1 番目の C A B A C コンテキスト変数、例えば、アフィンマージインデックスの第 1 番目のビットの同じコンテキスト変数を共有する。

30

#### 【0386】

したがって、例えば、C A B A C 符号化処理中に、これらのインデックス / フラグを処理するとき、インデックス / フラグの第 1 番目のビットに使用される別個の（独立した）コンテキスト変数の数は：

(正規) マージ (使用可能な場合) = 0 (アフィンマージと共有) または実装に応じて 1 ;

M H I I (使用可能な場合) = 0 (正規のマージと共有) ;

アフィンマージ = 1 ;

トライアングル = M M V D = 0 (アフィンマージと共有)

40

である。

#### 【0387】

第 3 の変形例のさらに別の変形例では、2つ以上のコンテキスト変数がトライアングルマージインデックス C A B A C 符号化 / 復号に使用されるか、または2つ以上のコンテキスト変数が M M V D マージインデックス C A B A C 符号化 / 復号に使用される場合、それらはすべて共有されるか、またはアフィンマージインデックス C A B A C 符号化 / 復号に使用される2つ以上の C A B A C コンテキスト変数と互換性がある場合には少なくとも部分的に共有されることができる。

#### 【0388】

50

第4の変形例では、MMVDマージモードが使用される（すなわち、現在のCUの動き情報予測子選択に関する情報がMMVDマージモードで処理／符号化／復号される）場合、そのインデックスシグナリングはマージモードまたはアフィンマージモードのインデックスシグナリングとコンテキスト変数を共有する。この変形例では、MMVDマージインデックス／フラグのCABACコンテキストが、マージインデックスの同じCABACコンテキスト、またはアフィンマージインデックスの同じCABACコンテキストである。

#### 【0389】

したがって、例えば、CABAC符号化処理中に、これらのインデックス／フラグを処理するとき、インデックス／フラグの第1番目のビットに使用される別個の（独立した）コンテキスト変数の数は：

（正規）マージ = 1 ;  
 MHI (使用可能な場合) = 0 (正規のマージと共有) ;  
 アフィンマージ (使用可能な場合) = 0 (正規のマージと共有) または実装によっては  
 1 ;  
 MMVD = 0 (正規のマージと共有)  
 または  
 （正規）マージ (使用可能な場合) = 0 (アフィンマージと共有) または実装に応じて  
 1 ;  
 MHI (使用可能な場合) = 0 (正規のマージと共有) ;  
 アフィンマージ = 1 ;  
 MMVD = 0 (アフィンマージと共有)  
 である。

#### 【0390】

第5の変形例では、トライアングルマージモードが使用される（すなわち、現在のCUの動き情報予測子選択に関する情報がトライアングルマージモードで処理／符号化／復号される）場合、そのインデックスシグナリングはマージモードまたはアフィンマージモードのためのインデックスシグナリングとコンテキスト変数を共有する。この変形例では、トライアングルマージインデックスのCABACコンテキストが、マージインデックスの同じCABACコンテキストまたはアフィンマージインデックスの同じCABACコンテキストである。

#### 【0391】

したがって、例えば、CABAC符号化処理中に、これらのインデックス／フラグを処理するとき、インデックス／フラグの第1番目のビットに使用される別個の（独立した）コンテキスト変数の数は：

（正規）マージ = 1 ;  
 MHI (使用可能な場合) = 0 (正規のマージと共有) ;  
 アフィンマージ (使用可能な場合) = 0 (正規のマージと共有) または実装によっては  
 1 ;  
 トライアングル = 0 (正規のマージと共有)  
 または  
 （正規）マージ (使用可能な場合) = 0 (アフィンマージと共有) または実装に応じて  
 1 ;  
 MHI (使用可能な場合) = 0 (正規のマージと共有) ;  
 アフィンマージ = 1 ;  
 トライアングル = 0 (アフィンマージと共有)  
 である。

#### 【0392】

第6の変形例では、トライアングルマージモードが使用される（すなわち、現在のCUの動き情報予測子選択に関する情報がトライアングルマージモードで処理／符号化／復号される）場合、そのインデックスシグナリングはMMVDマージモードのインデックスシ

10

20

30

40

50

グナリングとコンテキスト変数を共有する。この変形例では、トライアングルマージインデックスの C A B A C コンテキストが M M V D マージインデックスの同じ C A B A C コンテキストである。したがって、例えば、C A B A C 符号化処理中に、これらのインデックス / フラグを処理するとき、インデックス / フラグの第 1 番目のビットに使用される別個の（独立した）コンテキスト変数の数は：

M M V D = 1 ;

トライアングル = 0 ( M M V D と共有) ;

（正規）マージまたは M H I I またはアフィンマージ = 実装および使用可能かどうかに応じて

である。

#### 【 0 3 9 3 】

第 7 の変形例では、トライアングルマージモードまたは M M V D マージモードの一方または両方が使用される（すなわち、現在の C U の動き情報予測子選択に関する情報が関連するインター予測モードで処理 / 符号化 / 復号される）場合、その / それらのインデックスシグナリングは、候補のリストに A T M V P 予測子候補を含むことができるインター予測モードのためのインデックスシグナリングとコンテキスト変数を共有し、すなわち、前記インター予測モードは、利用可能な候補のうちの 1 つとして A T M V P 予測子候補を有することができる。この変形例では、トライアングルマージインデックスの C A B A C コンテキストおよび / または M M V D マージインデックス / フラグが A T M V P 予測子を使用することができるインター予測モードのインデックスの同じ C A B A C コンテキストを共有する。

#### 【 0 3 9 4 】

さらなる変形例では、トライアングルマージインデックスおよび / または M M V D マージインデックス / フラグのための C A B A C コンテキスト変数が、マージモードのマージインデックスのための同じ第 1 番目の C A B A C コンテキスト変数を、包含可能な A T M V P 候補と共有し、またはアフィンマージモードのアフィンマージインデックスを包含可能な A T M V P 候補と共有する。

#### 【 0 3 9 5 】

さらに別の変形例では、2 つ以上のコンテキスト変数がトライアングルマージインデックス C A B A C 符号化 / 復号のために使用されるか、または 2 つ以上のコンテキスト変数が M M V D マージインデックス C A B A C 符号化 / 復号のために使用される場合、それらはすべて、包含可能な A T M V P 候補を有するアフィンマージモードのアフィンマージインデックスまたはマージモードのマージインデックスのために使用される 2 つ以上の C A B A C コンテキスト変数と共用されるか、または少なくとも部分的に互換性があるどこでも共用ができる。

#### 【 0 3 9 6 】

A T M V P ( 予測因子 ) 候補は、他のタイプの予測因子と比較した場合、C A B A C 適応から最も利益を得る予測因子であるため、これらの変形例の利点は符号化効率の改善である。

#### 【 0 3 9 7 】

##### 第 1 9 の実施形態

第 1 9 の実施形態によれば、トライアングルマージモードまたは M M V D マージモードの一方または両方が符号化または復号処理で使用するために利用可能であり、これらのインター予測モードの一方または両方のためのインデックス / フラグは、インデックス / フラグをシグナリングするときに C A B A C バイパス符号化される。

#### 【 0 3 9 8 】

第 1 9 の実施形態の第 1 の変形例では、符号化または復号処理で使用するために利用可能なすべてのインター予測モードが、そのインデックス / フラグを C A B A C バイパス符号化 / 復号して、インデックス / フラグをシグナリングする。この変形例では、すべてのインター予測モードのすべてのインデックスが C A B A C コンテキスト変数を使用せずに

10

20

30

40

50

(例えば、図 17 のバイパス符号化エンジン 1705 によって) 符号化される。これは、図 26 のマージインデックス (2619)、アフィンマージインデックス (2615)、MMVDマージインデックス (2610)、トライアングルマージインデックス (2623) のすべてのビットが CABA C バイパス符号化されていることを意味する。図 30 (a) ~ 30 (c) は、この実施形態によるインデックス / フラグの符号化を示す。図 30 (a) は、初期 MMVD マージ候補の MMVD マージインデックス符号化を示している。図 30 (b) は、トライアングルマージインデックス符号化を示す。図 30 (c) は、マージインデックス符号化にも簡単に使用できるアフィンマージインデックス符号化を示している。

#### 【0399】

10

したがって、例えば、CABA C 符号化処理中に、これらのインデックス / フラグを処理するとき、インデックス / フラグに使用される別個の（独立した）コンテキスト変数の数は：

（正規）マージ（使用可能な場合） = MHI I（使用可能な場合） = アフィンマージ（使用可能な場合） = トライアングル（使用可能な場合） = MMVD（使用可能な場合） = 0（すべてバイパスコード）である。この変形例の利点は、記憶される必要があるコンテキストの量の減少であり、その結果、変形例を実施するビデオコーデックによって処理されるシーケンスの大部分の符号化効率にわずかな影響しか及ぼさずに、エンコーダおよびデコーダ側で記憶される必要がある状態の量の減少である。ただし、画面コンテンツの符号化に使用すると、損失が大きくなる可能性があることに留意されたい。この変形例は、他の変形例 / 実施形態と比較した場合、符号化効率と複雑さとの間の別の妥協を表す。符号化効率への影響はしばしば小さい。実際に、利用可能な多数のインター予測モードでは、各インター予測モードのためのインデックスをシグナリングするために必要なデータの平均量が、マージモードのみが使用可能 / 利用可能であるときにマージインデックスをシグナリングするために必要なデータの平均量よりも小さい（この比較が同じシーケンスおよび同じ符号化効率の妥協のためである）。これは、コンテキストに基づくビンの確率の適応からの CABA C 符号化 / 復号の効率が効率的でない可能性があることを意味する。

20

#### 【0400】

第 2 の変形例では、トライアングルマージモードまたは MMVD マージモードの一方または両方が使用される（すなわち、現在の CU の動き情報予測子選択に関する情報が関連するインター予測モードで処理 / 符号化 / 復号される）とき、その / それらのインデックス / フラグはインデックス / フラグを CABA C バイパス符号化 / 復号することによってシグナリングされる。この変形例では、MMVD マージインデックスおよび / またはトライアングルマージインデックスが CABA C バイパス符号化される。実装によって、すなわち、マージモードおよびアフィンマージモードが使用可能な場合、マージインデックスとアフィンマージインデックスには独自のコンテキストがある。さらに別の変形例では、アフィンマージインデックスおよびマージインデックスのコンテキストが共有される。

30

したがって、例えば、CABA C 符号化処理中に、これらのインデックス / フラグを処理するとき、インデックス / フラグの第 1 番目のビットに使用される別個の（独立した）コンテキスト変数の数は：

40

（正規）マージ = アフィンマージ = 実装に応じて 0 または 1 ;  
 MHI I（使用可能な場合） = 0（正規マージと共有） ;  
 トライアングル = MMVD = 0（バイパス符号化）  
 である。

#### 【0401】

これらの変形例の利点は、CABA C コンテキストの削減と符号化効率との間にさらに別の妥協点があるため、以前の変形例と比較して符号化効率が改善されることである。実際、トライアングルマージモードはしばしば選択されない。その結果、そのコンテキストが除去されるとき、すなわち、トライアングルマージモードが CABA C バイパス符号化を使用するとき、符号化効率への影響は小さい。MMVD マージモードはトライアングル

50

マージモードよりも頻繁に選択される傾向があるが、MMVDマージモードの第1番目および第2番目の候補を選択する確率は、マージモード以上またはアフィンマージモードなどの他のインター予測モードよりも等しくなる傾向があり、したがって、MMVDマージモードではCABA符号化のコンテキストを使用することから得られる利点はそれほど大きくない。これらの変形例の別の利点は、画面コンテンツのための最も影響力のあるインター予測モードがマージモードであるので、画面コンテンツシーケンスのための小さい符号化効率の影響である。

#### 【0402】

第3の変形例では、マージモード、トライアングルマージモード、またはMMVDマージモードが使用されるとき（すなわち、現在のCUの動き情報予測子選択に関する情報が関連するインター予測モードで処理／符号化／復号される）、その／それらのインデックス／フラグはインデックス／フラグをCABAバイパス符号化／復号することによってシグナリングされる。この変形例では、MMVDマージインデックス、トライアングルマージインデックス、およびマージインデックスはCABAバイパス符号化される。したがって、例えば、CABA符号化処理中に、これらのインデックス／フラグを処理するとき、インデックス／フラグの第1番目のビットに使用される別個の（独立した）コンテキスト変数の数は：

（正規）マージ = トライアングル（利用可能な場合） = MMVD（利用可能な場合） = 0（バイパス符号化）；  
MHII（使用可能な場合） = 0（正規のマージと同じ）；および  
アフィンマージ = 1  
である。

#### 【0403】

この変形例は、他の変形例と比較して代替の妥協を提供し、例えば、この変形例は、以前の変形例よりも画面コンテンツシーケンスについてより大きな符号化効率減少を与える。

#### 【0404】

第4の変形例では、アフィンマージモード、トライアングルマージモード、またはMMVDマージモードが使用される（すなわち、現在のCUの動き情報予測子選択に関する情報が関連するインター予測モードで処理／符号化／復号される）とき、その／それらのインデックス／フラグはインデックス／フラグをCABAバイパス符号化／復号することによってシグナリングされる。この変形例では、アフィンマージインデックス、MMVDマージインデックス、およびトライアングルマージインデックスはCABAバイパス符号化され、マージインデックスは1つまたは複数のCABAコンテキストで符号化される。したがって、例えば、CABA符号化処理中に、これらのインデックス／フラグを処理するとき、インデックス／フラグの第1番目のビットに使用される別個の（独立した）コンテキスト変数の数は：

（正規）マージ = 1；  
アフィンマージ = トライアングル（使用可能な場合） = MMVD（使用可能な場合） = 0（バイパス符号化）；

MHII（使用可能な場合） = 0（正規のマージと共有）である。以前の変形例と比較してこの変形例の利点は、画面コンテンツシーケンスについての符号化効率の増加である。

#### 【0405】

第5の変形例では、符号化または復号処理で使用するために利用可能なインター予測モードが、インデックス／フラグをシグナリングするために、そのインデックス／フラグをCABAバイパス符号化／復号するが、ただし、前記インター予測モードは候補のリストにATMVP予測子候補を含むことができる、すなわち、前記インター予測モードは利用可能な候補のうちの1つとしてATMVP予測子候補を有することができる場合を除く。この変形例では、インター予測モードがATMVP予測子候補を有することができる場合を除いて、すべてのインター予測モードのすべてのインデックスがCABAバイパス符号化される。したがって、例えば、CABA符号化処理中に、これらのインデックス

10

20

30

40

50

/ フラグを処理するとき、インデックス / フラグの第 1 番目のビットに使用される別個の（独立した）コンテキスト変数の数は：

包含可能な A T M V P 候補を持つ（正規）マージ = 1 ;

アフィンマージ（使用可能な場合）= トライアングル（使用可能な場合）= M M V D（使用可能な場合）= 0（バイパス符号化）；

実装によっては M H I I（使用可能な場合）= 1（正規のマージと共有）または 0

または

包含可能な A T M V P 候補を持つアフィンマージ = 1 ;

（正規）マージ（使用可能な場合）= トライアングル（使用可能な場合）= M M V D（使用可能な場合）= 0（バイパス符号化）；

M H I I（使用可能な場合）= 0（正規 M E R G E と同じ）

である。

#### 【 0 4 0 6 】

この変形例は、自然なシーケンスの大部分にも他の複雑さ / 符号化効率の妥協をもたらす。しかしながら、画面コンテンツシーケンスについては、正規マージ候補リスト内に A T M V P 予測子候補を有することが好ましい場合があることに留意されたい。

#### 【 0 4 0 7 】

第 6 の変形例では、符号化または復号処理で使用するために利用可能なインター予測モードが、前記インター予測モードがスキップモードではない場合（例えば、正規マージスキップモード、アフィンマージスキップモード、トライアングルマージスキップモード、または M M V D マージスキップモードのうちの 1 たない場合）に、インデックス / フラグをシグナリングするために、そのインデックス / フラグを C A B A C バイパス符号化 / 復号する。この変形例では、すべてのインデックスがスキップモードで処理されない、すなわちスキップされない任意の C U に対して C A B A C バイパス符号化される。スキップされた C U（すなわち、スキップモードで処理される C U）についてのインデックスは前述の実施形態 / 変形例に関連して説明された C A B A C 符号化技法のうちのいずれか 1 つを使用して処理され得る（たとえば、第 1 番目のビットのみ、または 2 つ以上のビットはコンテキスト変数を有し、コンテキスト変数は、共有されても共有されなくてもよい）。

#### 【 0 4 0 8 】

図 3 1 は、この変形例を示すインター予測モードの復号処理のフローチャートである。図 3 1 の処理は追加の「スキップモード」決定 / チェックステップ（3 1 2 7）を有し、その後、インデックス / フラグ（「M e r g e \_ i d x」）が C A B A C 復号（3 1 1 9）または C A B A C バイパス復号（3 1 2 8）のいずれかのコンテキストを使用して復号されることを除いて、図 2 9 と同様である。さらに別の変形例によれば、前のステップで行われた決定 / チェックの結果、C U はスキップ（2 9 0 2 / 3 1 0 2）、M M V D スキップ（2 9 0 8 / 3 1 0 8）であり、C U は、図 2 9 または図 3 1 のスキップ（2 9 1 6 / 3 1 1 6）を使用して、追加の「スキップモード」決定 / チェックステップ（3 1 2 7）の代わりに「スキップモード」決定 / チェックを行うことが理解される。

#### 【 0 4 0 9 】

この変形例は、スキップモードが一般に非スキップモード（すなわち、正規マージモード、M H I I マージモード、アフィンマージモード、トライアングルマージモード、または M M V D マージモードなどの非スキップインター予測モード）よりも頻繁に選択されるため、また第 1 の候補の選択が非スキップモードよりもスキップモードの方が可能性が高いため、符号化効率に及ぼす影響が低い。S K I P モードは、より予測可能な動きのために設計されているので、それらのインデックスもより予測可能でなければならない。したがって、C A B A C 符号化 / 復号を利用する確率は、スキップモードにとてより有用である可能性が高い。しかしながら、非スキップモードは予測子候補からのよりランダムな選択が生じる可能性がより高くなるように、動きがあまり予測可能でないときに使用される可能性が高い。したがって、非スキップモードでは、C A B A C 符号化 / 復号は効率的である可能性が低い。

10

20

30

40

50

## 【0410】

## 第20の実施形態

第20の実施形態によれば、データはビットストリームで提供され、前記データはC A B A Cバイパス符号化／復号、別個のコンテキスト変数を有するC A B A C符号化／復号、または1つまたは複数の共有コンテキスト変数を有するC A B A C符号化／復号を使用することによって、インター予測モードのうちの1つまたは複数のためのインデックス／フラグがシグナリングされるべきかどうかを判定するためのものである。例えば、そのような前記データは、インター予測モードのインデックス符号化／復号のための1つまたは複数の独立したコンテキストの使用をイネーブルまたはディセーブルするためのフラグであり得る。このようなデータを用いて、C A B A C符号化／復号またはC A B A Cバイパス符号化／復号におけるコンテキスト共有の使用または不使用を制御することが可能である。

10

## 【0411】

第20の実施形態の変形例では、2つ以上のインター予測モードの2つ以上のインデックス間で共有するC A B A Cコンテキストが例えば、C Uレベルよりも高いレベルで（例えば、シーケンス、フレーム、スライス、タイル、またはC T Uレベルなどの最小のC Uよりも大きい画像部分のレベルで）、ビットストリームで送信されるデータに依存する。例えば、このデータは、特定の画像部分内の任意のC Uについて、マージモードのマージインデックスのC A B A Cコンテキストが別のインター予測モードの1つまたは複数の他のC A B A Cコンテキストと共有される（または共有されない）ことを示し得る。

20

## 【0412】

別の変形例では、1つまたは複数のインデックスが、ビットストリームにおいて、例えば、C Uレベルよりも高いレベルで（例えば、スライスレベルで）送信されるデータに応じてC A B A Cバイパス符号化される。例えば、このデータは、特定の画像部分内の任意のC Uについて、特定のインター予測モードのインデックスがC A B A Cバイパス符号化されるべきであることを示し得る。

## 【0413】

一変形例では、符号化効率をさらに改善するために、エンコーダ側で、1つまたは複数のインター予測モードのうちの1つまたは複数のインデックスのコンテキストの共有を示すためのこのデータの値、またはC A B A Cバイパス符号化／復号は、前に符号化されたフレームにおいて1つまたは複数のインター予測モードがどれくらいの頻度で使用されるかに基づいて選択することができる。代替案は、処理されるシーケンスのタイプ、または変形例が実装されるアプリケーションのタイプに基づいて、このデータの値を選択することであってもよい。

30

## 【0414】

この実施形態の利点は、先の実施形態／変形例と比較して、制御された符号化効率の増加である。

## 【0415】

## 本発明の実施形態の実施

前述の実施形態のうちの1つまたは複数は、図3の処理デバイス300のプロセッサ311、または図5のデコーダ60の、図17のC A B A Cコーダの、図4のエンコーダ400の対応する機能モジュール／ユニット、またはその対応するC A B A Cデコーダによって実装され、1つまたは複数の前述の実施形態の方法ステップを実行する。

40

## 【0416】

図19は、本発明の1つまたは複数の実施形態の実施のためのコンピューティングデバイス2000の概略ブロック図である。コンピューティングデバイス2000は、マイクロコンピュータ、ワークステーション、またはライトポータブルデバイスなどのデバイスであってもよい。コンピューティングデバイス2000は、-マイクロプロセッサなどの中央処理装置（C P U）2001；-本発明の実施形態の方法の実行可能コードを記憶するためのランダムアクセスメモリ（R A M）2002ならびに本発明の実施形態に係る画

50

像の少なくとも一部を符号化または復号するための方法を実現するために必要な変数およびパラメータを記録するためのレジスタ、これらのメモリ容量が例えば、拡張ポートに接続されたオプションのRAMによって拡張することができる； - 本発明の実施形態を実現するためのコンピュータプログラムを記憶するリードオンリーメモリ(ROM)2003； - 処理されるデジタルデータが送信または受信される通信ネットワークに典型的に接続されるネットワークインターフェース(NET)2004、に接続された通信バスを備える。ネットワークインターフェース(NET)2004は、単一のネットワークインターフェースであってもよいし、異なるネットワークインターフェース(例えば、有線および無線インターフェース、または異なる種類の有線または無線インターフェース)のセットで構成されてもよい。データパケットは、CPU2001で実行されるソフトウェアアプリケーションの制御の下で、送信のためにネットワークインターフェースに書き込まれるか、または受信のためにネットワークインターフェースから読み出される。 - ユーザからの入力を受信したり、ユーザに情報を表示するためにユーザインターフェース(UI)2005が使用されてもよい。 - 大容量記憶装置としてハードディスク(HD)2006が提供されてもよい。 - 入力/出力モジュール(I/O)2007が、ビデオソースやディスプレイなどの外部装置との間でデータを送受信するために使用されてもよい。実行可能コードは、ROM2003、HD2006、または例えばディスクのようなリムーバブルデジタル媒体のいずれかに格納することができる。変形例によれば、プログラムの実行可能コードは、実行される前に、HD2006などの通信装置2000の記憶手段の1つに記憶されるために、NET2004を介して、通信ネットワークによって受信することができる。CPU2001は、前述の記憶手段の1つに命令が格納されている、本発明の実施形態によるプログラムまたはプログラムのソフトウェアコードの命令または部分の実行を制御および指示するように適合される。電源投入後、CPU2001は、例えば、プログラムROM2003またはHD2006からこれらの命令がロードされた後に、メインRAMメモリ2002から、ソフトウェアアプリケーションに関する命令を実行することができる。このようなソフトウェアアプリケーションは、CPU2001によって実行されると、本発明による方法のステップを実行させる。

#### 【0417】

また、本発明の他の実施形態によれば、コンピュータ、携帯電話(携帯電話)、タブレット、またはユーザにコンテンツを提供/表示することができる他の任意のタイプのデバイス(例えば、ディスプレイ装置)などのユーザ端末に、前述の実施形態によるデコーダが提供されることも理解される。さらに別の実施形態によれば、前述の実施形態によるエンコーダは、エンコーダがエンコードするためのコンテンツをキャプチャおよび提供するカメラ、ビデオカメラ、またはネットワークカメラ(例えば、閉回路テレビまたはビデオ監視カメラ)も備える画像キャプチャ装置において提供される。2つのこのような例を、図20および21を参照して以下に提供する。

#### 【0418】

図20は、ネットワークカメラ2102及びクライアント装置2104を含むネットワークカメラシステム2100を示す図である。

#### 【0419】

ネットワークカメラ2102は、撮像部2106と、符号化部2108と、通信部2110と、制御部2112とを有している。

#### 【0420】

ネットワークカメラ2102とクライアント装置2104とは、ネットワーク200を介して互いに通信可能に相互に接続されている。

#### 【0421】

撮像部2106は、レンズおよび撮像素子(例えば、電荷結合素子(CCD)または相補型金属酸化膜半導体(CMOS))を含み、物体の画像を撮像し、その画像に基づいて画像データを生成する。この画像は静止画像であってもよいし、ビデオ画像であってもよい。また、撮像部は、(光学的またはデジタル的に)ズームまたはパンするように適合さ

10

20

30

40

50

れたズーム手段および／またはパン手段を備えてもよい。

【0422】

符号化部2108は、前述の実施形態のうち1つもしくは複数で説明した前記符号化方法を用いて画像データを符号化する。符号化部2108は、前述の実施形態で説明した符号化方法の少なくとも1つを用いる。他の例では、符号化部2108は、前述の実施形態で説明した符号化方法の組合せを用いることができる。

【0423】

ネットワークカメラ2102の通信部2110は、符号化部2108で符号化された符号化画像データをクライアント装置2104に送信する。また、通信部2110は、クライアント装置2104からのコマンドを受信する。コマンドは、符号化部2108の符号化のためのパラメータを設定するコマンドを含む。

10

【0424】

制御部2112は、通信部2110が受信したコマンドに従って、ネットワークカメラ2102内の他のユニットを制御する。

【0425】

クライアント装置2104は、通信部2114と、復号部2116と、制御部2118とを有する。クライアント装置2104の通信部2114は、ネットワークカメラ2102にコマンドを送信する。また、クライアント装置2104の通信部2114は、ネットワークカメラ2102から符号化画像データを受信する。

20

【0426】

復号部2116は、前述の実施形態のうち1つもしくは複数で説明した前記復号方法を用いて、符号化画像データを復号する。他の例では、復号部2116は、前述の実施形態で説明した復号方法の組合せを用いることができる。

【0427】

クライアント装置2104の制御部2118は、通信部2114が受信したユーザ操作やコマンドに従って、クライアント装置2104内の他のユニットを制御する。クライアント装置2104の制御部2118は、復号部2116で復号された画像を表示するよう表示装置2120を制御する。また、クライアント装置2104の制御部2118は、GUI(Graphical User Interface)を表示するよう表示装置2120を制御し、符号化部2108の符号化のためのパラメータを含むネットワークカメラ2102のパラメータの値を指定する。

30

【0428】

また、クライアント装置2104の制御部2118は、表示装置2120が表示するGUIに対するユーザ操作入力に応じて、クライアント装置2104内の他のユニットを制御する。クライアント装置2104の制御部2118は、表示装置2120が表示するGUIに対するユーザ操作入力に応じて、ネットワークカメラ2102のパラメータの値を指定するコマンドをネットワークカメラ2102に送信するよう、クライアント装置2104の通信部2114を制御する。

【0429】

ネットワークカメラシステム2100は、ビデオの記録中にカメラ2102がズームまたはパンを利用するかどうかを判定することができ、撮影中のズームまたはパンは、ズーム、回転、および／または伸張（特にレンズが「魚眼」レンズである場合、パンニングの副作用であり得る）などの複雑な動きを符号化するのによく適したアフィンモードの使用から利益を得ることができるので、そのような情報はビデオストリームを符号化するときに使用することができる。

40

【0430】

図21は、スマートフォン2200を示す図である。

【0431】

スマートフォン2200は、通信部2202と、復号／符号化部2204と、制御部2206と、表示部2208とを備える。

50

**【 0 4 3 2 】**

通信部 2 2 0 2 は、ネットワーク 2 0 0 を介して符号化画像データを受信する。

**【 0 4 3 3 】**

復号 / 符号化部 2 2 0 4 は、通信部 2 2 0 2 が受信した符号化画像データを復号する。復号 / 符号化部 2 2 0 4 は、前述の実施形態うち 1 つもしくは複数で説明した前記復号方法を用いて、符号化画像データを復号する。復号 / 符号化部 2 2 0 4 は、前述の実施形態で説明した復号方法の少なくとも 1 つを用いることができる。他の例では、復号 / 符号化部 2 2 0 4 は、前述の実施形態で説明した復号又は符号化方法の組合せを用いることができる。

**【 0 4 3 4 】**

制御部 2 2 0 6 は、通信部 2 2 0 2 によりまたは入力部を介して受信したユーザ操作やコマンドに応じて、スマートフォン 2 2 0 0 内の他のユニットを制御する。例えば、制御部 2 2 0 6 は、復号部 2 2 0 4 により復号された画像を表示するように表示装置 2 2 0 8 を制御する。

10

**【 0 4 3 5 】**

スマートフォンは、画像またはビデオを記録するための画像記録デバイス 2 2 1 0 ( 例えば、デジタルカメラおよび関連する回路 ) をさらに備えることができる。このような記録された画像やビデオは、制御部 2 2 0 6 の指示の下、復号 / 符号化部 2 2 0 4 によって符号化されてもよい。スマートフォンはさらに、モバイルデバイスの向きを感知するように構成されたセンサ 2 2 1 2 を備えてもよい。このようなセンサは、加速度計、ジャイロスコープ、コンパス、全地球測位 ( G P S ) ユニット又は同様の位置センサを含むことができる。そのようなセンサ 2 2 1 2 は、スマートフォンが向きを変えているかどうかを判定することができ、そのような情報は、撮影中の向きの変化としてビデオストリームを符号化するときに使用され、回転のような複雑な動きを符号化するのによく適したアフィンモードの使用から利益を得ることができる。

20

**【 0 4 3 6 】****代替および変更**

本発明の目的は、アフィンモードが最も効率的な方法で利用されることを保証することであり、上述の特定の例は、アフィンモードが有用であると知覚される可能性に応じて、アフィンモードの使用をシグナリングすることに関することが理解されるのであろう。これのさらなる例は、複雑な動き ( アフィン変換が特に効率的である場合がある ) が符号化されていることが知られている場合に、エンコーダに適用され得る。このような場合の例は、

30

- a ) カメラズームイン / アウト
- b ) 撮影 ( すなわち、回転運動 ) 中に向きを変えるポータブルカメラ ( 例えば、携帯電話 )
- c ) 「魚眼」レンズカメラのパンニング ( 例えば、画像の一部の伸張 / 歪曲を含む。 )

**【 0 4 3 7 】**

したがって、アフィンモードがスライス、フレームシーケンス、または実際にビデオストリーム全体のために使用される可能性が高くなるように、記録処理中に複雑な動きの指示を上げることができる。

40

**【 0 4 3 8 】**

さらなる例では、アフィンモードが、ビデオを記録するために使用されるデバイスの特徴または機能性に応じて、使用される可能性がより高い。例えば、モバイルデバイスは、( 例えば ) 固定セキュリティカメラよりも向きを変える可能性が高いので、アフィンモードは、前者からのビデオを符号化するのにより適している可能性がある。特徴または機能の例には、ズーム手段の存在 / 使用、位置センサの存在 / 使用、パン手段の存在 / 使用、デバイスが携帯型であるか否か、またはデバイス上のユーザ選択が含まれる。

**【 0 4 3 9 】**

50

本発明を実施形態を参照して説明してきたが、本発明は、開示された実施形態に限定されないことを理解されたい。添付の特許請求の範囲に定義されるように、本発明の範囲から逸脱することなく、様々な変更および修正を行うことができることは、当業者には理解されよう。本明細書（任意の添付の特許請求の範囲、要約書、および図面を含む）に開示された特徴のすべて、および／またはそのように開示された任意の方法または処理のステップのすべては、そのような特徴および／またはステップの少なくともいくつかが相互に排他的である組合せを除いて、任意の組合せで組み合わせることができる。本明細書（任意の添付の特許請求の範囲、要約書、および図面を含む）に開示される各特徴は、特に断らない限り、同じ、同等の、または同様の目的を果たす代替の特徴によって置き換えることができる。したがって、特に断らない限り、開示される各特徴は、同等または同様の特徴の一般的なシリーズの一例にすぎない。

#### 【 0 4 4 0 】

また、上述の比較、判定、評価、選択、実行、実行、または考慮の任意の結果、例えば、符号化またはフィルタリング処理中に行われる選択は、ビットストリーム内のデータ、例えば、結果を示すフラグまたはデータに示されるか、またはそれらから決定可能／推論可能でありえ、その結果、示されるか、または決定された／推論された結果は、例えば、復号処理中に、比較、判定、評価、選択、実行、実行、または考慮を実際に実行する代わりに、処理において使用され得ることが理解される。

#### 【 0 4 4 1 】

特許請求の範囲において、単語「有する」は、他の要素又はステップを排除するものではなく、不定冠詞「 a 」又は「 a n 」は複数を排除するものではない。異なる特徴が相互に異なる従属請求項に記載されているという単なる事実は、これらの特徴の組合せが有利に使用されることを示すものではない。

#### 【 0 4 4 2 】

特許請求の範囲に記載されている参照符号は、例示のみを目的としたものであり、クレームの範囲に限定的な影響を及ぼさない。

#### 【 0 4 4 3 】

前述の実施形態では、説明された機能がハードウェア、ソフトウェア、ファームウェア、またはそれらの任意の組合せで実装され得る。ソフトウェアで実施される場合、機能は、1つ以上の命令またはコードとして、コンピュータ可読媒体上に記憶され、またはそれを介して送信され、ハードウェアベースの処理ユニットによって実行されてもよい。

#### 【 0 4 4 4 】

コンピュータ可読媒体は、データ記憶媒体のような有形媒体に対応するコンピュータ可読記憶媒体、または例えば通信プロトコルに従って、ある場所から別の場所へのコンピュータプログラムの転送を容易にする任意の媒体を含む通信媒体を含み得る。このようにして、コンピュータ可読媒体は一般に、（1）非一時的である有形のコンピュータ可読記憶媒体、または（2）信号または搬送波などの通信媒体に対応することができる。データ記憶媒体は、本開示に記載される技術の実施のための命令、コードおよび／またはデータ構造を検索するために、1つ以上のコンピュータまたは1つ以上のプロセッサによってアクセス可能な任意の利用可能な媒体であってもよい。コンピュータプログラム製品は、コンピュータ可読媒体を含み得る。

#### 【 0 4 4 5 】

限定ではなく、一例として、このようなコンピュータ可読記憶媒体は、RAM、ROM、EEPROM、CD-ROM又は他の光ディスク記憶装置、磁気ディスク記憶装置、又は他の磁気記憶装置、フラッシュメモリ、又は所望のプログラムコードを命令又はデータ構造の形成で記憶するために使用することができ、コンピュータによってアクセスすることができる他の任意の媒体を含むことができる。また、任意のコネクションは、コンピュータ可読媒体と適切に呼ばれる。例えば、命令が同軸ケーブル、光ファイバケーブル、ツイストペア、デジタル加入者回線（DSL）、または赤外線、無線、およびマイクロ波などの無線技術を使用して、ウェブサイト、サーバ、または他のリモートソースから送信さ

れる場合、同軸ケーブル、光ファイバケーブル、ツイストペア、D S L、または赤外線、無線、およびマイクロ波などの無線技術は、媒体の定義に含まれる。しかし、コンピュータ可読記憶媒体およびデータ記憶媒体は、接続、搬送波、信号、または他の一時的な媒体を含まず、代わりに非一時的な有形の記憶媒体を対象とすることを理解されたい。本明細書で使用されるディスク（D i s k）およびディスク（d i s k）は、コンパクトディスク（C D）、レーザディスク、光ディスク、デジタル多用途ディスク（D V D）、フロッピー（登録商標）ディスク、およびブルーレイディスクを含み、ここで、ディスクは、通常、磁気的にデータを再生し、ディスクは、レーザで光学的にデータを再生する。上記の組合せは、コンピュータ読み取り可能な媒体の範囲内にも含まれるべきである。

## 【 0 4 4 6 】

10

命令は、1つまたは複数のデジタル信号プロセッサ（D S P）、汎用マイクロプロセッサ、特定用途向け集積回路（A S I C）、フィールドプログラマブルゲート／論理アレイ（F P G A）、または他の同等の集積またはディスクリート論理回路などの1つまたは複数のプロセッサによって実行され得る。したがって、本明細書で使用される用語「プロセッサ」は、前述の構造のいずれか、または本明細書で説明される技術の実装に適した他の任意の構造を指すことがある。さらに、いくつかの態様では、本明細書に記載する機能性が、符号化および復号のために構成された専用のハードウェアおよび／またはソフトウェアモジュール内で提供されてもよく、あるいは結合されたコーデックに組み込まれてもよい。また、本技術は、1つまたは複数の回路または論理素子で完全に実装することができる。

20

30

40

50

## 【図面】

## 【図 1】

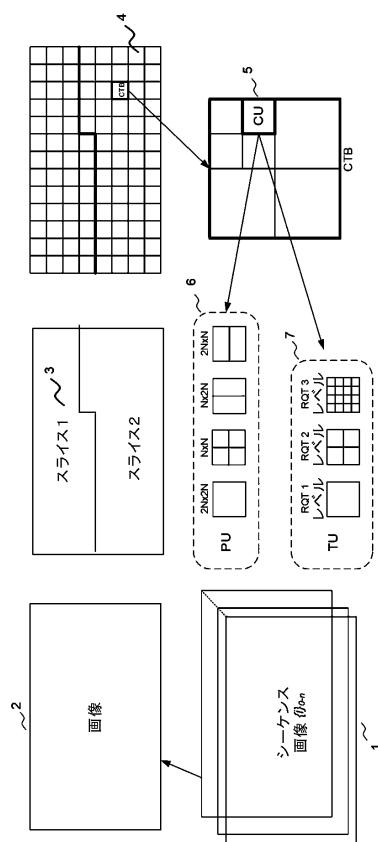


Figure 1

## 【図 2】



Figure 2

10

20

30

40

## 【図 3】

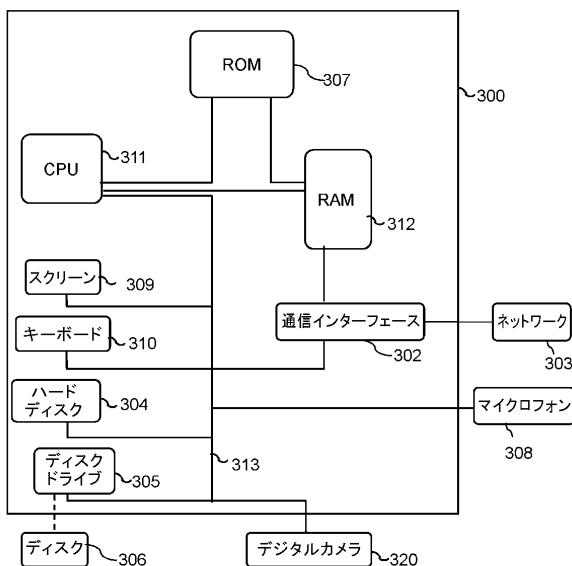


Figure 3

## 【図 4】

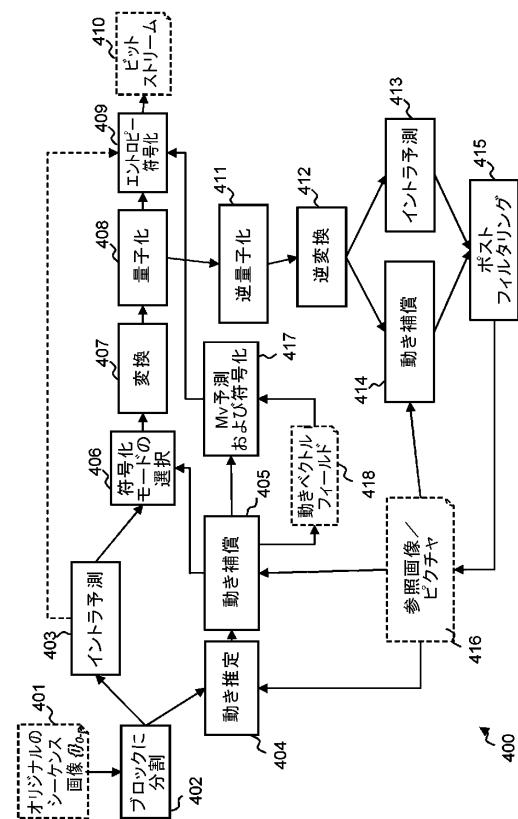
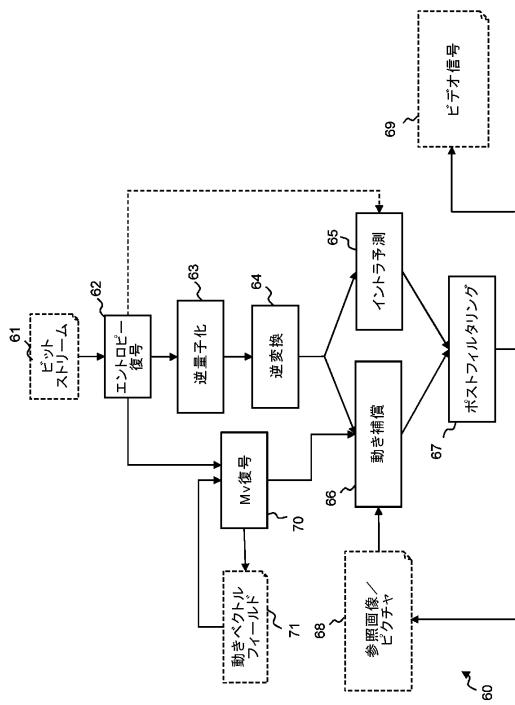


Figure 4

50

【図 5】



【図 6 a】

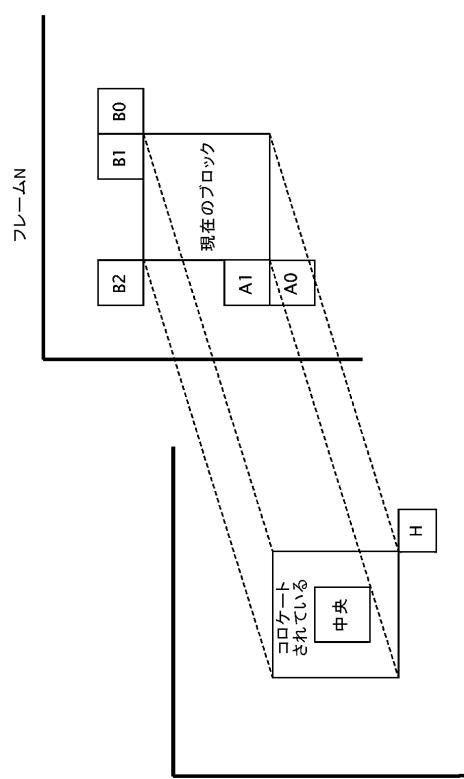


Figure 6a

10

20

30

40

【図 6 b】

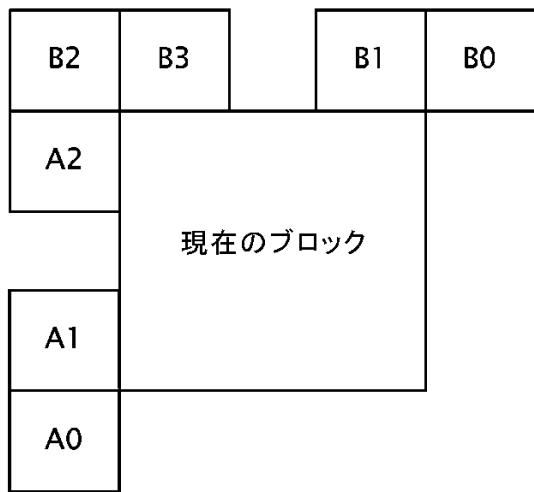


Figure 6b

【図 7】

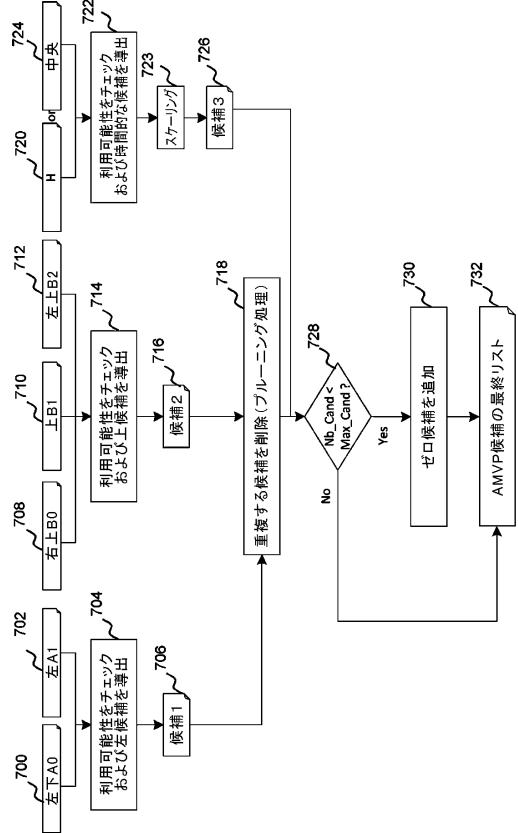
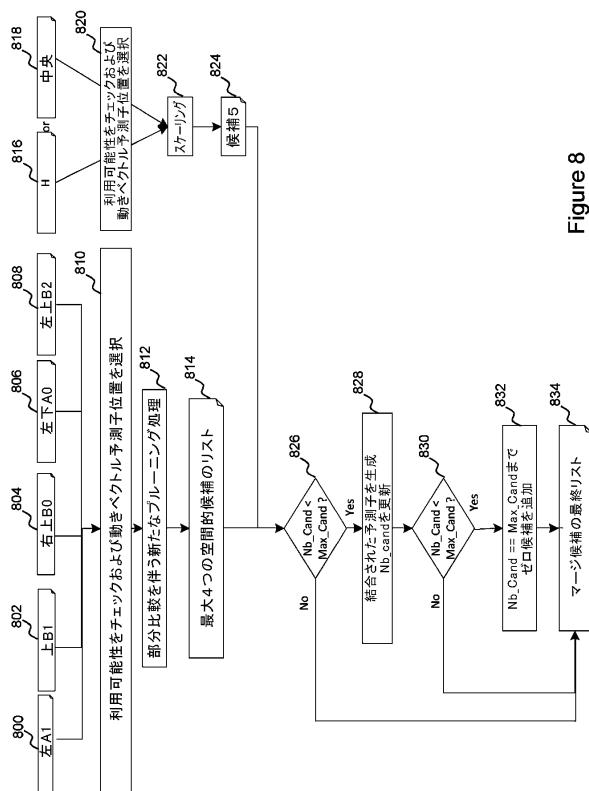


Figure 7

50

【図 8】



【図 9】

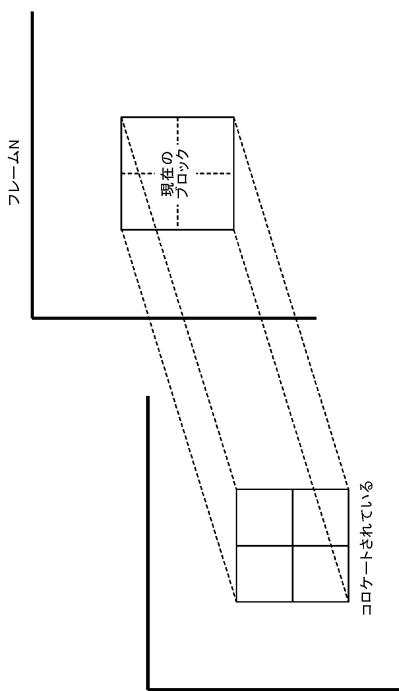


Figure 9

10

20

【図 10】

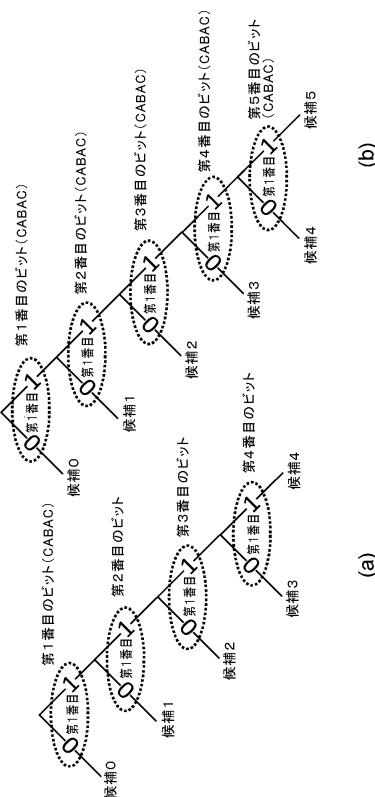


Figure 10

30

【図 11】

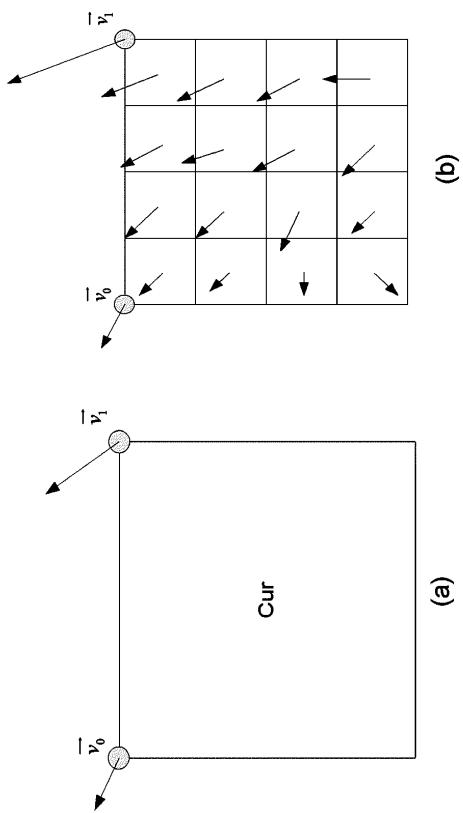
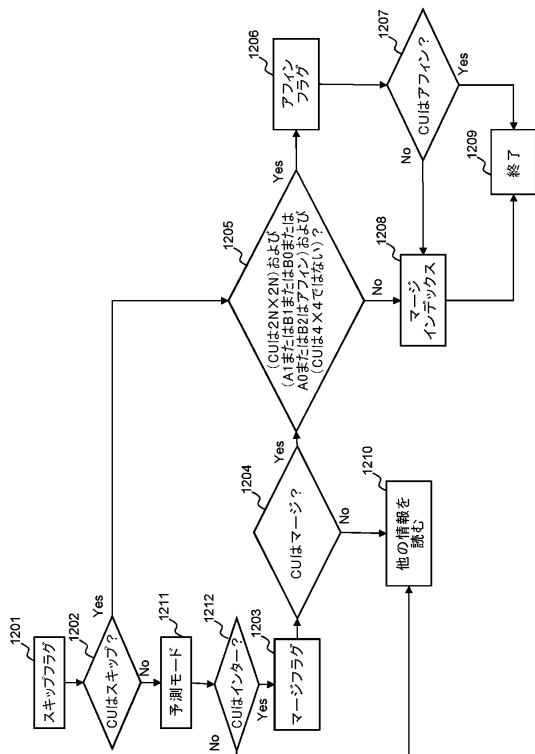


Figure 11

40

50

【図 1 2】



【図 1 3】

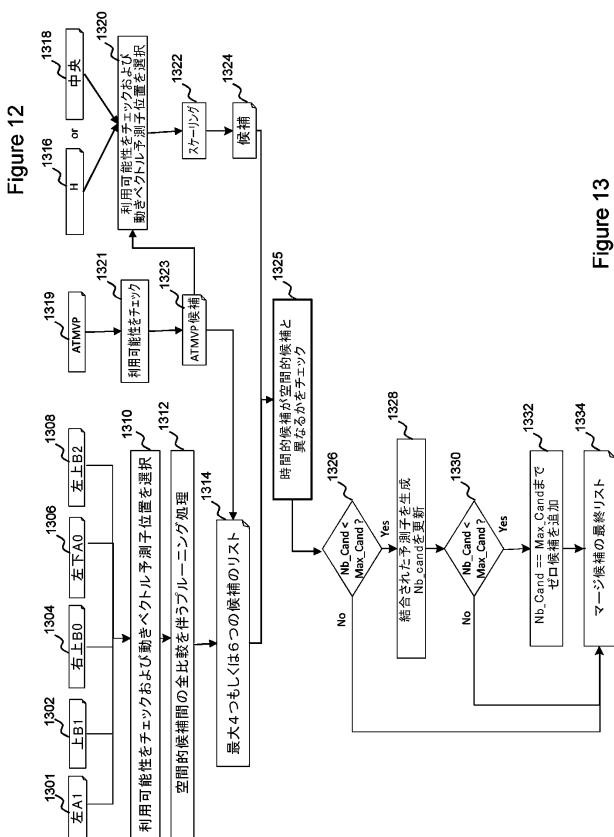


Figure 13

【図 1 4】

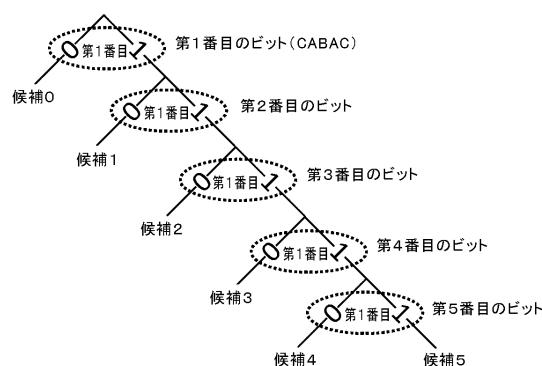


Figure 14

【図 1 5】

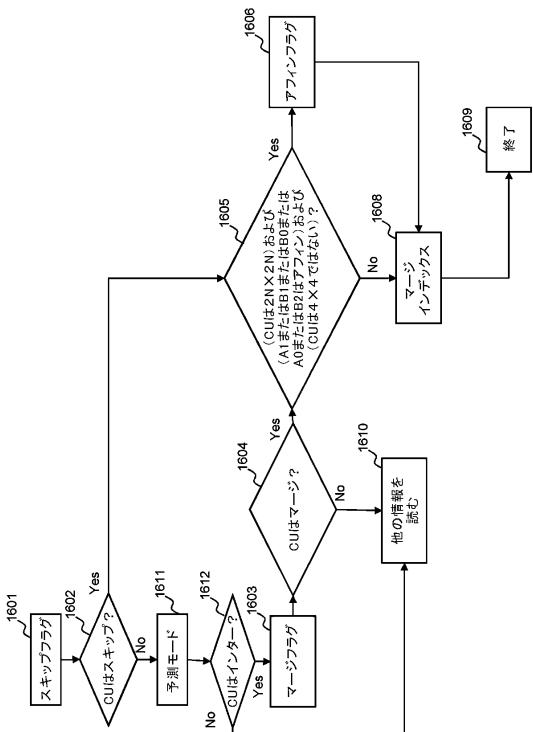
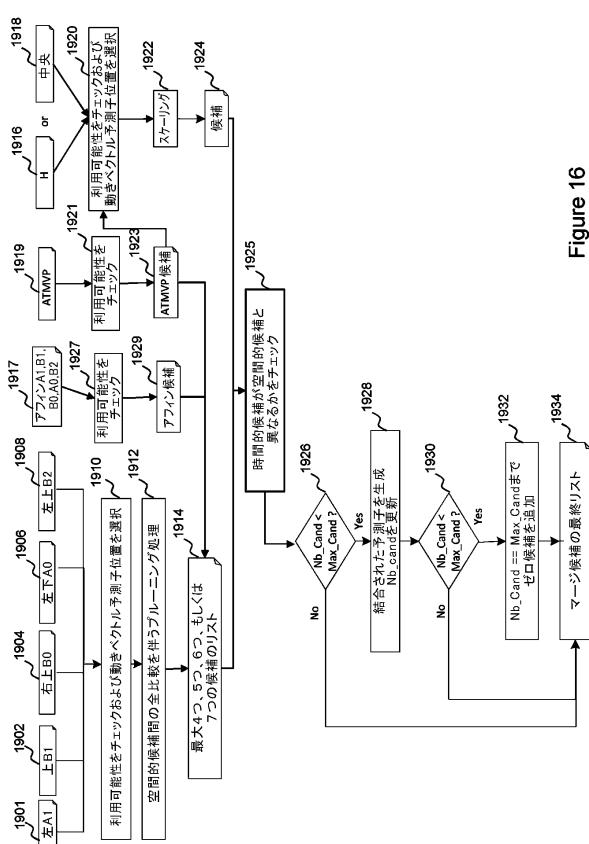


Figure 15

【図 16】



【図 17】

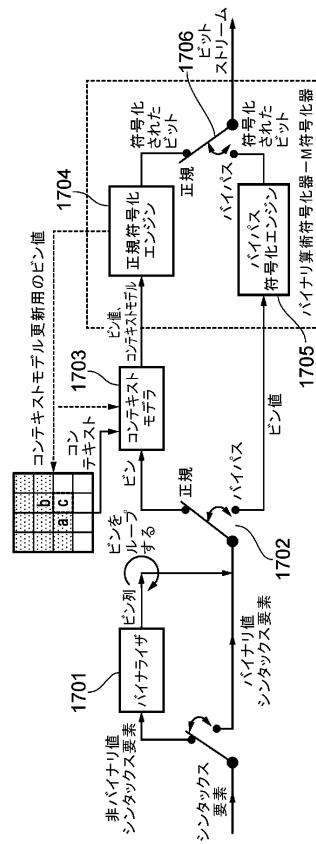


Figure 17

【図 18】

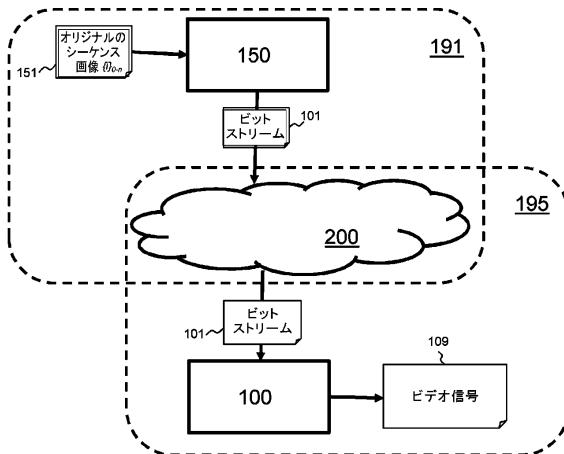


Figure 18

【図 19】

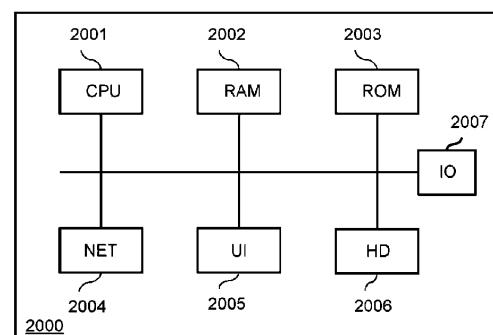
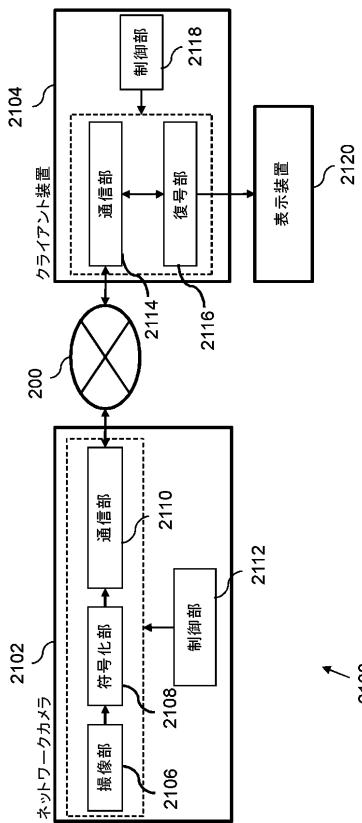


Figure 19

【図20】



【 図 2 1 】

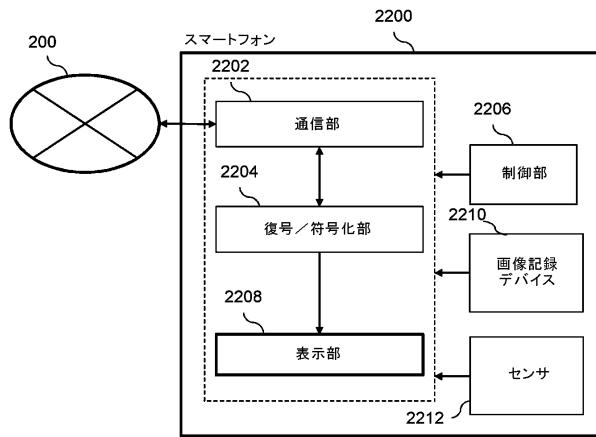
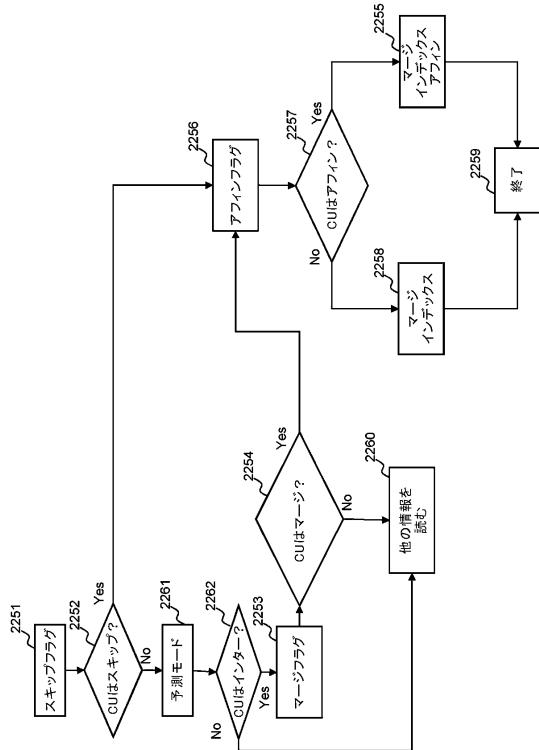


Figure 21

## 【図22】



### 【図23】

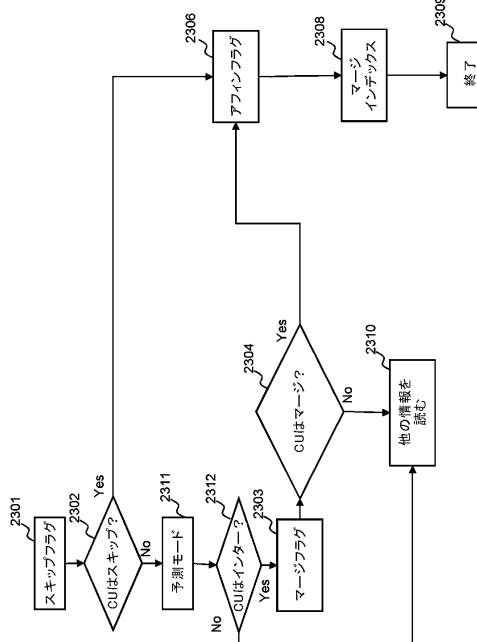
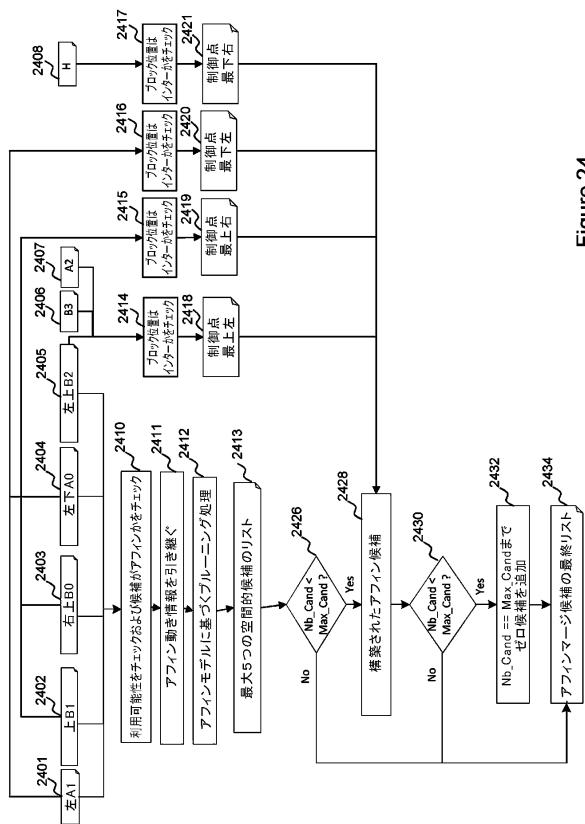
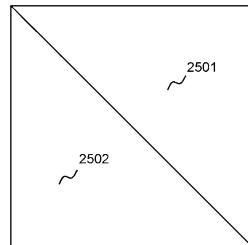


Figure 23

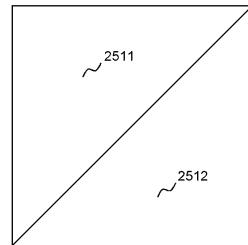
### 【図24】



【 図 25 】



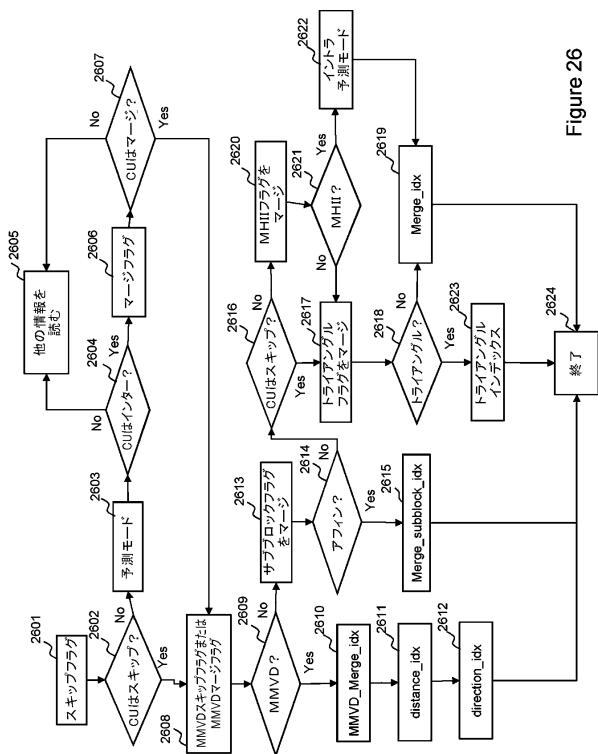
(b)



(b)

Figure 25

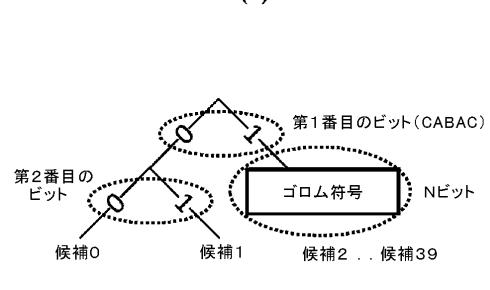
【図26】



【図27】



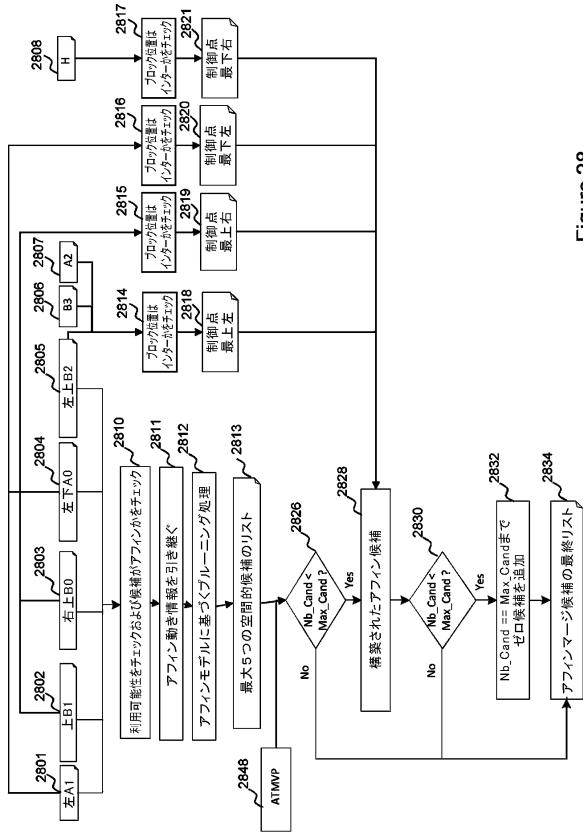
(a)



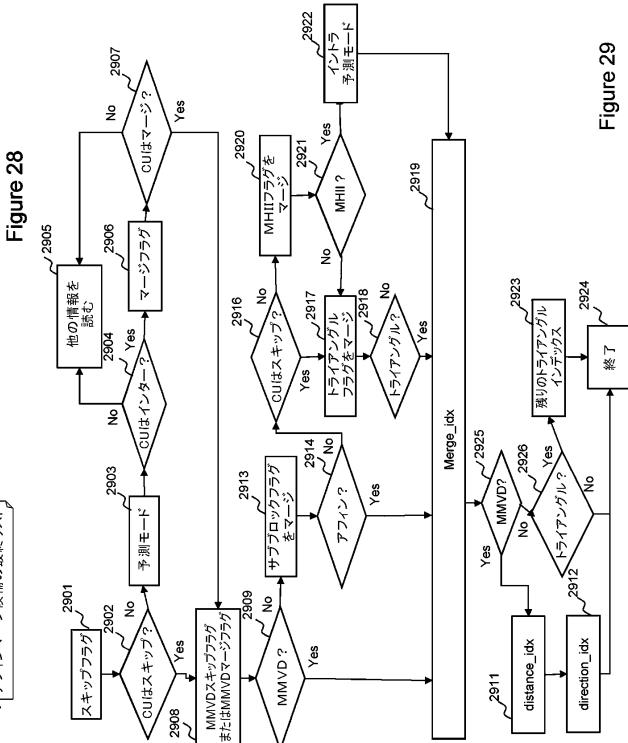
(b)

Figure 27

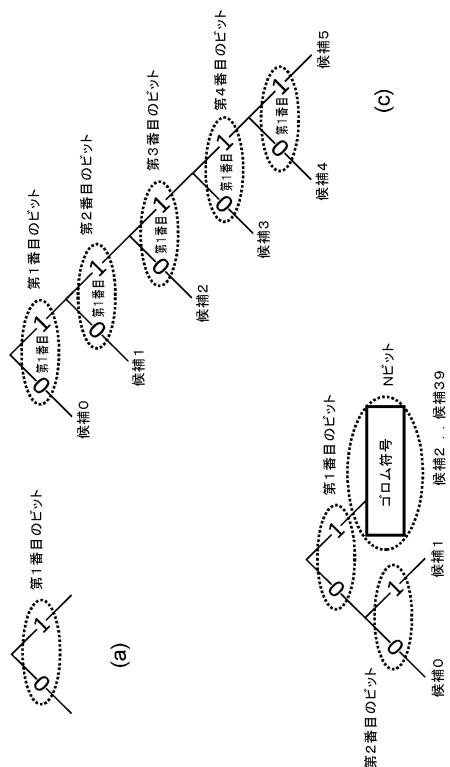
【 図 2 8 】



【図29】



【 図 3 0 】



【図31】

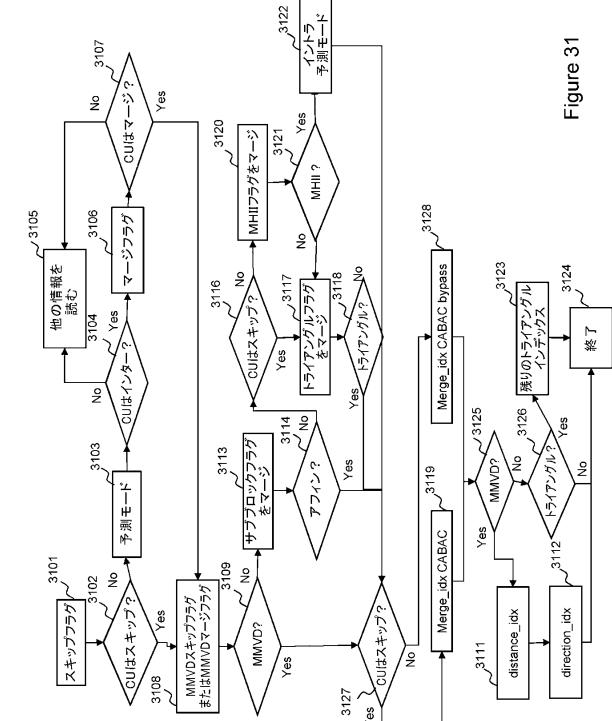


Figure 29

---

フロントページの続き

ル キヤノン リサーチ センター フランス エス . エー . エス 内

(72)発明者 オンノ , パトリス

フランス国 レンヌ - アタラント , セデックス セッソン - セヴィニエ 35517 , リュ ドゥ  
ラ トゥッシュランペール キヤノン リサーチ センター フランス エス . エー . エス 内

(72)発明者 タケ , ジョナサン

フランス国 レンヌ - アタラント , セデックス セッソン - セヴィニエ 35517 , リュ ドゥ  
ラ トゥッシュランペール キヤノン リサーチ センター フランス エス . エー . エス 内

審査官 松元 伸次

(56)参考文献 特表 2013 - 530658 (JP, A)

特表 2014 - 531819 (JP, A)

(58)調査した分野 (Int.Cl. , DB名)

H04N 19/00 - 19/98