(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7620055号
(P7620055)

(45)発行日　令和7年1月22日(2025.1.22)

(24)登録日　令和7年1月14日(2025.1.14)

(51)国際特許分類		FI	
H04N	*19/11 (2014.01)*	H04N	19/11
H04N	*19/186 (2014.01)*	H04N	19/186
H04N	*19/70 (2014.01)*	H04N	19/70

請求項の数　15　外国語出願　（全97頁）

(21)出願番号　特願2023-126598(P2023-126598)
(22)出願日　令和5年8月2日(2023.8.2)
(62)分割の表示　特願2021-578165(P2021-578165)の分割
原出願日　令和2年6月28日(2020.6.28)
(65)公開番号　特開2023-162203(P2023-162203A)
(43)公開日　令和5年11月8日(2023.11.8)
審査請求日　令和5年8月31日(2023.8.31)
(31)優先権主張番号　PCT/CN2019/093852
(32)優先日　令和1年6月28日(2019.6.28)
(33)優先権主張国・地域又は機関　中国(CN)
(31)優先権主張番号　PCT/CN2019/093982
(32)優先日　令和1年6月29日(2019.6.29)
最終頁に続く

(73)特許権者　520477474
バイトダンス インコーポレイテッド
ＢＹＴＥＤＡＮＣＥ ＩＮＣ．
アメリカ合衆国 カリフォルニア州 ９００６６，ロサンジェルス，ウェスト ジェファーソン ブールヴァード １２６５５，シックスス フロアー，スイート ナンバー・１３７
１２６５５ West Jefferson Boulevard, Sixth Floor, Suite No. 137 Los Angeles, California 90066 United States of America
(74)代理人　110002000
最終頁に続く

(54)【発明の名称】　スクリーンコンテンツ符号化におけるクロマイントラモードの導出

(57)【特許請求の範囲】

【請求項１】

映像データを処理する方法であって、

映像の現在のクロマブロックと前記映像のビットストリームとの変換のために、前記現在のクロマブロックに対応する第１の輝度ブロックの輝度コーディング情報に基づいて、前記現在のクロマブロックのクロマ予測モードを導出することと、

導出された前記クロマ予測モードに基づいて前記変換を行うことと、を含み、

前記第１の輝度ブロックが、輝度予測モードで導出された量子化残差と前記量子化残差の予測値との差分が前記ビットストリームに含まれる差分コーディングモードでコーディングされることに応じて、

前記輝度予測モードは、垂直予測モード又は水平予測モードであり、

前記クロマ予測モードは、前記輝度予測モードと同一であり、

前記第１の輝度ブロックが第２の輝度ブロックの隣接ブロックである場合、前記第２の輝度ブロックと前記第１の輝度ブロックとが異なるスライス又はタイルにあることに応じて、前記第２の輝度ブロックのイントラモード候補リストを構築するために、デフォルトモードが前記第１の輝度ブロックに割り当てられる、

方法。

【請求項２】

前記第１の輝度ブロックは、前記現在のクロマブロックの対応する中心位置をカバーする、

請求項１に記載の方法。
【請求項３】
前記クロマ予測モードが前記輝度予測モードと同一であることは、更に、構文要素ｉｎｔｒａ＿ｃｈｒｏｍａ＿ｐｒｅｄ＿ｍｏｄｅの値に基づいており、
前記構文要素ｉｎｔｒａ＿ｃｈｒｏｍａ＿ｐｒｅｄ＿ｍｏｄｅは、クロマ成分イントラモードと輝度成分イントラモードとのマッピング関係を規定するために使用される、
請求項１又は２に記載の方法。
【請求項４】
前記構文要素ｉｎｔｒａ＿ｃｈｒｏｍａ＿ｐｒｅｄ＿ｍｏｄｅは、導出モード（Ｄｅｒｉｖｅｄ　Ｍｏｄｅ（ＤＭ））が前記現在のクロマブロックに適用されることを規定し、
前記導出モードにおいて、前記クロマ成分イントラモードは、前記輝度成分イントラモードと同一である、
請求項３に記載の方法。
【請求項５】
前記輝度予測モードが前記垂直予測モードである場合、前記クロマ予測モードは前記垂直予測モードである、
請求項１から４のいずれか１項に記載の方法。
【請求項６】
前記輝度予測モードが前記水平予測モードである場合、前記クロマ予測モードは前記水平予測モードである、
請求項１から５のいずれか１項に記載の方法。
【請求項７】
前記第１の輝度ブロックが第１の予測モード又は前記第１の予測モードの変形でコーディングされることに応じて、前記クロマ予測モードがＤＣモードとして導出され、
前記第１の予測モードにおいて、再構成されたサンプルが、代表的な色値のセットによって表され、
前記代表的な色値のセットは、パレット予測子、エスケープサンプル、又は前記ビットストリームに含まれるパレット情報のうちの少なくとも１つを含む、
請求項３に記載の方法。
【請求項８】
前記第１の輝度ブロックが第２の予測モード又は第２の予測モードの変形でコーディングされることに応じて、前記クロマ予測モードがＤＣモードとして導出され、
前記第２の予測モードにおいて、予測サンプルは、ブロックベクトルによって決定される同じ復号ピクチャのサンプル値のブロックから導出される、
請求項３に記載の方法。
【請求項９】
前記第１の輝度ブロックが前記第２の輝度ブロックの隣接ブロックである場合、前記第１の輝度ブロックが第１の予測モード又は第２の予測モードでコーディングされることに応じて、前記第２の輝度ブロックの前記イントラモード候補リストを構築するために、前記デフォルトモードが前記第１の輝度ブロックに割り当てられ、
前記第１の予測モードでは、再構成されたサンプルは、代表的な色値のセットによって表され、
前記代表的な色値のセットは、パレット予測子、エスケープサンプル、又は前記ビットストリームに含まれるパレット情報のうちの少なくとも１つを含み、
前記第２の予測モードでは、予測サンプルは、ブロックベクトルによって決定されるように、同じ復号ピクチャのサンプル値のブロックから導出される、
請求項１に記載の方法。
【請求項１０】
前記デフォルトモードは、平面モードである、
請求項１又は９に記載の方法。

【請求項１１】
前記変換は、前記現在のクロマブロックを前記ビットストリームに符号化することを含む、
請求項１から１０のいずれか１項に記載の方法。
【請求項１２】
前記変換は、前記ビットストリームから前記現在のクロマブロックを復号することを含む、
請求項１から１０のいずれか１項に記載の方法。
【請求項１３】
処理装置と、命令が記憶された非一時的メモリと、を備える映像データを処理する装置であって、
前記命令は、前記処理装置による実行時に、前記処理装置に、
映像の現在のクロマブロックと前記映像のビットストリームとの変換のために、前記現在のクロマブロックに対応する第１の輝度ブロックの輝度コーディング情報に基づいて、前記現在のクロマブロックのクロマ予測モードを導出することと、
導出された前記クロマ予測モードに基づいて前記変換を行うことと、を実行させ、
前記第１の輝度ブロックが、輝度予測モードで導出された量子化残差と前記量子化残差の予測値との差分が前記ビットストリームに含まれる差分コーディングモードでコーディングされることに応じて、
前記輝度予測モードは、垂直予測モード又は水平予測モードであり、
前記クロマ予測モードは、前記輝度予測モードと同一であり、
前記第１の輝度ブロックが第２の輝度ブロックの隣接ブロックである場合、前記第２の輝度ブロックと前記第１の輝度ブロックとが異なるスライス又はタイルにあることに応じて、前記第２の輝度ブロックのイントラモード候補リストを構築するために、デフォルトモードが前記第１の輝度ブロックに割り当てられる、
装置。
【請求項１４】
命令を記憶する非一時的なコンピュータ可読記憶媒体であって、
前記命令は、処理装置に、
映像の現在のクロマブロックと前記映像のビットストリームとの変換のために、前記現在のクロマブロックに対応する第１の輝度ブロックの輝度コーディング情報に基づいて、前記現在のクロマブロックのクロマ予測モードを導出することと、
導出された前記クロマ予測モードに基づいて前記変換を行うことと、を実行させ、
前記第１の輝度ブロックが、輝度予測モードで導出された量子化残差と前記量子化残差の予測値との差分が前記ビットストリームに含まれる差分コーディングモードでコーディングされることに応じて、
前記輝度予測モードは、垂直予測モード又は水平予測モードであり、
前記クロマ予測モードは、前記輝度予測モードと同一であり、
前記第１の輝度ブロックが第２の輝度ブロックの隣接ブロックである場合、前記第２の輝度ブロックと前記第１の輝度ブロックとが異なるスライス又はタイルにあることに応じて、前記第２の輝度ブロックのイントラモード候補リストを構築するために、デフォルトモードが前記第１の輝度ブロックに割り当てられる、
非一時的なコンピュータ可読記憶媒体。
【請求項１５】
映像のビットストリームを記憶する方法であって、
映像の現在のクロマブロックのために、前記現在のクロマブロックに対応する第１の輝度ブロックの輝度コーディング情報に基づいて、前記現在のクロマブロックのクロマ予測モードを導出することと、
導出された前記クロマ予測モードに基づいて前記ビットストリームを生成することと、
前記ビットストリームを非一時的なコンピュータ可読記録媒体に記憶することと、を含み、

前記第１の輝度ブロックが、輝度予測モードで導出された量子化残差と前記量子化残差の予測値との差分が前記ビットストリームに含まれる差分コーディングモードでコーディングされることに応じて、

前記輝度予測モードは、垂直予測モード又は水平予測モードであり、

前記クロマ予測モードは、前記輝度予測モードと同一であり、

前記第１の輝度ブロックが第２の輝度ブロックの隣接ブロックである場合、前記第２の輝度ブロックと前記第１の輝度ブロックとが異なるスライス又はタイルにあることに応じて、前記第２の輝度ブロックのイントラモード候補リストを構築するために、デフォルトモードが前記第１の輝度ブロックに割り当てられる、

方法。

【発明の詳細な説明】

【技術分野】

【０００１】

関連出願の相互参照

本願は、２０２１年１２月２８日に提出された日本特許出願２０２１－５７８１６５号に基づく分割出願であり、この日本特許出願は、２０２０年６月２８日出願の国際特許出願ＰＣＴ／ＵＳ２０２０／０４００２７号に基づくものであり、この国際特許出願は、２０１９年６月２８日出願の国際特許出願ＰＣＴ／ＣＮ２０１９／０９３８５２号、２０１９年６月２９日出願の国際特許出願ＰＣＴ／ＣＮ２０１９／０９３９８２号の優先権および利益を主張する。全ての前述の特許出願は、参照によりその全体が本明細書に組み込まれる。

【０００２】

本明細書は、映像および画像符号化、復号化技術に関する。

【背景技術】

【０００３】

デジタル映像は、インターネット及び他のデジタル通信ネットワークにおいて最大の帯域幅の使用量を占めている。映像を受信及び表示することが可能である接続されたユーザ機器の数が増加するにつれ、デジタル映像の使用に対する帯域幅需要は増大し続けることが予測される。

【発明の概要】

【０００４】

開示された技術は、映像を符号化または復号化するためにパレット符号化モードを用いる映像または画像デコーダまたはエンコーダの実施形態によって使用してもよい。

【０００５】

１つの例示的な態様において、映像復号化方法が開示される。この方法は、クロマブロックであり、クロマＤＭモードで符号化された映像の現在の映像ブロックに対して、前記現在の映像ブロックの符号化モードに関する規則に従って、前記現在の映像ブロックのためのイントラ予測モードを導出することと、現在の映像ブロックと前記映像の符号化表現との変換を行うことと、を含み、前記規則は、クロマＤＭモードを導出するために使用され、現在のビデオブロックに対応する量子化残差差分パルス符号変調表現を用いて符号化された対応する輝度ブロックのイントラ予測方向に基づいてイントラ予測モードを導出することを規定する。

【０００６】

別の例示的な態様において、別の映像処理方法が開示される。この方法は、クロマブロックである映像の現在の映像ブロックと前記映像の符号化表現との変換を行うことを含み、現在の映像ブロックの対応する輝度映像ブロックが非イントラモードで符号化されるため、現在の映像ブロックのための予測ブロックを生成するためのイントラ予測方法が使用される。

【０００７】

さらに別の例示的な態様において、別の映像処理方法が開示される。この方法は、映像

のピクチャの現在の映像ブロックと前記映像の符号化表現とを変換するために、現在の映像ブロックの特性に基づいて、前記現在の映像ブロックに対してイントラブロックコピー（ＩＢＣ）モードを使用することを示す構文要素のための１つ以上のコンテクストを構成することと、前記構成に基づいて前記変換を行うことと、を含み、前記ＩＢＣモードを使用することは、前記ピクチャにおける別の映像ブロックの予め符号化されたサンプルを使用して前記現在の映像ブロックのサンプル値を予測することを含む。

【０００８】

さらに別の例示的な態様において、別の映像処理方法が開示される。この方法は、映像の現在の映像ブロックと前記映像の符号化表現との変換のために、前記現在の映像ブロックの近傍のブロックに関連付けられた第２の構文要素に基づいて、第１の構文要素に関連付けられたビンを符号化するためのコンテクストモデルを構成することと、前記構成に基づいて前記変換を行うことと、を含む。

【０００９】

さらに別の例示的な態様において、別の映像処理方法が開示される。この方法は、映像の現在の映像ブロックとこの映像の符号化表現との変換を行うことを含み、ここで、現在の映像ブロックに対応する量子化残差の差分パルス符号変調表現を使用して、この符号化表現が、この符号化表現において表されるこの現在の映像ブロックのための非ゼロ係数を含むかどうかを示すために、１つのフラグを決定する。

【００１０】

さらに別の例示的な態様において、別の映像処理方法が開示される。この方法は、クロマブロックである映像の現在の映像ブロックとこの映像の符号化表現との変換を行うことを含み、現在の映像ブロックのためのクロマＤＭモードは、クロマＤＭモードを導出するために使用されイントラモードで符号化されない対応する輝度ブロックに起因して無効化される。

【００１１】

さらに別の例示的な態様において、別の映像処理方法が開示される。この方法は、規則に従って、クロマブロックである映像の現在の映像ブロックとこの映像の符号化表現とを変換することを含み、現在の映像ブロックは、対応する輝度ブロックの符号化された情報から予測ブロックのためのモードを導出する導出モードを使用して計算された予測ブロックを使用して、符号化表現で表現され、この規則は、導出されたモードを使用することによって、現在の映像ブロックが無効なブロックベクトルを有すると仮定して、予測ブロックを決定することを指定する。

【００１２】

さらに別の例示的な態様において、別の映像処理方法が開示される。この方法は、映像の現在の映像ブロックとこの映像の符号化表現との変換のために、規則に従って現在の映像ブロックのデフォルトのイントラモードを決定することと、前記デフォルトのイントラモードを使用して前記変換を実行することと、を含む。

【００１３】

さらに別の例示的な態様において、別の映像処理方法が開示される。この方法は、最大確率モード（ＭＰＭ）リストを構築する間、サンプル値を量子化するパルス符号化変調（ＰＣＭ）モードで符号化された近傍のブロックに起因して、映像の現在の映像ブロックを直流（ＤＣ）予測モードで符号化することを決定することと、現在の映像ブロックと映像の符号化表現との変換を行うことと、を含む。

【００１４】

さらに別の例示的な態様において、映像処理の別の方法が開示される。この方法は、現在の映像ブロックの現在の映像ブロックとこの映像の符号化表現との変換のために、この現在の映像ブロックの１つ以上の近傍のブロックの符号化された情報に基づいて、三角形分割モード（ＴＰＭ）および／または動き記憶処理を使用する処理の適用可能性を決定することと、前記決定に基づいて前記変換を行うことと、を含み、前記ＴＰＭを使用することは、１つの映像ブロックを、少なくとも１つの非長方形パーティションを含むサブパー

ティションに分割することを含む。

【００１５】

別の例示的な態様において、上述された方法は、処理装置を含む映像デコーダによって実装されてもよい。

【００１６】

別の例示的な態様において、上述された方法は、処理装置を含む映像エンコーダによって実装されてもよい。

【００１７】

さらに別の例示的な態様において、これらの方法は、処理装置実行可能命令の形式で実施されてもよく、コンピュータ可読プログラム媒体に記憶されてもよい。

【００１８】

これらの、および他の態様は、本明細書でさらに説明される。

【図面の簡単な説明】

【００１９】

【図１】イントラブロックコピーの説明図を示す。

【図２】空間的に近傍の候補の例を示す。

【図３】パレットモードで符号化されたブロックの例を示す。

【図４】パレットエントリを信号通知するためのパレット予測子の使用例を示す。

【図５】水平方向および垂直方向の横断走査の例を示す。

【図６】パレットインデックスの符号化の例を示す。

【図７】マルチタイプのツリー分割モードの例を示す。

【図８】フィルタのオン／オフ決定および強／弱フィルタの選択に関与する画素の例を示す。

【図９】イントラ予測を組み合わせたパレットモードの方法の実装の例を示す。

【図１０】複合パレットモードの例示的な実装の例を示す。

【図１１】６７個のイントラ予測モード（方向）の例を示す。

【図１２】現在の映像ブロックの左上の近傍の例を示す。

【図１３】クロマブロックに対応する輝度範囲を例示している。

【図１４】三角形パーティションに基づくインター予測の例を示す。

【図１５】単一予測候補リストを構築するために用いられる空間的及び時間的な近傍のブロックの例を示す。

【図１６】ブレンド処理に用いられる重みの例を示す。

【図１７Ａ】映像処理装置の例を示すブロック図である。

【図１７Ｂ】映像処理装置の例を示すブロック図である。

【図１８Ａ】開示される技術のいくつかの実装形態に基づくビデオ処理の例を示すフローチャートである。

【図１８Ｂ】開示される技術のいくつかの実装形態に基づくビデオ処理の例を示すフローチャートである。

【図１８Ｃ】開示される技術のいくつかの実装形態に基づくビデオ処理の例を示すフローチャートである。

【図１８Ｄ】開示される技術のいくつかの実装形態に基づくビデオ処理の例を示すフローチャートである。

【図１８Ｅ】開示される技術のいくつかの実装形態に基づくビデオ処理の例を示すフローチャートである。

【図１８Ｆ】開示される技術のいくつかの実装形態に基づくビデオ処理の例を示すフローチャートである。

【図１８Ｇ】開示される技術のいくつかの実装形態に基づくビデオ処理の例を示すフローチャートである。

【発明を実施するための形態】

【００２０】

本明細書は、伸張または復号化されたデジタル映像または画像の品質を向上させるために、画像または映像ビットストリームのデコーダによって使用できる様々な技術を提供する。簡潔にするために、本明細書では、用語「映像」は、一連のピクチャ（従来から映像と呼ばれる）および個々の画像の両方を含むように使用される。さらに、映像エンコーダは、さらなる符号化に使用される復号化されたフレームを再構成するために、符号化の処理中にこれらの技術を実装してもよい。

【００２１】

本明細書では、理解を容易にするために章の見出しを使用しており、１つの章に開示された実施形態をその章にのみ限定するものではない。このように、ある章の実施形態は、他の章の実施形態と組み合わせることができる。

【００２２】

１．　発明の概要

【００２３】

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

【００２４】

２．　初期の協議

【００２５】

映像符号化規格は、主に周知のＩＴＵ－ＴおよびＩＳＯ／ＩＥＣ規格の開発によって発展してきた。ＩＴＵ－ＴはＨ．２６１とＨ．２６３を作り、ＩＳＯ／ＩＥＣはＭＰＥＧ－１とＭＰＥＧ－４　Ｖｉｓｕａｌを作り、両団体はＨ．２６２／ＭＰＥＧ－２　ＶｉｄｅｏとＨ．２６４／ＭＰＥＧ－４　ＡＶＣ（Ａｄｖａｎｃｅｄ　Ｖｉｄｅｏ　Ｃｏｄｉｎｇ）とＨ．２６５／ＨＥＶＣ規格を共同で作った。Ｈ．２６２以来、映像符号化規格は、時間予測と変換符号化が利用されるハイブリッド映像符号化構造に基づく。ＨＥＶＣを超えた将来の映像符号化技術を探索するため、２０１５年には、ＶＣＥＧとＭＰＥＧが共同でＪＶＥＴ（Ｊｏｉｎｔ　Ｖｉｄｅｏ　Ｅｘｐｌｏｒａｔｉｏｎ　Ｔｅａｍ）を設立した。それ以来、多くの新しい方法がＪＶＥＴによって採用され、ＪＥＭ（Ｊｏｉｎｔ　Ｅｘｐｌｏｒａｔｉｏｎ　Ｍｏｄｅ）［３，４］と呼ばれる参照ソフトウェアに組み込まれてきた。２０１８年４月には、ＶＣＥＧ（Ｑ６／１６）とＩＳＯ／ＩＥＣ　ＪＴＣ１　ＳＣ２９／ＷＧ１１（ＭＰＥＧ）の間にＪｏｉｎｔ　Ｖｉｄｅｏ　Ｅｘｐｅｒｔ　Ｔｅａｍ（ＪＶＥＴ）が発足し、ＨＥＶＣと比較して５０％のビットレート削減を目標にＶＶＣ規格の策定に取り組んでいる。

【００２６】

ＶＶＣドラフトの最新バージョン、即ち、Ｖｅｒｓａｔｉｌｅ　Ｖｉｄｅｏ　Ｃｏｄｉｎｇ（ドラフト４）は、以下を参照することができる。

【００２７】

ｐｈｅｎｉｘ．ｉｔ－ｓｕｄｐａｒｉｓ．ｅｕ／ｊｖｅｔ／ｄｏｃ＿ｅｎｄ＿ｕｓｅｒ／ｃｕｒｒｅｎｔ＿ｄｏｃｕｍｅｎｔ．ｐｈｐ？ｉｄ＝５７５５

【００２８】

ＶＴＭと呼ばれるＶＶＣの最新の参照ソフトウェアは、以下で確認することができる。

【００２９】

ｖｃｇｉｔ．ｈｈｉ．ｆｒａｕｎｈｏｆｅｒ．ｄｅ／ｊｖｅｔ／ＶＶＣＳｏｆｔｗａｒｅ＿ＶＴＭ／ｔａｇｓ／ＶＴＭ－４．０

【００３０】

２．１　イントラブロックコピー

【００３１】

イントラブロックコピー（ＩＢＣ）、別名、現在のピクチャの参照（ＣＰＲ）は、ＨＥＶＣスクリーンコンテンツ符号化拡張機能（ＨＥＶＣ－ＳＣＣ）と現在のＶＶＣテストモ

デル（ＶＴＭ－４．０）に採用されている。ＩＢＣは、動き補償の概念をインターフレーム符号化からイントラフレーム符号化に拡張する。図１に示すように、現在のブロックは、ＩＢＣが適用される場合、同じピクチャ内の１つの参照ブロックによって予測される。現在のブロックを符号化または復号化する前に、参照ブロックにおけるサンプルは既に再構成されていなければならない。ＩＢＣは、カメラでキャプチャされたほとんどのシーケンスに対してそれほど効率的ではないが、スクリーンコンテンツに対しては、有意な符号化利得を示す。その理由は、スクリーンコンテンツピクチャにおいて、アイコン、文字等の繰り返しパターンが多いためである。ＩＢＣは、これらの繰り返しパターン間の冗長性を有効に除去することができる。ＨＥＶＣ－ＳＣＣにおいて、インター符号化ユニット（ＣＵ）は、現在のピクチャをその参照ピクチャとして選択する場合、ＩＢＣを適用することができる。この場合、動きベクトル（ＭＶ）をブロックベクトル（ＢＶ）と改称し、ＢＶは常に整数画素精度を有する。メインプロファイルＨＥＶＣに適合するように、現在のピクチャは、復号化ピクチャバッファ（ＤＰＢ）における「長期」参照ピクチャとしてマークされる。なお、同様に、複数のビュー／３Ｄ映像符号化規格において、ビュー間の参照ピクチャも「長期」参照ピクチャとしてマークされる。

【００３２】

ＢＶがその参照ブロックを見つけた後、この参照ブロックをコピーすることで予測を生成することができる。残差は、元の信号から参照画素を減算することによって得ることができる。そして、他の符号化モードと同様に、変換および量子化を適用することができる。

【００３３】

図１は、イントラブロックコピーの説明図を示す。

【００３４】

しかしながら、参照ブロックがピクチャの外にある場合、または現在のブロックと重複する場合、または再構成された領域の外にある場合、或いは何らかの制約によって制限された有効領域の外にある場合、画素値の一部または全部は規定されない。基本的に、このような問題に対処するために２つの解決策がある。１つは、このような状況、例えばビットストリーム適合性を許可しないことである。もう１つは、これらの未定義の画素値にパディングを適用することである。以下のサブセッションでは、解決策を詳細に説明する。

【００３５】

２．２　ＨＥＶＣスクリーンコンテンツ符号化拡張機能におけるＩＢＣ

【００３６】

ＨＥＶＣのスクリーンコンテンツ符号化拡張機能において、１つのブロックが現在のピクチャを参照として使用する場合、以下の仕様のテキストに示すように、参照ブロック全体が利用可能な再構成された領域内にあることを保証すべきである。

【００３７】

変数ｏｆｆｓｅｔＸおよびｏｆｆｓｅｔＹは、以下のように導出される。

ｏｆｆｓｅｔＸ＝（ＣｈｒｏｍａＡｒｒａｙＴｙｐｅ＝＝０）？０：（ｍｖＣＬＸ［０］＆０×７？２：０）　　（２－１）

ｏｆｆｓｅｔＹ＝（ＣｈｒｏｍａＡｒｒａｙＴｙｐｅ＝＝０）？０：（ｍｖＣＬＸ［１］＆０×７？２：０）　　（２－２）

【００３８】

参照ピクチャが現在のピクチャである場合、輝度動きベクトルｍｖＬＸは、以下の制約に従うべきであることが、ビットストリーム適合性の要件である。

－６．４．１項で規定されたようなｚ走査順序ブロックの可用性に対する導出処理が、（ｘＣｂ，ｙＣｂ）と等しく設定された（ｘＣｕｒｒ，ｙＣｕｒｒ）と、（ｘＰｂ＋（ｍｖＬＸ［０］＞＞２）－ｏｆｆｓｅｔＸ，ｙＰｂ＋（ｍｖＬＸ［１］＞＞２）－ｏｆｆｓｅｔＹ）に等しく設定された近傍の輝度位置（ｘＮｂＹ，ｙＮｂＹ）と、が入力として呼び出されると、出力はＴＲＵＥとなる。

－６．４．１項で規定されたようなｚ走査順序ブロックの可用性に対する導出処理が、（

ｘＣｂ．ｙＣｂ）と等しく設定された（ｘＣｕｒｒ，ｙＣｕｒｒ）、（ｘＰｂ＋（ｍｖＬＸ［０］＞＞２）＋ｎＰｂＷ－１＋ｏｆｆｓｅｔＸ，ｙＰｂ＋（ｍｖＬＸ［１］＞＞２）＋ｎＰｂＨ－１＋ｏｆｆｓｅｔＹ）に等しく設定された近傍の輝度位置（ｘＮｂＹ，ｙＮｂＹ）を入力として呼び出されると、出力はＴＲＵＥとなる。
－以下の条件の一方または両方がＴＲＵＥであること。
　－（ｍｖＬＸ［０］＞＞２）＋ｎＰｂＷ＋ｘＢ１＋ｏｆｆｓｅｔＸの値が０以下である。
　－（ｍｖＬＸ［１］＞＞２）＋ｎＰｂＨ＋ｙＢ１＋ｏｆｆｓｅｔＹの値が０以下である。
－以下の条件がＴＲＵＥとなること。
（ｘＰｂ＋（ｍｖＬＸ［０］＞＞２）＋ｎＰｂＳｗ－１＋ｏｆｆｓｅｔＸ）／ＣｔｂＳｉｚｅＹ－ｘＣｕｒｒ／ＣｔｂＳｉｚｅＹ＜＝
ｙＣｕｒｒ／ＣｔｂＳｉｚｅＹ－（ｙＰｂ＋（ｍｖＬＸ［１］＞＞２）＋ｎＰｂＳｈ－１＋ｏｆｆｓｅｔＹ）／ＣｔｂＳｉｚｅＹ　　（２－３）

【００３９】
　このように、参照ブロックが現在のブロックと重複するケース、または参照ブロックがピクチャの外にあるケースは発生しない。参照ブロックまたは予測ブロックを埋める必要がない。

【００４０】
２．３．　ＶＶＣ試験モデルにおけるＩＢＣ

【００４１】
　現在のＶＶＣ試験モデル、すなわち、ＶＴＭ－４．０設計において、参照ブロック全体は現在の符号化ツリーユニット（ＣＴＵ）を有するべきであり、現在のブロックと重複しない。よって、参照または予測ブロックをパディングする必要がない。ＩＢＣフラグは、現在のＣＵの予測モードとして符号化される。このように、各ＣＵに対して、ＭＯＤＥ＿ＩＮＴＲＡ、ＭＯＤＥ＿ＩＮＴＥＲ、およびＭＯＤＥ＿ＩＢＣという全部で３つの予測モードが存在する。

【００４２】
２．３．１　ＩＢＣマージモード

【００４３】
　ＩＢＣマージモードにおいて、ＩＢＣマージ候補リストにおけるエントリを指すインデックスをビットストリームから構文解析する。このＩＢＣマージリストの構築は、以下のステップのシーケンスに従ってまとめることができる。
●　ステップ１：空間的候補の導出
●　ステップ２：ＨＭＶＰ候補の挿入
●　ステップ３：対の平均候補の挿入

【００４４】
　空間的マージ候補の導出において、位置A_1、B_1、B_0、A_0およびB_2にある候補の中から、最大４つのマージ候補を選択する。導出の順序はA_1、B_1、B_0、A_0、B_2である。位置A_1、B_1、B_0、A_0のいずれかのＰＵが利用可能でない場合（例えば、別のスライスまたはタイルに属しているため）、またはＩＢＣモードで符号化されていない場合にのみ、位置B_2が考慮される。位置A_1の候補を加えた後、残りの候補を挿入すると、冗長性チェックを受け、それにより、同じ動き情報を有する候補を確実にリストから排除でき、符号化効率を向上させることができる。

【００４５】
　空間的候補を挿入した後、ＩＢＣマージリストサイズが依然として最大ＩＢＣマージリストサイズより小さい場合、ＨＭＶＰテーブルからのＩＢＣ候補を挿入することができる。ＨＭＶＰ候補の挿入にあたり、冗長性チェックを行う。

【００４６】
　最後に、対の平均候補をＩＢＣマージリストに挿入する。

【００４７】
　マージ候補によって特定される参照ブロックがピクチャの外にある場合、または現在のブロックと重複する場合、または再構成された領域の外にある場合、或いは何らかの制約によって制限された有効領域の外にある場合、マージ候補は無効なマージ候補と呼ばれる。
【００４８】
　なお、ＩＢＣマージリストに無効なマージ候補を挿入してもよい。
【００４９】
　ＶＶＣにはＪＶＥＴ－Ｎ０８４３が採用されている。ＪＶＥＴ－Ｎ０８４３では、ＩＢＣのマージモードとＡＭＶＰモードのＢＶ予測子は、次の要素で構成される共通の予測子リストを共有する。
●　２つの空間的近傍位置（図２のようにＡ１、Ｂ１）
●　５　ＨＭＶＰのエントリ
●　デフォルトでゼロベクトル
【００５０】
　マージモードの場合、このリストの最初の６つまでのエントリが使用され、ＡＭＶＰモードの場合、このリストの最初の２つのエントリが使用される。また、リストは共有マージリスト領域の要件に準拠している（ＳＭＲ内で同じリストを共有）。
【００５１】
　上述のＢＶ予測子候補リストに加え、ＪＶＥＴ－Ｎ０８４３は、ＨＭＶＰ候補と既存のマージ候補（Ａ１，Ｂ１）との間のプルーニング操作を簡素化することも提案している。簡素化した場合、第１のＨＭＶＰ候補と空間的マージ候補とを比較するだけであるため、最大２回までのプルーニング操作でよい。
【００５２】
　最近のＶＶＣ及びＶＴＭ５において、前回のＶＴＭ及びＶＶＣバージョンにおいて、現在のビットストリーム制約に加え、１２８×１２８のＩＢＣモードを無効化するための構文制約を明示的に使用することが提案され、これにより、ＩＢＣフラグの存在がＣＵのサイズ＜１２８×１２８に依存するようになる。
【００５３】
２．３．２　ＩＢＣ　ＡＭＶＰモード
【００５４】
　ＩＢＣ　ＡＭＶＰモードでは、ＩＢＣ　ＡＭＶＰリストにおけるエントリを指すＡＭＶＰインデックスが、ビットストリームから構文解析される。このＩＢＣ　ＡＭＶＰリストの構築は、以下のステップのシーケンスに従ってまとめることができる。
●　ステップ１：空間的候補の導出
　○利用可能な候補が見つかるまで、A_0、A_1をチェックする。
　○利用可能な候補が見つかるまで、B_0、B_1、B_2をチェックする。
●　ステップ２：ＨＭＶＰ候補の挿入
●　ステップ３：ゼロ候補の挿入
【００５５】
　空間的候補を挿入した後、ＩＢＣ　ＡＭＶＰリストサイズが依然として最大ＩＢＣ　ＡＭＶＰリストサイズより小さい場合、ＨＭＶＰテーブルからのＩＢＣ候補を挿入することができる。
【００５６】
　最後に、ＩＢＣ　ＡＭＶＰリストにゼロ候補を挿入する。
【００５７】
２．３．３　クロマＩＢＣモード
【００５８】
　現在のＶＶＣにおいて、クロマＩＢＣモードにおける動き補償は、サブブロックレベルで行われる。クロマブロックは、複数のサブブロックに分割される。各サブブロックは、

対応する輝度ブロックが１つのブロックベクトルを有するかどうかを判定し、存在する場合、有効性を判定する。現在のＶＴＭにはエンコーダ制約があり、現在のクロマＣＵにおけるすべてのサブブロックが有効な輝度ブロックベクトルを有するかどうかについて、クロマＩＢＣモードをテストする。例えば、ＹＵＶ４２０映像において、クロマブロックはＮ×Ｍであり、そして、コロケーション輝度領域は２Ｎ×２Ｍである。クロマブロックのサブブロックサイズは２×２である。クロマｍｖ導出、次にブロックコピー処理を実行するには、いくつかのステップがある。

１）　クロマブロックは、まず、（Ｎ＞＞１）＊（Ｍ＞＞１）個のサブブロックに分割される。

２）　左上のサンプルが（ｘ，ｙ）に配置されている各サブブロックは、（２ｘ，２ｙ）に配置されている同じ左上のサンプルを含んだ対応する輝度ブロックをフェッチする。

３）　エンコーダは、フェッチした輝度ブロックのブロックベクトル（ｂｖ）をチェックする。以下の条件の１つを満たす場合、ｂｖは無効であると見なされる。

　ａ．対応する輝度ブロックのｂｖは存在しない。

　ｂ．ｂｖによって識別される予測ブロックは、まだ再構成されていない。

　ｃ．ｂｖで識別される予測ブロックは、現在のブロックと部分的にまたは完全に重複している。

４）　サブブロックのクロマ動きベクトルは、対応する輝度サブブロックの動きベクトルに設定される。

【００５９】

　すべてのサブブロックが有効なｂｖを見つけた場合、エンコーダにおいてＩＢＣモードが許可される。

【００６０】

　ＩＢＣブロックの復号処理を以下に示す。ＩＢＣモードでのクロマｍｖの導出に関する部分は、グレーで強調表示されている。

【００６１】

８．６．１　ＩＢＣ予測において符号化されたユニットを符号化するための一般的な復号処理

【００６２】

　この処理への入力は以下の通りである。

－現在のピクチャの左上の輝度サンプルに対する現在の符号化ブロックの左上のサンプルを規定する輝度位置（ｘＣｂ，ｙＣｂ）、

－輝度サンプルにおける現在の符号化ブロックの幅を規定する変数ｃｂＷｉｄｔｈ、

－輝度サンプルにおける現在の符号化ブロックの高さを規定する変数ｃｂＨｅｉｇｈｔ、

－単一ツリーを使用するか二重ツリーを使用するかを指定する変数ツリータイプ、および二重ツリーを使用する場合、現在のツリーが輝度成分に対応するか色度成分に対応するかを指定する。

【００６３】

　この処理の出力は、インループ・フィルタリング前の修正された再構成画像である。

【００６４】

　輝度位置（ｘＣｂ，ｙＣｂ）、輝度サンプルｃｂＷｉｄｔｈにおける現在の符号化ブロックの幅、輝度サンプルｃｂＨｅｉｇｈｔにおける現在の符号化ブロックの高さ、及び変数ツリータイプを入力として、８．７．１項に規定される量子化パラメータの導出処理を呼び出す。

【００６５】

　ｉｂｃ予測モードで符号化されたユニットを符号化するための復号プロセスは、以下の順序付けられたステップからなる。

【００６６】

１．現在の符号化ユニットの動きベクトル成分は、以下のように導出される。

１．ツリータイプがＳＩＮＧＬＥ＿ＴＲＥＥまたはＤＵＡＬ＿ＴＲＥＥ＿ＬＵＭＡに等し

い場合、以下が適用される。
－輝度符号化ブロックの位置（ｘＣｂ，ｙＣｂ）、輝度符号化ブロックの幅ｃｂＷｉｄｔｈ、輝度符号化ブロックをと、高さｃｂＨｅｉｇｈｔを入力とし、輝度動きベクトルｍｖＬ［０］［０］を出力として、項目８．６．２．１に規定された動きベクトル成分の導出処理が呼び出される。
－ツリータイプがＳＩＮＧＬＥ＿ＴＲＥＥに等しい場合、８．６．２．９項のクロマ動きベクトルの導出処理が、入力として輝度動きベクトルｍｖＬ［０］［０］、出力としてクロマ動きベクトルｍｖＣ［０］［０］を用いて実行される。
－水平方向ｎｕｍＳｂＸ及び垂直方向ｎｕｍＳｂＹにおける輝度符号化サブブロックの数は、いずれも１に設定される。
１．そうでない場合、もしツリータイプがＤＵＡＬ＿ＴＲＥＥ＿ＣＨＲＯＭＡに等しい場合、以下が適用される。
－水平方向ｎｕｍＳｂＸ及び垂直方向ｎｕｍＳｂＹにおけるＬＵＭＡ符号化サブブロックの数は、次のように導出される。
ｎｕｍＳｂＸ＝（ｃｂＷｉｄｔｈ＞＞２）　（８－８８６）
ｎｕｍＳｂＹ＝（ｃｂＨｅｉｇｈｔ＞＞２）　（８－８８７）
－クロマ動きベクトルｍｖＣ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］は、ｘＳｂＩｄｘ＝０．．ｎｕｍＳｂＸ－１，ｙＳｂＩｄｘ＝０．．ｎｕｍＳｂＹ－１の場合、次のように導出される。：
－輝度動きベクトルｍｖＬ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］は、以下のように導出される。
－並置されたＬＵＭＡ符号化ユニットの位置（ｘＣｕＹ，ｙＣｕＹ）は、次のように導出される。
ｘＣｕＹ＝ｘＣｂ＋ｘＳｂＩｄｘ＊４　（８－８８８）
ｙＣｕＹ＝ｙＣｂ＋ｙＳｂＩｄｘ＊４　（８－８８９）
－ＣｕＰｒｅｄＭｏｄｅ［ｘＣｕＹ］［ｙＣｕＹ］がＭＯＤＥ＿ＩＮＴＲＡに等しい場合、次のようになる。
ｍｖＬ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］［０］＝０　（８－８９０）
ｍｖＬ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］［１］＝０　（８－８９１）
ｐｒｅｄＦｌａｇＬ０［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］＝０　（８－８９２）
－　ｐｒｅｄＦｌａｇＬ１［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］＝０　（８－８９３）
－その他の場合（ＣｕＰｒｅｄＭｏｄｅ［ｘＣｕＹ］［ｙＣｕＹ］がＭＯＤＥ＿ＩＢＣに等しい）、次のことが該当する。
－　ｍｖＬ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］［０］＝ＭｖＬ０［ｘＣｕＹ］［ｙＣｕＹ］［０］　（８－８９４）
－　ｍｖＬ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］［１］＝ＭｖＬ０［ｘＣｕＹ］［ｙＣｕＹ］［１］　（８－８９５）
－　ｐｒｅｄＦｌａｇＬ０［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］＝１　（８－８９６）
－　ｐｒｅｄＦｌａｇＬ１［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］＝０　（８－８９７）
－８．６．２．９項のクロマ動きベクトルの導出処理は、ｍｖＬ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］を入力とし、ｍｖＣ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］を出力として呼び出される。
－クロマ動きベクトルｍｖＣ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］が以下の制約に従うことは、ビットストリーム適合性の要件である。
－　６．４．Ｘ項で規定されているブロック利用可能性の導出プロセス［ＥＤ．（ＢＢ）：隣接ブロック利用可能性チェックプロセスＴＢＤ］が、現在のクロマ位置（ｘＣｕｒｒ，ｙＣｕｒｒ）を（ｘＣｂ／ＳｕｂＷｉｄｔｈＣ，ｙＣｂ／ＳｕｂＨｅｉｇｈｔＣ）に等しく、隣接するクロ位置（ｘＣｂ／ＳｕｂＷｉｄｔｈＣ＋（ｍｖＣ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］［０］＞＞５），ｙＣｂ／ＳｕｂＨｅｉｇｈｔＣ＋（ｍｖＣ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］［１］＞＞５））を入力として呼び出されると、出力はＴＲＵ

Ｅとなる。

－　６．４．Ｘ項で規定されているブロック利用可能性の導出プロセス［ＥＤ．（ＢＢ）：隣接ブロック利用可能性チェックプロセスＴＢＤ］が、現在のクロマ位置（ｘＣｕｒｒ，ｙＣｕｒｒ）を（ｘＣｂ／ＳｕｂＷｉｄｔｈＣ，ｙＣｂ／ＳｕｂＨｅｉｇｈｔＣ）に等しく、隣接するクロ位置（ｘＣｂ／ＳｕｂＷｉｄｔｈＣ＋（ｍｖＣ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］［０］＞＞５）＋ｃｂＷｉｄｔｈ／ＳｕｂＷｉｄｔｈＣ－１，ｙＣｂ／ＳｕｂＨｅｉｇｈｔＣ＋（ｍｖＣ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］［１］＞＞５）＋ｃｂＨｅｉｇｈｔ／ＳｕｂＨｅｉｇｈｔＣ－１）を入力として呼び出されると、出力はＴＲＵＥとなる。

－次の条件の一方または両方がｔｒｕｅであること。

－　（ｍｖＣ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］［０］＞＞５）＋ｘＳｂＩｄｘ＊２＋２　は、０より小さい又は等しい。

－　（ｍｖＣ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］［１］＞＞５）＋ｙＳｂＩｄｘ＊２＋２　は、０より小さい又は等しい。

【００６７】

２．現在の符号化ユニットの予測サンプルは、以下のように導出される。

－ツリータイプがＳＩＮＧＬＥ＿ＴＲＥＥまたはＤＵＡＬ＿ＴＲＥＥ＿ＬＵＭＡに等しい場合、現在のコーディングユニットの予測サンプルは以下のように導出される。

●　８．６．３．１項に規定されるｉｂｃブロックの復号処理は、輝度符号化ブロックの位置（ｘＣｂ，ｙＣｂ）、輝度符号化ブロックの幅ｃｂＷｉｄｔｈ、輝度符号化ブロックの高さｃｂＨｅｉｇｈｔ、水平方向ｎｕｍＳｂＸ、垂直方向ｎｕｍＳｂＹにおける輝度符号化サブブロックの数、輝度動きベクトルｍｖＬ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］ｘＳｂＩｄｘ＝０．．ｎｕｍＳｂＸ－１、およびＳｂＩｄｘ＝０．．ｎｕｍＳｂＹ－１、および変数ｃＩｄｘを０に等しく入力として設定し、予測輝度サンプルの（ｃｂＷｉｄｔｈ）×（ｃｂＨｅｉｇｈｔ）配列ｐｒｅｄＳａｍｐｌｅｓＬであるｉｂｃ予測サンプル（ｐｒｅｄＳａｍｐｌｅｓ）を出力として呼び出される。

－あるいは、ツリータイプがＳＩＮＧＬＥ＿ＴＲＥＥまたはＤＵＡＬ＿ＴＲＥＥ＿ＣＨＲＯＭＡに等しい場合、現在の符号化ユニットの予測サンプルは、以下のように導出される。

●　８．６．３．１項に規定されるｉｂｃブロックの復号処理は、輝度符号化ブロックの位置（ｘＣｂ，ｙＣｂ）、輝度符号化ブロックの幅ｃｂＷｉｄｔｈ、輝度符号化ブロックの高さｃｂＨｅｉｇｈｔ、水平方向ｎｕｍＳｂＸ、垂直方向ｎｕｍＳｂＹにおける輝度符号化サブブロックの数、ｘＳｂＩｄｘ＝０．．ｎｕｍＳｂＸ－１、ｙＳｂＩｄｘ＝０．．ｎｕｍＳｂＹ－１を有するクロマ動きベクトルｍｖＣ［ｘＳｂＩｄｘ］、ｙＳｂＩｄｘ＝０．．ｎｕｍＳｂＹ－１、および変数ｃＩｄｘを１に等しく入力として設定し、ｉｂｃ予測サンプル（ｐｒｅｄＳａｍｐｌｅｓ）は、クロマ成分Ｃｂに対する予測クロマサンプルの（ｃｂＷｉｄｔｈ／２）×（ｃｂＨｅｉｇｈｔ／２）配列ｐｒｅｄＳａｍｐｌｅｓ$_{Cb}$であるｉｂｃ予測サンプル（ｐｒｅｄＳａｍｐｌｅｓ）を出力として呼び出される。

●８．６．３．１項に規定されるｉｂｃブロックの復号処理は、輝度符号化ブロックの位置（ｘＣｂ，ｙＣｂ）、輝度符号化ブロックの幅ｃｂＷｉｄｔｈ、輝度符号化ブロックの高さｃｂＨｅｉｇｈｔ、水平方向ｎｕｍＳｂＸ、垂直方向ｎｕｍＳｂＹにおける輝度符号化サブブロックの数、ｘＳｂＩｄｘ＝０．．ｎｕｍＳｂＸ－１、ｙＳｂＩｄｘ＝０．．ｎｕｍＳｂＹ－１を有するクロマ動きベクトルｍｖＣ［ｘＳｂＩｄｘ］、ｙＳｂＩｄｘ＝０．．ｎｕｍＳｂＹ－１、および変数ｃＩｄｘを２に等しく入力として設定し、ｉｂｃ予測サンプル（ｐｒｅｄＳａｍｐｌｅｓ）は、クロマ成分Ｃｒに対する予測クロマサンプルの（ｃｂＷｉｄｔｈ／２）×（ｃｂＨｅｉｇｈｔ／２）配列ｐｒｅｄＳａｍｐｌｅｓ$_{Cr}$であるｉｂｃ予測サンプル（ｐｒｅｄＳａｍｐｌｅｓ）を出力として呼び出される。

【００６８】

３．変数ＮｕｍＳｂＸ［ｘＣｂ］［ｙＣｂ］およびＮｕｍＳｂＹ［ｘＣｂ］［ｙＣｂ］は、それぞれｎｕｍＳｂＸおよびｎｕｍＳｂＹに等しく設定される。

【００６９】
４．現在の符号化ユニットの残差サンプルは、以下のように導出される。
　－ツリータイプがＳＩＮＧＬＥ＿ＴＲＥＥに等しい場合またはツリータイプがＤＵＡＬ＿ＴＲＥＥ＿ＬＵＭＡに等しい場合、８．５．８項に規定されるインター予測モードで符号化された符号化ブロックの残差信号の復号処理は、位置（ｘＴｂ０，ｙＴｂ０）を輝度位置（ｘＣｂ，ｙＣｂ）、幅ｎＴｂＷを輝度符号化ブロック幅ｃｂＷｉｄｔｈ、高さｎＴｂＨを輝度符号化ブロック高さｃｂＨｅｉｇｈｔに等しく、変数ｃｌｄｘｓｅｔを０に等しく入力として設定し、配列ｒｅｓＳａｍｐｌｅｓ$_{L}$を出力として、呼び出される。
　－ツリータイプがＳＩＮＧＬＥ＿ＴＲＥＥに等しい場合またはツリータイプがＤＵＡＬ＿ＴＲＥＥ＿ＣＨＲＯＭＡに等しい場合、８．５．８項に規定されるインター予測モードで符号化された符号化ブロックの残差信号の復号処理は、位置（ｘＴｂ０，ｙＴｂ０）をクロマ位置（ｘＣｂ／２，ｙＣｂ／２）、幅ｎＴｂＷをクロマ符号化ブロック幅ｃｂＷｉｄｔｈ／２、高さｎＴｂＨをクロマ符号化ブロック高さｃｂＨｅｉｇｈｔ／２に等しく、変数ｃｌｄｘｓｅｔを１に等しく入力として設定し、配列ｒｅｓＳａｍｐｌｅｓ$_{Cb}$を出力として、呼び出される。
　－ツリータイプがＳＩＮＧＬＥ＿ＴＲＥＥに等しい場合またはツリータイプがＤＵＡＬ＿ＴＲＥＥ＿ＣＨＲＯＭＡに等しい場合、８．５．８項に規定されるインター予測モードで符号化された符号化ブロックの残差信号の復号処理は、位置（ｘＴｂ０，ｙＴｂ０）をクロマ位置（ｘＣｂ／２，ｙＣｂ／２）、幅ｎＴｂＷをクロマ符号化ブロック幅ｃｂＷｉｄｔｈ／２、高さｎＴｂＨをクロマ符号化ブロック高さｃｂＨｅｉｇｈｔ／２に等しく、変数ｃｌｄｘｓｅｔを２に等しく入力として設定し、配列ｒｅｓＳａｍｐｌｅｓ$_{Cr}$を出力として、呼び出される。
【００７０】
５．現在の符号化ユニットの再構成されたサンプルは、以下のように導出される。
　－ツリータイプが、ＳＩＮＧＬＥ＿ＴＲＥＥに等しい場合またはツリータイプがＤＵＡＬ＿ＴＲＥＥ＿ＬＵＭＡに等しい場合、８．７．５項で規定されている色成分の画像再構成処理がブロック位置（ｘＢ，ｙＢ）を（ｘＣｂ，ｙＣｂ）、ブロック幅ｂＷｉｄｔｈをｃｂＷｉｄｔｈ、ブロック高さｂＨｅｉｇｈｔをｃｂＨｅｉｇｈｔ、変数ｃＩｄｘを０、（ｃｂＷｉｄｔｈ）×（ｃｂＨｅｉｇｈｔ）配列ｐｒｅｄＳａｍｐｌｅｓをｐｒｅｄＳａｍｐｌｅｓ$_{L}$、（ｃｂＷｉｄｔｈ）×（ｃｂＨｅｉｇｈｔ）配列ｒｅｓＳａｍｐｌｅｓをｒｅｓＳａｍｐｌｅｓ$_{L}$に等しく、入力設定して起動され、出力はループ内フィルタリング前の修正再構成画像となる。
　－ツリータイプがＳＩＮＧＬＥ＿ＴＲＥＥに等しい場合またはツリータイプがＤＵＡＬ＿ＴＲＥＥ＿ＣＨＲＯＭＡに等しい場合、８．７．５項で規定されている色成分の画像再構成処理がブロック位置（ｘＢ，ｙＢ）を（ｘＣｂ／２，ｙＣｂ／２）、ブロック幅ｂＷｉｄｔｈをｃｂＷｉｄｔｈ／２、ブロック高さｂＨｅｉｇｈｔをｃｂＨｅｉｇｈｔ／２、変数ｃＩｄｘを１、（ｃｂＷｉｄｔｈ／２）×（ｃｂＨｅｉｇｈｔ／２）配列ｐｒｅｄＳａｍｐｌｅｓをｐｒｅｄＳａｍｐｌｅｓ$_{Cb}$、（ｃｂＷｉｄｔｈ／２）×（ｃｂＨｅｉｇｈｔ／２）配列ｒｅｓＳａｍｐｌｅｓをｒｅｓＳａｍｐｌｅｓ$_{Cb}$に等しく、入力設定して起動され、出力はループ内フィルタリング前の修正再構成画像となる。
　－ツリータイプがＳＩＮＧＬＥ＿ＴＲＥＥに等しい場合またはツリータイプがＤＵＡＬ＿ＴＲＥＥ＿ＣＨＲＯＭＡに等しい場合、８．７．５項で規定されている色成分の画像再構成処理がブロック位置（ｘＢ，ｙＢ）を（ｘＣｂ／２，ｙＣｂ／２）、ブロック幅ｂＷｉｄｔｈをｃｂＷｉｄｔｈ／２、ブロック高さｂＨｅｉｇｈｔをｃｂＨｅｉｇｈｔ／２、変数ｃＩｄｘを２、（ｃｂＷｉｄｔｈ／２）×（ｃｂＨｅｉｇｈｔ／２）配列ｐｒｅｄＳａｍｐｌｅｓをｐｒｅｄＳａｍｐｌｅｓ$_{Cr}$、（ｃｂＷｉｄｔｈ／２）×（ｃｂＨｅｉｇｈｔ／２）配列ｒｅｓＳａｍｐｌｅｓをｒｅｓＳａｍｐｌｅｓ$_{Cr}$に等しく、入力設定して起動され、出力はループ内フィルタリング前の修正再構成画像となる。
【００７１】
２．４　適応型動きベクトル解像度（ＡＭＶＲ）

【００７２】
ＨＥＶＣにおいて、ｕｓｅ＿ｉｎｔｅｇｅｒ＿ｍｖ＿ｆｌａｇがスライスヘッダにおいて０であるとき、１／４輝度サンプルの単位で動きベクトル差分（ＭＶＤ：Ｍｏｔｉｏｎ　Ｖｅｃｔｏｒ　Ｄｉｆｆｅｒｅｎｃｅ）（動きベクトルとＣＵの予測動きベクトルとの差）が信号通知される。ＶＶＣにおいて、ＣＵレベルの適応型動きベクトル解像度（ＡＭＶＲ）スキームが導入される。ＡＭＶＲは、ＣＵのＭＶＤを異なる精度で符号化することを可能にする。現在のＣＵのモード（通常のＡＭＶＰモードまたはアフィンＡＶＭＰモード）に基づいて、現在のＣＵのＭＶＤは、以下のように適応的に選択できる。
－通常ＡＭＶＰモード：１／４輝度サンプル、整数輝度サンプルまたは４輝度サンプル。
－アフィンＡＭＶＰモード：１／４輝度サンプル、整数輝度サンプルまたは１／１６輝度サンプル。
【００７３】
現在のＣＵが少なくとも１つの非ゼロＭＶＤコンポーネントを有する場合、ＣＵレベルＭＶＤ解像度指示が条件付きで信号通知される。すべてのＭＶＤコンポーネント（すなわち、参照リストＬ０及び参照リストＬ１の水平及び垂直ＭＶＤの両方）がゼロである場合、１／４輝度サンプルＭＶＤ解像度が推測される。
【００７４】
少なくとも１つの非ゼロＭＶＤコンポーネントの構成要素を有するＣＵの場合、１／４輝度サンプルＭＶＤ精度がＣＵにおいて使用されるかどうかを示すために、第１のフラグが信号通知される。第１のフラグが０である場合、さらなる信号伝達は必要とされず、現在のＣＵのために１／４輝度サンプルＭＶＤ精度が使用される。そうでない場合、通常のＡＭＶＰ　ＣＵのために整数輝度サンプルまたは４輝度サンプルのＭＶＤ精度が使用されるかどうかを示すために、第２のフラグが信号通知される。同じ第２のフラグは、整数輝度サンプルまたは１／１６輝度サンプルのＭＶＤ精度がアフィンＡＭＶＰ　ＣＵに使用されるかどうかを示すために使用される。再構成されたＭＶが意図された精度（１／４輝度サンプル、１／４輝度サンプル、または４輝度サンプル）を有することを保証するために、ＣＵの動きベクトル予測子は、ＭＶＤと加算される前に、ＭＶＤと同じ精度に丸められる。動きベクトル予測子をゼロに向かって丸める（すなわち、負の動きベクトル予測子を正の無限大に向かって丸め、正の動きベクトル予測子を負の無限大に向かって丸める）。
【００７５】
エンコーダは、ＲＤチェックを使用して、現在のＣＵの動きベクトルの解像度を決定する。各ＭＶＤ解像度に対してＣＵレベルのＲＤチェックを常に３回実行することを回避するために、ＶＴＭ４では、１／４ルマサンプル以外のＭＶＤ精度のＲＤチェックは、条件付きでのみ呼び出される。通常のＡＶＭＰモードの場合、まず、１／４輝度サンプルＭＶＤ精度及び整数輝度サンプルＭＶ精度のＲＤコストを計算する。次に、整数輝度サンプルＭＶＤ精度のＲＤコストと１／４輝度サンプルＭＶＤ精度のＲＤコストを比較し、４輝度サンプルＭＶＤ精度のＲＤコストをさらにチェックする必要があるかどうかを決定する。１／４輝度サンプルＭＶＤ精度のＲＤコストが整数輝度サンプルＭＶＤ精度のＲＤコストよりもずっと小さい場合、４輝度サンプルＭＶＤ精度のＲＤチェックは省略される。アフィンＡＭＶＰモードにおいて、アフィンマージ／スキップモード、マージ／スキップモード、１／４輝度サンプリングＭＶＤ精度通常ＡＭＶＰモード、１／４輝度サンプリングＭＶＤ精度アフィンＡＭＶＰモードのレート－歪みコストをチェックした後、アフィンインターモードを選択しなかった場合、１／１６ルマサンプリングＭＶ精度、１－ｐｅｌ　ＭＶ精度アフィンインターモードはチェックされない。また、１／１６輝度サンプル及び１／４輝度サンプルＭＶ精度アフィンインターモードにおける探索開始点として、１／４輝度サンプルＭＶ精度アフィンインターモードで得られたアフィンパラメータを用いる。
【００７６】
２．５　ＨＥＶＣスクリーンコンテンツ符号化拡張機能におけるパレットモード（ＨＥＶＣ－ＳＣＣ）
【００７７】

パレットモードの背景にある基本的な考えは、ＣＵにおけるサンプルを代表的な色値の小さな集合で表現することである。この集合をパレットと呼ぶ。また、エスケープシンボルの後に（場合によっては量子化された）成分値を信号通知することによって、パレットの外側にあるサンプルを示すこともできる。これを図３に示す。

【００７８】

ＨＥＶＣ－ＳＣＣにおけるパレットモードでは、パレットおよびインデックスマップを符号化するために予測方式が用いられる。

【００７９】

２．５．１　パレットエントリの符号化

【００８０】

パレットエントリを符号化するために、パレット予測子が維持される。ＳＰＳにおいて、パレットの最大サイズおよびパレット予測子が信号通知される。ＨＥＶＣ－ＳＣＣにおいて、ｐａｌｅｔｔｅ＿ｐｒｅｄｉｃｔｏｒ＿ｉｎｉｔｉａｌｉｚｅｒ＿ｐｒｅｓｅｎｔ＿ｆｌａｇがＰＰＳに導入される。このフラグが１である場合、ビットストリームにおいて、パレット予測子を初期化するためのエントリが信号通知される。パレット予測子は、各ＣＴＵ行、各スライス、および各タイルの始めに初期化される。ｐａｌｅｔｔｅ＿ｐｒｅｄｉｃｔｏｒ＿ｉｎｉｔｉａｌｉｚｅｒ＿ｐｒｅｓｅｎｔ＿ｆｌａｇの値によって、ｐａｌｅｔｔｅ＿ｐｒｅｄｉｃｔｏｒを０にリセットするか、またはＰＰＳに信号通知されたパレット予測子の初期化エントリを使用してパレット予測子を初期化する。ＨＥＶＣ－ＳＣＣでは、ＰＰＳレベルでパレット予測子の初期化を明確に無効にするために、サイズ０のパレット予測子初期化子を有効化した。

【００８１】

パレット予測子のエントリごとに、再利用フラグが信号通知され、現在のパレットの一部であるかどうかが示される。これを図４に示す。再利用フラグは、ゼロのランレングス符号化を使用して送信される。この後、新しいパレットエントリの数は、次数０の指数ゴロンブコードを使用して信号通知される。最後に、新しいパレットエントリのコンポーネント値が信号通知される。

【００８２】

２．５．２　パレットインデックスの符号化

【００８３】

パレットインデックスは、図５に示すように、水平方向および垂直方向の横断走査を使用して符号化される。ｐａｌｅｔｔｅ＿ｔｒａｎｓｐｏｓｅ＿ｆｌａｇを使用して、ビットストリームにおける走査順序を明確に信号通知する。以下のサブセクションでは、走査が水平であると仮定する。

【００８４】

パレットインデックスは、２つのメインパレットサンプルモード、すなわち‘ＩＮＤＥＸ’および‘ＣＯＰＹ＿ＡＢＯＶＥ’を使用して符号化される。前述のように、エスケープシンボルも「ＩＮＤＥＸ」モードとして信号通知され、最大パレットサイズに等しいインデックスが割り当てられる。モードは、一番上の行を除いて、または前のモードが「ＣＯＰＹ＿ＡＢＯＶＥ」であった場合を除いて、フラグを使用して通知される。「ＣＯＰＹ＿ＡＢＯＶＥ」モードでは、上の行のサンプルのパレットインデックスをコピーする。「ＩＮＤＥＸ」モードにおいて、パレットインデックスは明確に信号通知される。「ＩＮＤＥＸ」モードと「ＣＯＰＹ＿ＡＢＯＶＥ」モードの両方の場合、同じモードを使用して符号化される後続のサンプルの数を規定する実行値を信号通知する。エスケープシンボルが「ＩＮＤＥＸ」または「ＣＯＰＹ＿ＡＢＯＶＥ」モードにおける実行の一部である場合、エスケープシンボルごとにエスケープ成分値が信号通知される。パレットインデックスの符号化を図６に示す。

【００８５】

この構文順序は、以下のようにして実行される。まず、ＣＵのためのインデックス値の数が信号通知される。これに続いて、トランケーテッドバイナリ符号化（ｔｒｕｎｃａｔ

ｅｄ ｂｉｎａｒｙ ｃｏｄｉｎｇ）を使用して、ＣＵ全体の実際のインデックス値を信号通知する。バイパスモードでは、インデックスの数およびインデックス値の両方が符号化される。これにより、インデックス関連バイパスビンがグループ化される。次に、パレットサンプルモード（必要な場合）および実行は、インターリーブ方式で信号通知される。最後に、ＣＵ全体のためのエスケープサンプルに対応する成分エスケープ値をグループ化し、バイパスモードで符号化する。

【００８６】

インデックス値を信号通知した後、追加の構文要素ｌａｓｔ＿ｒｕｎ＿ｔｙｐｅ＿ｆｌａｇを信号通知する。この構文要素は、インデックスの数と連動して、ブロックにおける最後の実行に対応する実行値を信号通知する必要をなくす。

【００８７】

ＨＥＶＣ－ＳＣＣでは、パレットモードは、４：２：２、４：２：０、およびモノクロのクロマフォーマットに対しても有効になる。パレットエントリおよびパレットインデックスの信号通知は、すべてのクロマフォーマットに対してほぼ同じである。非モノクロフォーマットの場合、各パレットエントリは、３つの成分からなる。モノクロフォーマットの場合、各パレットエントリは単一の成分からなる。サブサンプリングされたクロマ方向の場合、クロマサンプルは、２で割り切れる輝度サンプル指数に関連付けられる。ＣＵのパレットインデックスを再構成した後、１つのサンプルに単一の成分しか関連付けられていない場合、パレットエントリの第１の成分のみが使用される。信号通知における唯一の違いは、エスケープ成分値である。エスケープサンプルごとに、信号通知されるエスケープ成分値の数は、そのサンプルに関連付けられた成分の数によって異なってもよい。

【００８８】

２．６ 変換スキップモードにおける係数符号化

【００８９】

ＪＶＥＴ－Ｍ０４６４号及びＪＶＥＴ－Ｎ０２８０号において、残差符号化を変換スキップレベルの統計及び信号特性に適応させるために、ＴＳ（Ｔｒａｎｓｆｏｒｍ Ｓｋｉｐ）モードにおける係数符号化についていくつかの改良が提案されている。

【００９０】

提案された変形例を以下に示す。

【００９１】

<u>前回の有意な走査位置なし：</u>

残差信号は予測後の空間残差を反映しており、ＴＳに対して変換によるエネルギー圧縮は行われないので、変換ブロックの右下隅の末尾にゼロまたは有意でないレベルがより高い確率で後続することはもはやなくなる。従って、この場合、最後の重要な走査位置信号送信は省略される。

【００９２】

<u>サブブロックＣＢＦ：</u>

最後の重要な走査位置信号がない場合、ＴＳのｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇを有するサブブロックＣＢＦ信号は、次のように変更されることが必要である。

● 量子化のために、前述の有意でないシーケンスは、依然として変換ブロック内でローカルに発生し得る。したがって、前記のように最後の有意な走査位置が削除され、最後を除くすべてのＣＧがゼロ係数を持つ場合を除いて、すべてのサブブロックに対してｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇが符号化されるため、最後のＣＧに対してｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇを符号化する必要がない。

● ＤＣ周波数位置をカバーするサブブロック（左上のサブブロック）のｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇは、特殊な場合を提示する。ＶＶＣ草案３において、このサブブロックのｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇは決して信号通知されず、常に１に等しいと推測される。最後の有意な走査位置が別のサブブロックに位置する場合、それは、ＤＣサブブロックの外側に少なくとも１つの有意なレベルがあることを意味する。その結果、ＤＣサブブロックは、このサブブロックのｃｏｄｅｄ＿ｓｕｂｂｌｏｃｋ＿ｆ

ｌａｇが１に等しいと推測されるが、ゼロ／非有意レベルのみを含んでもよい。ＴＳに最後の走査位置情報がない場合、各サブブロックのｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇが通知される。これは、他のｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇ構文要素が既に０に等しい場合を除き、ＤＣサブブロックのｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇをも含む。この場合、ＤＣ　ｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇは１に等しいと推測される（ｉｎｆｅｒＤｃＳｂＣｂｆ＝１）。このＤＣサブブロックには少なくとも１つの有意なレベルがなければならないので、このＤＣサブブロックにおける他のすべてのｓｉｇ＿ｃｏｅｆｆ＿ｆｌａｇ構文要素が０に等しい場合、（０，０）における第１の位置のｓｉｇ＿ｃｏｅｆｆ＿ｆｌａｇ構文要素は信号通知されず、１に等しくなるように導出される（ｉｎｆｅｒＳｂＤｃＳｉｇＣｏｅｆｆｌａｇ＝１）。

● ｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇのコンテクストモデリングを変更する。コンテクストモデルインデックスは、ｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇの左側と、現在のサブブロック上のｃｏｄｅｄ＿ｓｕｂ＿ｂｌｏｃｋ＿ｆｌａｇとの合計として、両方の論理和の代わりに計算される。

【００９３】

ｓｉｇ＿ｃｏｅｆｆ＿ｆｌａｇコンテクストモデリング：

ｓｉｇ＿ｃｏｅｆｆ＿ｆｌａｇコンテクストモデルにおけるローカルテンプレートは、現在地の走査位置の左側（ＮＢ$_0$）及び上側（ＮＢ$_1$）の近傍のみを含むように修正される。コンテクストモデルオフセットは、重要な近傍位置ｓｉｇ＿ｃｏｅｆｆ＿ｆｌａｇ［ＮＢ$_0$］＋ｓｉｇ＿ｃｏｅｆｆ＿ｆｌａｇ［ＮＢ$_1$］の個数に過ぎない。そこで、今回の変換区間における対角ｄに応じて異なるコンテクスト集合を選択することを排除する。その結果、ｓｉｇ＿ｃｏｅｆｆ＿ｆｌａｇを符号化するための３つのコンテクストモデル及び１つのコンテクストモデル集合が得られる。

【００９４】

ａｂｓ＿ｌｅｖｅｌ＿ｇｔ１＿ｆｌａｇ及びｐａｒ＿ｌｅｖｅｌ＿ｆｌａｇコンテクストモデリング：

ａｂｓ＿ｌｅｖｅｌ＿ｇｔ１＿ｆｌａｇ及びｐａｒ＿ｌｅｖｅｌ＿ｆｌａｇには１つのコンテクストモデルを用いる。

【００９５】

ａｂｓ＿ｒｅｍａｉｎｄｅｒ符号化：

変換スキップ残差絶対レベルの経験的分布は、典型的には、依然としてラプラシアンまたは幾何学的分布に適応するが、変換係数絶対レベルよりも大きい不安定性が存在する。具体的には、残留絶対値レベルの場合、連続して実現する窓内の分散が高くなる。これは、ａｂｓ＿ｒｅｍａｉｎｄｅｒシンタックスの二値化及びコンテクストモデリングを以下のように修正する動機となる。

● ２値化においてより高いカットオフ値、即ち、ｓｉｇ＿ｃｏｅｆｆ＿ｆｌａｇ、ａｂｓ＿ｌｅｖｅｌ＿ｇｔ１＿ｆｌａｇ、ｐａｒ＿ｌｅｖｅｌ＿ｆｌａｇ、及びａｂｓ＿ｌｅｖｅｌ＿ｇｔ３を用いた符号化からａｂｓ＿ｒｅｍａｉｎｄｅｒのＲｉｃｅ符号への移行点と、各ビンの位置に専用のコンテクストモデルを用いることで、より高い圧縮効率が得られる。カットオフを大きくすると、より多くの「Ｘより大きい」フラグがもたらされ、例えば、カットオフに達するまで、ａｂｓ＿ｌｅｖｅｌ＿ｇｔ５＿ｆｌａｇ、ａｂｓ＿ｌｅｖｅｌ＿ｇｔ７＿ｆｌａｇ等を導入していく。カットオフ自体は５に固定される（ｎｕｍＧｔＦｌａｇｓ＝５）。

● ライスパラメータ導出のためのテンプレートを修正し、すなわち、現在の走査位置の左側の近傍及び上側の近傍のみを、ｓｉｇ＿ｃｏｅｆｆ＿ｆｌａｇコンテクストモデリングのためのローカルテンプレートに類似していると見なす。

【００９６】

ｃｏｅｆｆ＿ｓｉｇｎ＿ｆｌａｇコンテクストモデリング：

符号のシーケンス内の不安定性及び予測残差がしばしば偏っていることに起因して、全体的な経験的分布がほぼ均一に分布している場合であっても、符号はコンテクストモデルを

使用して符号化され得る。符号の符号化には１つの専用コンテクストモデルが使用され、符号はｓｉｇ＿ｃｏｅｆｆ＿ｆｌａｇの後に構文解析されて、すべてのコンテクスト符号化ビンが一緒にまとめられる。

【００９７】

２．７　量子化残差ブロック差動パルスコード変調（ＱＲ－ＢＤＰＣＭ）

【００９８】

ＪＶＥＴ－Ｍ０４１３において、量子化残差ブロック差動パルスコード変調（ＱＲ－ＢＤＰＣＭ）が、スクリーンコンテンツを効率的に符号化するために提案されている。

【００９９】

ＱＲ－ＢＤＰＣＭで使用される予測方向は、垂直予測モードおよび水平予測モードであり得る。イントラ予測は、イントラ予測と同様に、予測方向（水平または垂直予測）にサンプルコピーすることで、ブロック全体で予測する。残差を量子化し、量子化残差とその予測子（水平または垂直）量子化値との間のデルタを符号化する。これは、以下のように説明することができる。サイズＭ（行）×Ｎ（列）のブロックについて、$r_{i,j}$，０≦ｉ≦Ｍ－１、０≦ｊ≦Ｎ－１を、上または左ブロックの境界サンプルからのフィルタリングされていないサンプルを使用して、水平方向（予測ブロックに対して左隣の画素値を１ラインずつコピーする）または垂直方向（予測ブロックにおける各ラインに上隣のラインをコピーする）にイントラ予測を行った後の予測残差とする。Ｑ（$r_{i,j}$）、０≦ｉ≦Ｍ－１、０≦ｊ≦Ｎ－１は、残差$r_{i,j}$の量子化バージョンを表し、この場合、残差は、元のブロックと予測ブロック値との間の差である。次に、ブロックＤＰＣＭが量子化残差サンプルに適用され、その結果、要素$\tilde{r}_{i,j}$を有する修正されたＭ×Ｎ個の配列$\tilde{R}$が得られる。垂直ＢＤＰＣＭが信号通知されると、以下のようになる。

【０１００】

【数１】

$$\tilde{r}_{i,j} = \begin{cases} Q(r_{i,j}), & i = 0,\ \ 0 \le j \le (N-1) \\ Q(r_{i,j}) - Q(r_{(i-1),j}), & 1 \le i \le (M-1),\ \ 0 \le j \le (N-1) \end{cases}$$

【０１０１】

水平予測の場合、類似した規則が適用され、残差量子化サンプルは、以下の式によって得られる。

【０１０２】

【数２】

$$\tilde{r}_{i,j} = \begin{cases} Q(r_{i,j}), & 0 \le i \le (M-1),\ \ j = 0 \\ Q(r_{i,j}) - Q(r_{i,(j-1)}), & 0 \le i \le (M-1),\ \ 1 \le j \le (N-1) \end{cases}$$

【０１０３】

残差量子化サンプル$\tilde{r}_{i,j}$はデコーダに送られる。

【０１０４】

デコーダ側では、上記の計算を逆にして、Ｑ（$r_{i,j}$）、０≦ｉ≦Ｍ－１、０≦ｊ≦Ｎ－１を生成する。垂直予測の場合、以下である。

【０１０５】

【数３】

$$Q(r_{i,j}) = \sum_{k=0}^{i} \tilde{r}_{k,j}, \quad 0 \le i \le (M-1), \quad 0 \le j \le (N-1)$$

【０１０６】
水平方向の場合、以下である。
【０１０７】
【数４】

$$Q(r_{i,j}) = \sum_{k=0}^{j} \tilde{r}_{i,k}, \quad 0 \le i \le (M-1), \quad 0 \le j \le (N-1)$$

【０１０８】
逆量子化された残差Ｑ$^{-1}$（Ｑ（ｒ$_{i}$，$_{j}$））をイントラブロック予測値に加算し、再構成されたサンプル値を生成する。
【０１０９】
このスキームの主な利点は、逆方向のＤＰＣＭを、係数の構文解析中にオンザフライで行うことができ、係数の構文解析中に予測子を追加するだけで済むこと、または、構文解析後に行うことができることである。
【０１１０】
ＱＲ－ＢＤＰＣＭの本文変更案を以下に示す。
【０１１１】

【表１】

７．３．６．５　符号化ユニット構文

coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	Descriptor
if(tile_group_type != I || sps_ibc_enabled_flag) {	
if(treeType != DUAL_TREE_CHROMA)	
cu_skip_flag[x0][y0]	ae(v)
if(cu_skip_flag[x0][y0] = = 0 && tile_group_type != I)	
pred_mode_flag	ae(v)
if(((tile_group_type = = I && cu_skip_flag[x0][y0] = =0) || (tile_group_type != I && CuPredMode[x0][y0] != MODE_INTRA)) && sps_ibc_enabled_flag)	
pred_mode_ibc_flag	ae(v)
}	
if(CuPredMode[x0][y0] = = MODE_INTRA) {	
if(pred_mode_flag = = MODE_INTRA && (cIdx == 0) && (cbWidth <= 32) && (CbHeight <= 32)) {	
bdpcm_flag[x0][y0]	ae(v)
if(bdpcm_flag[x0][y0]) {	
bdpcm_dir_flag[x0][y0]	ae(v)
}	
else {	
if(sps_pcm_enabled_flag && cbWidth >= MinIpcmCbSizeY && cbWidth <= MaxIpcmCbSizeY && cbHeight >= MinIpcmCbSizeY && cbHeight <= MaxIpcmCbSizeY)	
pcm_flag[x0][y0]	ae(v)
if(pcm_flag[x0][y0]) {	
while(!byte_aligned( ))	
pcm_alignment_zero_bit	f(1)
pcm_sample(cbWidth, cbHeight, treeType)	
} else {	
if(treeType = = SINGLE_TREE || treeType = = DUAL_TREE_LUMA) {	

【表２】

if((y0 % CtbSizeY) > 0)	
intra_luma_ref_idx[x0][y0]	ae(v)
if (intra_luma_ref_idx[x0][y0] = = 0 && (cbWidth <= MaxTbSizeY || cbHeight <= MaxTbSizeY) && (cbWidth * cbHeight > MinTbSizeY * MinTbSizeY))	
intra_subpartitions_mode_flag[x0][y0]	ae(v)
if(intra_subpartitions_mode_flag[x0][y0] = = 1 && cbWidth <= MaxTbSizeY && cbHeight <= MaxTbSizeY)	
intra_subpartitions_split_flag[x0][y0]	ae(v)
if(intra_luma_ref_idx[x0][y0] = = 0 && intra_subpartitions_mode_flag[x0][y0] = = 0)	
intra_luma_mpm_flag[x0][y0]	ae(v)
if(intra_luma_mpm_flag[x0][y0])	
intra_luma_mpm_idx[x0][y0]	ae(v)
else	
intra_luma_mpm_remainder[x0][y0]	ae(v)
}	
}	
if(treeType = = SINGLE_TREE || treeType = = DUAL_TREE_CHROMA)	
intra_chroma_pred_mode[x0][y0]	ae(v)
}	
} else if(treeType != DUAL_TREE_CHROMA) { /* MODE_INTER or MODE_IBC */	
...	
}	

【０１１２】
ｂｄｐｃｍ＿ｆｌａｇ［ｘ０］［ｙ０］が１の場合、位置（ｘ０，ｙ０）の輝度符号化ブロックを含む符号化ユニットにｂｄｐｃｍ＿ｄｉｒ＿ｆｌａｇが存在することを指定する。
ｂｄｐｃｍ＿ｄｉｒ＿ｆｌａｇ［ｘ０］［ｙ０］が０の場合は、ｂｄｐｃｍブロックで使用され
る予測方向が水平であることを指定し、そうでない場合、垂直である。
【０１１３】
２．８　パーティション構造
【０１１４】
　ＨＥＶＣにおいて、ＣＴＵは、様々な局所的特徴に適応するように、符号化ツリーと呼ばれる四分木構造を用いてＣＵに分割される。インターピクチャ（時間的）予測またはイントラピクチャ（空間的）予測を使用する、ピクチャ領域を符号化するかどうかの決定は、リーフＣＵレベルで行われる。各リーフＣＵは、ＰＵ分割タイプに応じて１つ、２つまたは４つのＰＵに更に分割することができる。１つのＰＵの内部では、同じ予測処理が適用され、ＰＵ単位で関連情報がデコーダに送信される。ＰＵ分割タイプに基づく予測処理

を適用して残差ブロックを得た後、ＣＵのためのコーディングツリー符号化ツリーに類似した別の四分木構造に基づいて、リーフＣＵを変換ユニット（ＴＵ）に分割することができる。ＨＥＶＣ構造の重要な特徴の１つは、ＣＵ、ＰＵ、ＴＵを含む複数のパーティション概念を有することである。

【０１１５】

ＶＶＣにおいて、２値及び３値分割セグメンテーション構造を使用するネストされたマルチタイプツリーを有する四分木は、複数の区分ユニットタイプの概念に取って代わる。即ち、それは、最大変換長さに対して大き過ぎるサイズを有するＣＵに必要な場合を除き、ＣＵ、ＰＵ、及びＴＵ概念の分離を排除し、且つＣＵ区分形状のためのより多くの柔軟性をサポートする。符号化ツリー構造において、ＣＵは正方形または長方形のいずれかを有することができる。まず、符号化ツリーユニット（ＣＴＵ）を別名、四分木構造で分割する。そして、四分木のリーフのノードは、マルチタイプのツリー構造によってさらに区分され得る。図７に示すとおり、マルチタイプツリー構造の分岐タイプには、垂直二分岐（ＳＰＬＩＴ＿ＢＴ＿ＶＥＲ）、水平二分岐（ＳＰＬＩＴ＿ＢＴ＿ＨＯＲ）、垂直三分岐（ＳＰＬＩＴ＿ＴＴ＿ＶＥＲ）、水平三分岐（ＳＰＬＩＴ＿ＴＴ＿ＨＯＲ）の４つがある。マルチタイプツリーのリーフのノードは、符号化ユニット（ＣＵ）と呼ばれ、ＣＵが大き過ぎて最大変換長にならない限り、このセグメント化は、それ以上の分割なしに、予測及び変換処理に使用される。これは、ほとんどの場合、ＣＵ、ＰＵ、及びＴＵが、ネストされたマルチタイプのツリー符号化ブロック構造を有する四分木において、同じブロックサイズを有することを意味する。サポートされる最大変換長がＣＵの色成分の幅または高さよりも小さい場合、この例外が生じる。また、輝度及びクロマ成分は、Ｉタイル上に別個の区分構造を有する。さらに、ＪＶＥＴ－Ｋ０３５３およびＪＶＥＴ－Ｋ０３５４は、ＣＴＵ／ＣＵレベルで別個のパーティション構造を使用するかどうかを判定するためにフラグを信号化通知することを提案している。

【０１１６】

２．９　ＶＴＭ－４．０における非ブロック化スキーム

【０１１７】

なお、以下の説明において、ｐＮＭは、垂直エッジに対してＭ行目の左側のＮ番目のサンプル、または水平エッジに対してＭ列目の上側のＮ番目のサンプルを表し、ｑＮＭは、垂直エッジに対してＭ行目の右側のＮ番目のサンプル、または水平エッジに対して、Ｍ列目の下側のＮ番目のサンプルを表す。pN_MおよびqN_Mの例を図８に示す。

【０１１８】

なお、以下の説明において、ｐＮは、垂直エッジに対して行の左側のＮ番目のサンプル、または、水平エッジに対して列の上側のＮ番目のサンプルを表し、ｑＮは、垂直エッジに対して行の右側のＮ番目のサンプル、または水平エッジに対して、列の下側のＮ番目のサンプルを表す。

【０１１９】

１つの単位として４行に対してフィルタのオン／オフの決定を行う。図８は、フィルタのオン／オフ決定に関与する画素を示す。最初の４行のための２つの赤いボックスにおける６つの画素は、４行のためのフィルタのオン／オフを判定するために用いられる。２番目の４行のための２つの赤いボックス内の６つの画素は、第２の４行のためのフィルタのオン／オフを判定するために用いられる。

【０１２０】

現在のＶＴＭ、つまりＶＴＭ－４．０では、ＪＶＥＴ－Ｍ０４７１で説明されている非ブロック化スキームが使用される。まずピクチャの垂直エッジを選別する。そして、垂直エッジフィルタリング処理で修正されたサンプルを入力として、ピクチャの水平エッジをフィルタリングする。各ＣＴＵのＣＴＢにおける垂直および水平エッジは、符号化ユニットごとに別個に処理される。符号化ユニットにおける符号化ブロックの垂直エッジは、符号化ブロックの左側のエッジから始まり、符号化ブロックの右側に向かってそれらの幾何学的順にエッジを通って進むようにフィルタリングされる。符号化ユニットにおける符号

化ブロックの水平エッジは、符号化ブロックの上側のエッジから始まり、符号化ブロックの下側に向かってそれらの幾何学的順にエッジを通って進むようにフィルタリングされる。

【０１２１】

２．９．１　境界の決定

【０１２２】

８×８のブロック境界にフィルタリングを適用する。さらに、それは、（例えば、アフィン動き予測、ＡＴＭＶＰを使用しているため）変換ブロックの境界または符号化サブブロックの境界でなければならない。そのような境界でない場合、フィルタは無効にされる。

【０１２３】

２．９．２　境界強度計算

【０１２４】

変換ブロックの境界／符号化サブブロックの境界の場合、それが８×８グリッドに位置する場合、それをフィルタリングしてもよくジのためのｂＳ［ｘＤ$_i$］［ｙＤ$_j$］（［ｘＤ$_i$］［ｙＤ$_j$］は座標を表す）の設定は、以下のように定義される。

－　ｂＳ［ｘＤ$_i$］［ｙＤ$_j$］は、２に等しく設定され、この場合、サンプルｐ$_0$またはｑ$_0$は、イントラ予測モードで符号化された符号化ユニットの符号化ブロックにある。

－　あるいは、ブロックエッジが変換ブロックエッジでもあり、サンプルｐ$_0$またはｑ$_0$が、１つ以上の非ゼロ変換係数レベルを含む変換イントラブロックにある場合、ｂＳ［ｘＤ$_i$］［ｙＤ$_j$］は、１に等しく設定される。

－　あるいは、サンプルｐ$_0$を含む符号化サブブロックの予測モードが、サンプルｑ$_0$を含む符号化サブブロックの予測モードと異なる場合、ｂＳ［ｘＤ$_i$］［ｙＤ$_j$］は、１に等しく設定される。

－　あるいは、以下の条件の１つ以上がＴＲＵＥである場合、ｂＳ［ｘＤ$_i$］［ｙＤ$_j$］を１に等しく設定する。

－　サンプルｐ$_0$を含む符号化サブブロックおよびサンプルｑ$_0$を含む符号化サブブロックは、いずれもＩＢＣ予測モードで符号化され、２つの符号化サブブロックの予測に用いられる動きベクトルの水平または垂直成分の絶対差は、１／４輝度サンプル単位で４以上である。

－　サンプルｐ$_0$を含む符号化サブブロックの予測のために、サンプルｑ$_0$を含む符号化サブブロックの予測とは異なる参照ピクチャまたは異なる数の動きベクトルが使用される。

注１－２つの符号化サブロックに使用される参照ピクチャが同じであるかまたは異なるかは、予測を形成するのに参照ピクチャリスト０へのインデックスを使用するか、または参照ピクチャリスト１へのインデックスを使用して形成するかに関わらず、且つ参照ピクチャリスト内のインデックス位置が異なるかどうかに関わらず、どのピクチャが参照されるかによってのみに基づいて判定される。

注２－（ｘＳｂ，ｙＳｂ）を含む左上のサンプルを有する符号化サブブロックの予測に使用される動きベクトルの数は、ＰｒｅｄＦｌａｇＬ０［ｘＳｂ］［ｙＳｂ］＋ＰｒｅｄＦｌａｇＬ１［ｘＳｂ］［ｙＳｂ］に等しい。

－　１つの動きベクトルは、サンプルｐ$_0$を含む符号化サブブロックを予測するために使用され、１つの動きベクトルは、サンプルｑ$_0$を含む符号化サブブロックを予測するために使用され、使用される動きベクトルの水平または垂直成分の絶対差は、１／４輝度サンプル単位で４以上である。

－　２つの動きベクトルおよび２つの異なる参照ピクチャを使用して、サンプルｐ$_0$を含む符号化サブブロックを予測し、同じ２つの参照ピクチャの２つの動きベクトルを使用して、サンプルｑ$_0$を含む符号化サブブロックを予測し、同じ参照ピクチャの２つの符号化サブブロックの予測に使用される２つの動きベクトルの水平または垂直成分の絶対差は、１／４輝度サンプル単位で４以上である。

－　同じ参照ピクチャの２つの動きベクトルを使用して、サンプルp_0を含む符号化サブブロックを予測し、同じ参照ピクチャの２つの動きベクトルを使用して、サンプルq_0を含む符号化サブブロックを予測し、以下の条件の両方が成り立つ。

－　２つの符号化サブブロックの予測に使用されるリスト０の動きベクトルの水平または垂直成分の間の絶対差は、１／４輝度サンプルにおいて４以上である、または２つの符号化サブブロックの予測に使用されるリスト１の動きベクトルの水平または垂直成分の間の絶対差は、４分の１輝度サンプル単位で４以上である。

－　サンプルp_0を含む符号化サブブロックの予測に使用されるリスト０動きベクトルの水平または垂直成分と、サンプルq_0を含む符号化サブブロックの予測に使用されるリスト１動きベクトルとの間の絶対差は、１／４輝度サンプル単位で４以上であるか、またはサンプルp_0を含む符号化サブブロックの予測に使用されるリスト１動きベクトルの水平または垂直成分と、サンプルq_0を含む符号化サブブロックの予測に使用されるリスト０動きベクトルとの間の絶対差は、１／４輝度サンプル単位で４以上である。

－　あるいは、変数ｂＳ［xD_i］［yD_j］を０に設定する。

【０１２５】

表１及び表２にＢＳの算出規則をまとめる。

【０１２６】

【表３】

表１：境界強度（ＳＰＳ　ＩＢＣが無効な場合）

優先順位	条件	Y	U	V
5	隣接するブロックのうちの少なくとも１つは、イントラである	2	2	2
4	TU境界および隣接するブロックのうちの少なくとも１つは、非ゼロ変換係数を有する	1	1	1
3	隣接するブロックの参照ピクチャまたはMVの数が異なる（単一予測の場合、1、双予測の場合、2）	1	N/A	N/A
2	隣接するブロックに属する同じ参照ピクチャの動きベクトル間の絶対差は、1つの整数輝度サンプル以上である	1	N/A	N/A
1	そうでない場合	0	0	0

【０１２７】

【表４】

表２：境界強度（ＳＰＳ　ＩＢＣが有効な場合）

優先順位	条件	Y	U	V
8	隣接するブロックのうちの少なくとも１つは、イントラである	2	2	2
7	TU境界および隣接するブロックのうちの少なくとも１つは、非ゼロ変換係数を有する	1	1	1
6	隣接するブロックの予測モードが異なる(例えば、1つはIBCであり、1つはインターである)。	1		
5	隣接するブロックに属する動きベクトル間のIBCおよび絶対差は、いずれも1つの整数輝度サンプル以上である。	1	N/A	N/A
4	隣接するブロックの参照ピクチャまたはMVの数が異なる（単一予測の場合、1、双予測の場合、2）	1	N/A	N/A
3	隣接するブロックに属する同じ参照ピクチャの動きベクトル間の絶対差は、1つの整数輝度サンプル以上である	1	N/A	N/A
1	そうでない場合	0	0	0

【０１２８】

２．９．３　輝度成分の非ブロック化決定

【０１２９】

非ブロック化決定処理は、このサブセクションで説明する。

【０１３０】

より広く、強い輝度フィルタは、条件１、条件２、および条件３の全てがＴＲＵＥである場合にのみ使用されるフィルタである。

【０１３１】

条件１は、「大ブロック条件」である。この条件は、Ｐ側およびＱ側のサンプルが、それぞれ変数ｂＳｉｄｅＰｉｓＬａｒｇｅＢｌｋおよびｂＳｉｄｅＱｉｓＬａｒｇｅＢｌｋによって表現される大ブロックに属するかどうかを検出する。ｂＳｉｄｅＰｉｓＬａｒｇｅＢｌｋおよびｂＳｉｄｅＱｉｓＬａｒｇｅＢｌｋは、以下のように定義される。

ｂＳｉｄｅＰｉｓＬａｒｇｅＢｌｋ＝（（ｅｄｇｅ ｔｙｐｅ ｉｓ ｖｅｒｔｉｃａｌ ａｎｄ p_0 ｂｅｌｏｎｇｓ ｔｏ ＣＵ ｗｉｔｈ ｗｉｄｔｈ＞＝３２）｜｜（ｅｄｇｅ ｔｙｐｅ ｉｓ ｈｏｒｉｚｏｎｔａｌ ａｎｄ p_0 ｂｅｌｏｎｇｓ ｔｏ ＣＵ ｗｉｔｈ ｈｅｉｇｈｔ＞＝３２））？ＴＲＵＥ：ＦＡＬＳＥ

ｂＳｉｄｅＱｉｓＬａｒｇｅＢｌｋ＝（（ｅｄｇｅ ｔｙｐｅ ｉｓ ｖｅｒｔｉｃａｌ ａｎｄ q_0 ｂｅｌｏｎｇｓ ｔｏ ＣＵ ｗｉｔｈ ｗｉｄｔｈ＞＝３２）｜｜（ｅｄｇｅ ｔｙｐｅ ｉｓ ｈｏｒｉｚｏｎｔａｌ ａｎｄ q_0 ｂｅｌｏｎｇｓ ｔｏ ＣＵ ｗｉｔｈ ｈｅｉｇｈｔ＞＝３２））？ＴＲＵＥ：ＦＡＬＳＥ

【０１３２】

ｂＳｉｄｅＰｉｓＬａｒｇｅＢｌｋおよびｂＳｉｄｅＱｉｓＬａｒｇｅＢｌｋに基づいて、条件１を以下のように定義する。

条件１＝（ｂＳｉｄｅＰｉｓＬａｒｇｅＢｌｋ ｜｜ ｂＳｉｄｅＰｉｓＬａｒｇｅＢｌｋ）？ＴＲＵＥ：ＦＡＬＳＥ

【０１３３】

次に、条件１がＴＲＵＥである場合、さらに条件２をチェックする。まず、以下の変数を導出する。

－ ｄｐ０，ｄｐ３，ｄｑ０，ｄｑ３をまずＨＥＶＣとして導出する

－ もし（ｐ側が３２以上）なら、
ｄｐ０＝（ｄｐ０＋Ａｂｓ（ｐ５$_0$－２＊ｐ４$_0$＋ｐ３$_0$）＋１）＞＞１
ｄｐ３＝（ｄｐ３＋Ａｂｓ（ｐ５$_3$－２＊ｐ４$_3$＋ｐ３$_3$）＋１）＞＞１
－ もし（ｑ側が３２以上）なら、
ｄｑ０＝（ｄｑ０＋Ａｂｓ（ｑ５$_0$－２＊ｑ４$_0$＋ｑ３$_0$）＋１）＞＞１
ｄｑ３＝（ｄｑ３＋Ａｂｓ（ｑ５$_3$－２＊ｑ４$_3$＋ｑ３$_3$）＋１）＞＞１
条件２＝（ｄ＜β）？ＴＲＵＥ：ＦＡＬＳＥ
章２．２．４に示すとおり、式中、ｄ＝ ｄｐ０＋ｄｑ０＋ｄｐ３＋ｄｑ３である。
【０１３４】
条件１および条件２が有効である場合、いずれかのブロックがサブブロックを使用するかどうかをさらにチェックする。

```
Ｉｆ（ｂＳｉｄｅＰｉｓＬａｒｇｅＢｌｋ）
  Ｉｆ（ｍｏｄｅ ｂｌｏｃｋ Ｐ＝＝ＳＵＢＢＬＯＣＫＭＯＤＥ）
     Ｓｐ＝５
  ｅｌｓｅ
     Ｓｐ＝７
ｅｌｓｅ
  Ｓｐ＝３
Ｉｆ （ｂＳｉｄｅＱｉｓＬａｒｇｅＢｌｋ）
   Ｉｆ （ｍｏｄｅ ｂｌｏｃｋ Ｑ＝＝ＳＵＢＢＬＯＣＫＭＯＤＥ）
      Ｓｑ＝５
   ｅｌｓｅ
      Ｓｑ＝７
ｅｌｓｅ
   Ｓｑ＝３
```

【０１３５】
最後に、条件１および条件２の両方が有効である場合、提案された非ブロック化方法は、以下のように定義される条件３（大ブロックの強いフィルタ条件）をチェックする。条件３のＳｔｒｏｎｇＦｉｌｔｅｒＣｏｎｄｉｔｉｏｎにおいて、以下の変数を導出する。
ｄｐｑはＨＥＶＣと同様に導出される。
ｓｐ$_3$＝Ａｂｓ（ｐ$_3$－ｐ$_0$）、ＨＥＶＣと同様に導出される
もし（ｐ側が３２以上）なら、

```
      ｉｆ（Ｓｐ＝＝５）
         ｓｐ$_3$＝（ｓｐ$_3$＋Ａｂｓ（ｐ$_5$－ｐ$_3$）＋１）＞＞１
      ｅｌｓｅ
         ｓｐ$_3$＝（ｓｐ$_3$＋Ａｂｓ（ｐ$_7$－ｐ$_3$）＋１）＞＞１
```

ｓｑ$_3$＝Ａｂｓ（ｑ$_0$－ｑ$_3$）は、ＨＥＶＣと同様に導出される
もし（ｑ側が３２以上）なら、

```
   Ｉｆ（Ｓｑ＝＝５）
      ｓｑ$_3$＝（ｓｑ$_3$＋Ａｂｓ（ｑ$_5$－ｑ$_3$）＋１）＞＞１
   ｅｌｓｅ
      ｓｑ$_3$＝（ｓｑ$_3$＋Ａｂｓ（ｑ$_7$－ｑ$_3$）＋１）＞＞１
```

【０１３６】
ＨＥＶＣと同様に、ＳｔｒｏｎｇＦｉｌｔｅｒＣｏｎｄｉｔｉｏｎ＝（ｄｐｑは（β＞＞２）未満であり、ｓｐ$_3$＋ｓｑ$_3$は（３＊β＞＞５）未満であり、且つＡｂｓ（ｐ$_0$－ｑ$_0$）は（５＊ｔ$_C$＋１）＞＞１未満である）？ＴＲＵＥ：ＦＡＬＳＥ．
【０１３７】
２．９．４ 輝度のためのより強い非ブロック化フィルタ（より大きいブロックのために設計される）

【０１３８】
　バイリニアフィルタは、境界の両側のサンプルが１つの大ブロックに属する場合に用いられる。１つの大ブロックに属する１つのサンプルは、垂直エッジの場合、幅≧３２であり、水平エッジの場合、高さ≧３２であるときとして定義される。
【０１３９】
　バイリニアフィルタを以下に示す。
【０１４０】
　次に、上述のＨＥＶＣ非ブロック化において、ｉ＝０～Ｓｐ－１でのブロックの境界サンプルp_i及びｊ＝０～Ｓｑ－１のブロックの境界サンプルq_i（ｐｉ、ｑｉは、垂直エッジをフィルタリングする行内のｉ番目のサンプル、または水平エッジをフィルタリングする列内のｉ番目のサンプル）を、以下のように線形補間によって置き換える。
【０１４１】
【数５】

$$- \ p_i' = (f_i * Middle_{s,t} + (64 - f_i) * P_s + 32) \gg 6), clipped\ to\ p_i \pm tcPD_i$$
$$- q_j' = (g_j * Middle_{s,t} + (64 - g_j) * Q_s + 32) \gg 6), clipped\ to\ q_j \pm tcPD_j$$

【０１４２】
$tcPD_i$およびcPD_j項は、章２．９．７に記載の位置依存クリッピングであり、g_j，f_i，$Middle_{s,t}$，P_sおよびQ_sは、表３に示される。
【０１４３】

【表５】

表３：ロングタップ非ブロック化フィルタ

Sp, Sq 7, 7 (p side: 7, q side: 7)	$f_i = 59 - i * 9$, *can also be described as* $\boldsymbol{f} = \{59,50,41,32,23,14,5\}$ $g_j = 59 - j * 9$, *can also be described as* $\boldsymbol{g} = \{59,50,41,32,23,14,5\}$ $Middle_{7,7} = (2 * (p_o + q_o) + p_1 + q_1 + p_2 + q_2 + p_3 + q_3 + p_4 + q_4 + p_5 + q_5 + p_6 + q_6 + 8) \gg 4$ $P_7 = (p_6 + p_7 + 1) \gg 1, \qquad Q_7 = (q_6 + q_7 + 1) \gg 1$
7, 3 (p side: 7 q side: 3)	$f_i = 59 - i * 9$, *can also be described as* $\boldsymbol{f} = \{59,50,41,32,23,14,5\}$ $g_j = 53 - j * 21$, *can also be described as* $\boldsymbol{g} = \{53,32,11\}$ $Middle_{7,3} = (2 * (p_o + q_o) + q_0 + 2 * (q_1 + q_2) + p_1 + q_1 + p_2 + p_3 + p_4 + p_5 + p_6 + 8) \gg 4$ $P_7 = (p_6 + p_7 + 1) \gg 1, \qquad Q_3 = (q_2 + q_3 + 1) \gg 1$
3, 7 (p side: 3 q side: 7)	$g_j = 59 - j * 9$, *can also be described as* $\boldsymbol{g} = \{59,50,41,32,23,14,5\}$ $f_i = 53 - i * 21$, *can also be described as* $\boldsymbol{f} = \{53,32,11\}$ $Middle_{3.7} = (2 * (q_o + p_o) + p_0 + 2 * (p_1 + p_2) + q_1 + p_1 + q_2 + q_3 + q_4 + q_5 + q_6 + 8) \gg 4$ $Q_7 = (q_6 + q_7 + 1) \gg 1, \qquad P_3 = (p_2 + p_3 + 1) \gg 1$
7, 5 (p side: 7 q side: 5)	$g_j = 58 - j * 13$, *can also be described as* $\boldsymbol{g} = \{58,45,32,19,6\}$ $f_i = 59 - i * 9$, *can also be described as* $\boldsymbol{f} = \{59,50,41,32,23,14,5\}$ $Middle7{,}5 = (2 * (p_o + q_o + p_1 + q_1) + q_2 + p_2 + q_3 + p_3 + q_4 + p_4 + q_5 + p_5 + 8) \gg 4$ $Q_5 = (q_4 + q_5 + 1) \gg 1, \qquad P_7 = (p_6 + p_7 + 1) \gg 1$
5, 7 (p side: 5 q side: 7)	$g_j = 59 - j * 9$, *can also be described as* $\boldsymbol{g} = \{59,50,41,32,23,14,5\}$ $f_i = 58 - i * 13$, *can also be described as* $\boldsymbol{f} = \{58,45,32,19,6\}$ $Middle5{,}7 = (2 * (q_o + p_o + p_1 + q_1) + q_2 + p_2 + q_3 + p_3 + q_4 + p_4 + q_5 + p_5 + 8) \gg 4$ $Q_7 = (q_6 + q_7 + 1) \gg 1, \qquad P_5 = (p_4 + p_5 + 1) \gg 1$
5, 5 (p side: 5 q side: 5)	$g_j = 58 - j * 13$, *can also be described as* $\boldsymbol{g} = \{58,45,32,19,6\}$ $f_i = 58 - i * 13$, *can also be described as* $\boldsymbol{f} = \{58,45,32,19,6\}$ $Middle5{,}5 = (2 * (q_o + p_o + p_1 + q_1 + q_2 + p_2) + q_3 + p_3 + q_4 + p_4 + 8) \gg 4$ $Q_5 = (q_4 + q_5 + 1) \gg 1, \qquad P_5 = (p_4 + p_5 + 1) \gg 1$
5, 3 (p side: 5 q side: 3)	$g_j = 53 - j * 21$, *can also be described as* $\boldsymbol{g} = \{53,32,11\}$ $f_i = 58 - i * 13$, *can also be described as* $\boldsymbol{f} = \{58,45,32,19,6\}$ $Middle5{,}3 = (q_o + p_o + p_1 + q_1 + q_2 + p_2 + q_3 + p_3 + 4) \gg 3$ $Q_3 = (q_2 + q_3 + 1) \gg 1, \qquad P_5 = (p_4 + p_5 + 1) \gg 1$

【表６】

3, 5 (p side: 3 q side: 5)	$g_j = 58 - j * 13$, *can also be described as* $\boldsymbol{g} = \{58,45,32,19,6\}$ $f_i = 53 - i * 21$, *can also be described as* $\boldsymbol{f} = \{53,32,11\}$ $Middle3{,}5 = (q_o + p_o + p_1 + q_1 + q_2 + p_2 + q_3 + p_3 + 4) \gg 3$ $Q_5 = (q_4 + q_5 + 1) \gg 1, \qquad P_3 = (p_2 + p_3 + 1) \gg 1$

【０１４４】
２．９．５　クロマの非ブロック化制御
【０１４５】
　クロマの強いフィルタは、ブロックの境界の両側に用いられる。ここで、クロマフィルタは、クロマエッジの両側が８（クロマ位置）以上である場合に選択され、のための３つの条件付きで、１番目のものが、境界強度ならびに大ブロックのためのものである、という決定が満たされる。提案されたフィルタは、クロマサンプルドメインにおいて、ブロックのエッジに直交するブロックの幅または高さが８以上である場合に適用できる。第２および第３のものは、基本的にはＨＥＶＣ輝度の非ブロック化の決定と同じであり、それぞれオン／オフ決定および強いフィルタの決定となっている。
第１の決定において、表２に示すように、クロマフィルタリングのために境界強度（ｂＳ）が修正される。表２の条件を順次チェックする。条件が満たされている場合、残りの優先順位の低い条件はスキップされる。
【０１４６】
　大ブロックの境界が検出された場合、ｂＳが２に等しいか、またはｂＳが１に等しいとき、クロマ非ブロック化が行われる。
【０１４７】
　第２および第３の条件は、基本的には、以下のように、ＨＥＶＣ輝度の強いフィルタの決定と同様である。
【０１４８】
　第２の条件において、
　その後、ＨＥＶＣ輝度非ブロック化と同様にｄを導出する。
　　第２の条件は、ｄがβより小さい場合、ＴＲＵＥとなる。
【０１４９】
　第３の条件において、ＳｔｒｏｎｇＦｉｌｔｅｒＣｏｎｄｉｔｉｏｎは、以下のように導出される。
ｄｐｑはＨＥＶＣと同様に導出される。
sp_3＝Ａｂｓ（p_3－p_0）、ＨＥＶＣと同様に導出される
sq_3＝Ａｂｓ（q_0－q_3）は、ＨＥＶＣと同様に導出される
【０１５０】
　ＨＥＶＣ設計におけるように、ＳｔｒｏｎｇＦｉｌｔｅｒＣｏｎｄｉｔｉｏｎ＝（ｄｐｑが（β＞＞２）未満であり、sp_3＋sq_3が（β＞＞３）未満であり、Ａｂｓ（p_0－q_0）が（５＊t_C＋１）＞＞１未満である。）
【０１５１】
２．９．６　クロマ用の強い非ブロック化フィルタ
【０１５２】
　以下のようなクロマ用の強い非ブロック化フィルタが定義される。
p_2’＝（３＊p_3＋２＊p_2＋p_1＋p_0＋q_0＋４）＞＞３
p_1’＝（２＊p_3＋p_2＋２＊p_1＋p_0＋q_0＋q_1＋４）＞＞３
p_0’＝（p_3＋p_2＋p_1＋２＊p_0＋q_0＋q_1＋q_2＋４）＞＞３
【０１５３】
　提案されたクロマフィルタは、４×４クロマサンプルグリッドに対して非ブロック化を行う。
【０１５４】
２．９．７　位置依存クリッピング
【０１５５】
　位置依存クリッピングｔｃＰＤは、境界で７個、５個、および３個のサンプルを修正する強く長いフィルタを含む輝度フィルタリング処理の出力サンプルに適用される。量子化誤差分布を仮定して、より高い量子化ノイズを有すると予想されるサンプルに対して、クリッピング値を増加させることが提案され、よって、再構成されたサンプル値のＴＲＵＥ

のサンプル値からのより高い偏差を有することが予想される。

【０１５６】

非対称フィルタでフィルタリングされた各ＰまたはＱ境界について、章２．９．２における意思決定処理の結果に基づいて、位置依存閾値テーブルが、副情報としてデコーダに提供される２つのテーブル（即ち、Ｔｃ７およびＴｃ３を以下にまとめる）から選択される。

Ｔｃ７＝｛６，５，４，３，２，１，１｝；

Ｔｃ３＝｛６，４，２｝；

ｔｃＰＤ＝（Ｓｐ＝＝３）？Ｔｃ３：Ｔｃ７；

ｔｃＱＤ＝（Ｓｑ＝＝３）？Ｔｃ３：Ｔｃ７；

【０１５７】

短い対称フィルタでフィルタリングされるＰまたはＱ境界に対しては、より小さい位置依存閾値が適用される。

Ｔｃ３＝｛３，２，１｝；

【０１５８】

閾値を定義した後、ｔｃＰおよびｔｃＱクリッピング値に従って、フィルタリングされたｐ’$_i$およびｑ’$_i$サンプル値をクリッピングする。

ｐ’$_i$’＝Ｃｌｉｐ３（ｐ’$_i$＋ｔｃＰ$_i$，ｐ’$_i$－ｔｃＰ$_i$，ｐ’$_i$）；

ｑ’$_j$’＝Ｃｌｉｐ３（ｑ’$_j$＋ｔｃＱ$_j$，ｑ’$_j$－ｔｃＱ$_j$，ｑ’$_j$）；

ここで、ｐ’$_i$およびｑ’$_i$はフィルタリングされたサンプル値であり、ｐ’$_i$’およびｑ’$_j$’はクリッピング後の出力サンプル値であり、ｔｃＰ$_i$　ｔｃＰ$_i$はＶＶＣ　ｔｃパラメータ、ｔｃＰＤおよびｔｃＱＤから導出されるクリッピング閾値である。関数Ｃｌｉｐ３は、ＶＶＣに規定されているような、クリッピング関数である。

【０１５９】

２．９．８　サブブロックの非ブロック化調整

【０１６０】

両方のロングフィルタを使用する並列フレンドリな非ブロック化およびサブブロック非ブロック化を可能にするために、ロングフィルタは、ロングフィルタのための輝度制御に示すように、サブブロック非ブロック化（ＡＦＦＩＮＥ、ＡＴＭＶＰ、またはＤＭＶＲ）を使用する側でのサンプルの修正が、最大で５つまでに制限される。さらに、サブブロックの非ブロック化は、ＣＵまたは暗黙のＴＵ境界に近い８×８グリッド上のサブブロック境界の修正が、各側において最大２つまでのサンプルに制限されるように調整される。

【０１６１】

以下は、ＣＵ境界と整列されていないサブブロック境界に適用される。

```
Ｉｆ（ｍｏｄｅ　ｂｌｏｃｋ　Ｑ＝＝ＳＵＢＢＬＯＣＫＭＯＤＥ　＆＆　ｅｄｇｅ　！＝
０）｛
    ｉｆ（！（ｉｍｐｌｉｃｉｔＴＵ　＆＆　（ｅｄｇｅ＝＝（６４／４）））
        ｉｆ　（ｅｄｇｅ＝＝２　｜｜　ｅｄｇｅ＝＝（ｏｒｔｈｏｇｏｎａｌＬｅｎｇ
ｔｈ－２）　｜｜　ｅｄｇｅ＝＝（５６／４）　｜｜　ｅｄｇｅ＝＝（７２／４））
            Ｓｐ＝Ｓｑ＝２；
        ｅｌｓｅ
            Ｓｐ＝Ｓｑ＝３；
    ｅｌｓｅ
        Ｓｐ＝Ｓｑ＝ｂＳｉｄｅＱｉｓＬａｒｇｅＢｌｋ？５：３
｝
```

【０１６２】

この場合、０に等しいエッジはＣＵ境界に対応し、２に等しいかまたは直交長さ－２に等しいエッジは、ＣＵ境界からのサブブロックの境界８サンプルに対応する。ここで、ＴＵの暗黙的分割が使用される場合、暗黙的ＴＵはＴＲＵＥである。

【０１６３】

２．９．９　輝度／クロマ用の４ＣＴＵ／２ＣＴＵ行バッファへの制限

【０１６４】

水平エッジがＣＴＵ境界と整列している場合、水平エッジのフィルタリングは、輝度の場合、Ｓｐ＝３、クロマの場合、Ｓｐ＝１、Ｓｑ＝１に制限する。

【０１６５】

２．１０　イントラ予測モードと組み合わせたパレットモード（ＰＣＩＰ）

【０１６６】

ＪＶＥＴ－Ｍ００５１では、イントラ予測と組み合わせたパレットモードが提案されている。このスキームでは、デコーダは最初にイントラ予測方法に基づいて予測ブロックを導出する。次に、デコーダはパレットとインデックスマップを復号化する。デコーダは、復号化パレット情報を使用して、予測ブロックを微調整し、ブロックを再構成する。

【０１６７】

図９にパレット情報とイントラ予測情報を組み合わせた例を示す。最初に、デコーダは予測ブロックを生成する（画素値Ａ０～Ａ１５）。そして、デコーダはパレットモードでインデックスマップを復号化する。ブロックを再構成するために、復号化されたインデックスが「０」である場合、図２－６に示すように、対応する画素は「Ｐ」としてマークされる。「Ｐ」とマークされた画素は、イントラ予測ブロックの画素値によって再構成される。それ以外の場合は、例えば、Ｃ０、Ｃ１などのパレットカラーで画素が再構成される。

【０１６８】

２．１１　複合パレットモード（ＣＰＭ）

【０１６９】

ＪＶＥＴ－Ｎ０２５９と同様に、複合パレットモードのサンプルは、現在のインデックスが０のときにＩＢＣ予測で同一位置に配置されたサンプルをコピーすることで再構成できる。それ以外の場合、現在のサンプルはパレットの色によって再構成される。図１０は、提案されたパレットモードを示している。

【０１７０】

複合パレットモードでは、輝度ブロックで、最初に利用可能なマージ候補を使用してＩＢＣ予測が生成されるため、ＩＢＣ予測の動き情報を信号通知する必要はない。クロマブロックでは、ＣＰＭの動き補償はＶＴＭ４の動き補償と同じである。

【０１７１】

さらに、エスケープ画素の符号化も修正される。元のサンプルと同一位置に配置されたＩＢＣ予測サンプルの間の量子化残差が信号通知される。デコーダでは、提案されたパレットモードのすべてのエスケープ画素が、復号化された残差とＩＢＣ予測を組み合わせて再構成される。

【０１７２】

２．１２　ＪＶＥＴ－Ｎ０１８５における６７個のイントラ予測モードを有するイントラモード符号化

【０１７３】

自然映像に表される任意のエッジ方向をキャプチャするために、ＶＴＭ４における指向性イントラモードの数は、ＨＥＶＣで使用されるように、３３から６５に拡張される。ＨＥＶＣにない新しい指向性モードは、図１１に赤い点線の矢印で示されており、平面モードと直流モードは同じままである。これらのより密度の高い指向性イントラ予測モードは、すべてのブロックサイズ、および輝度およびクロマイントラ予測の両方に適用される。

【０１７４】

ＭＲＬ符号化ツールおよびＩＳＰ符号化ツールの適用の有無にかかわらず、イントラブロックに対して統一された６－ＭＰＭリストが提案される。ＭＰＭリストは、図１２に示すように、ＶＴＭ４．０などの場合、左上の近傍のブロックのイントラモードに基づいて構成される。

【０１７５】

ここで、左側のモードをＬｅｆｔとし、上記ブロックのモードをＡｂｏｖｅとすると、

統合ＭＰＭリストは、以下のステップで順に構成される。
－　近傍のブロックのイントラ予測モードが無効である場合、そのイントラモードはデフォルトで平面に設定される。
－　ＬｅｆｔとＡｂｏｖｅが同じかつ両方ともに角度がある場合、
　ｏ　ＭＰＭリスト→｛Ｐｌａｎａｒ、Ｌｅｆｔ、Ｌｅｆｔ－１、Ｌｅｆｔ＋１、ＤＣ、Ｌｅｆｔ－２｝
－　ＬｅｆｔとＡｂｏｖｅが異なりかつ両方ともに角度がある場合、
　ｏ　ＭａｘモードをＬｅｆｔとＡｂｏｖｅでより大きいモードに設定する。
　ｏ　ＬｅｆｔモードとＡｂｏｖｅモードとの差が２～６２の範囲内にある場合
　　■　ＭＰＭ　ｌｉｓｔ→｛Ｐｌａｎａｒ，Ｌｅｆｔ，Ａｂｏｖｅ，ＤＣ，Ｍａｘ－１，Ｍａｘ＋１｝
　ｏ　あるいは、
　　■　ＭＰＭリスト→｛Ｐｌａｎａｒ、Ｌｅｆｔ、Ａｂｏｖｅ、ＤＣ、Ｍａｘ－２、Ｍａｘ＋２｝
－　ＬｅｆｔとＡｂｏｖｅが異なり、ＬｅｆｔおよびＡｂｏｖｅの一方が角度モードであり、他方が非角度モードである場合、
　ｏ　ＭａｘモードをＬｅｆｔとＡｂｏｖｅでより大きいモードに設定する。
　ｏ　ＭＰＭリスト→｛Ｐｌａｎａｒ、Ｍａｘ、ＤＣ、Ｍａｘ－１、Ｍａｘ＋１、Ｍａｘ－２｝
－モードＬｅｆｔおよびＡｂｏｖｅの両方が非角度モードである場合、
　ｏ　ＭＰＭリスト→｛Ｐｌａｎａｒ、ＤＣ、Ｖ、Ｈ、Ｖ－４、Ｖ＋４｝

【０１７６】
なお、モードインデックス（例えば、Ｍａｘ＋２）が範囲［０、６６］を超えた場合、他の有効なモードインデックスに置き換えられてもよい。

【０１７７】
第１のＭＰＭ候補、すなわち、Ｐｌａｎａｒモードは、残りのＭＰＭ候補とは別個に信号通知される。

【０１７８】
より詳細な構文、意味論、及び復号化処理を以下に説明する。：

【０１７９】

【表７】

７．３．６．５　符号化ユニット構文

coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	Descriptor
if(tile_group_type != I || sps_ibc_enabled_flag) {	
if(treeType != DUAL_TREE_CHROMA)	
cu_skip_flag[x0][y0]	ae(v)
if(cu_skip_flag[x0][y0] = = 0 && tile_group_type != I)	
pred_mode_flag	ae(v)
if(((tile_group_type = = I && cu_skip_flag[x0][y0] = =0) || (tile_group_type != I && CuPredMode[x0][y0] != MODE_INTRA)) && sps_ibc_enabled_flag)	
pred_mode_ibc_flag	ae(v)
}	
if(CuPredMode[x0][y0] = = MODE_INTRA) {	
if(sps_pcm_enabled_flag && cbWidth >= MinIpcmCbSizeY && cbWidth <= MaxIpcmCbSizeY && cbHeight >= MinIpcmCbSizeY && cbHeight <= MaxIpcmCbSizeY)	
pcm_flag[x0][y0]	ae(v)
if(pcm_flag[x0][y0]) {	
while(!byte_aligned( ))	
pcm_alignment_zero_bit	f(1)
pcm_sample(cbWidth, cbHeight, treeType)	
} else {	
if(treeType = = SINGLE_TREE || treeType = = DUAL_TREE_LUMA) {	
if((y0 % CtbSizeY) > 0)	
intra_luma_ref_idx[x0][y0]	ae(v)
if (intra_luma_ref_idx[x0][y0] = = 0 && (cbWidth <= MaxTbSizeY || cbHeight <= MaxTbSizeY) && (cbWidth * cbHeight > MinTbSizeY * MinTbSizeY))	
intra_subpartitions_mode_flag[x0][y0]	ae(v)
if(intra_subpartitions_mode_flag[x0][y0] = = 1 && cbWidth <= MaxTbSizeY && cbHeight <= MaxTbSizeY)	
intra_subpartitions_split_flag[x0][y0]	ae(v)

【表８】

if(intra_luma_ref_idx[x0][y0] = = 0 && intra_subpartitions_mode_flag[x0][y0] = = 0)	
intra_luma_mpm_flag[x0][y0]	ae(v)
if(intra_luma_mpm_flag[x0][y0]) {	
if(intra_luma_ref_idx[x0][y0] = = 0)	
intra_luma_not_planar_flag[x0][y0]	ae(v)
if(intra_luma_not_planar_flag[x0][y0])	
intra_luma_mpm_idx[x0][y0]	ae(v)
} else	
intra_luma_mpm_remainder[x0][y0]	ae(v)
}	
if(treeType = = SINGLE_TREE || treeType = = DUAL_TREE_CHROMA)	
intra_chroma_pred_mode[x0][y0]	ae(v)
}	
} else if(treeType != DUAL_TREE_CHROMA) { /* MODE_INTER or MODE_IBC */	
...	
}	
if(!pcm_flag[x0][y0]) {	
if(CuPredMode[x0][y0] != MODE_INTRA && merge_flag[x0][y0] = = 0)	
cu_cbf	ae(v)

【表９】

if(cu_cbf) {	
if(CuPredMode[x0][y0] == MODE_INTER && sps_sbt_enabled_flag && !ciip_flag[x0][y0]) {	
if(cbWidth <= MaxSbtSize && cbHeight <= MaxSbtSize) {	
allowSbtVerH = cbWidth >= 8	
allowSbtVerQ = cbWidth >= 16	
allowSbtHorH = cbHeight >= 8	
allowSbtHorQ = cbHeight >= 16	
if(allowSbtVerH || allowSbtHorH || allowSbtVerQ || allowSbtHorQ)	
cu_sbt_flag	
}	ae(v)
if(cu_sbt_flag) {	
if((allowSbtVerH || allowSbtHorH) && (allowSbtVerQ || allowSbtHorQ))	
cu_sbt_quad_flag	
if((cu_sbt_quad_flag && allowSbtVerQ && allowSbtHorQ) || (!cu_sbt_quad_flag && allowSbtVerH && allowSbtHorH))	ae(v)
cu_sbt_horizontal_flag	
cu_sbt_pos_flag	ae(v)
}	ae(v)
}	
transform_tree(x0, y0, cbWidth, cbHeight, treeType)	
}	
}	
}	

【０１８０】
構文要素ｉｎｔｒａ＿ｌｕｍａ＿ｍｐｍ＿ｆｌａｇ［ｘ０］［ｙ０］、ｉｎｔｒａ＿ｌｕｍａ＿ｎｏｔ＿ｐｌａｎａｒ＿ｆｌａｇ［ｘ０］［ｙ０］、ｉｎｔｒａ＿ｌｕｍａ＿ｍｐｍ＿ｉｄｘ［ｘ０］［ｙ０］およびｉｎｔｒａ＿ｌｕｍａ＿ｍｐｍ＿ｒｅｍａｉｎｄｅｒ［ｘ０］［ｙ０］は、輝度サンプルのイントラ予測モードを指定する。配列インデックスｘ０，ｙ０は、ピクチャの左上輝度サンプルに対する、考慮される符号化ブロックの左上輝度サンプルの位置（ｘ０，ｙ０）を指定する。ｉｎｔｒａ＿ｌｕｍａ＿ｍｐｍ＿ｆｌａｇ［ｘ０］［ｙ０］が１に等しい場合、第８．４．２項に従って、隣接するイントラ予測符号化ユニットから前記イントラ予測モードを推論する。
【０１８１】
ｉｎｔｒａ＿ｌｕｍａ＿ｍｐｍ＿ｆｌａｇ［ｘ０］［ｙ０］が存在しない（例えば、ＩＳＰが有効化されている、またはＭＲＬが有効化されている（参照インデックス＞０を有する））場合、それは１に等しいと推論される。
【０１８２】

ｉｎｔｒａ＿ｌｕｍａ＿ｎｏｔ＿ｐｌａｎａｒ＿ｆｌａｇ［ｘ０］［ｙ０］が存在しない場合（例えば、ＭＲＬが使用可能）、それは１に等しいと推測される。

【０１８３】

８．４．２　輝度イントラ予測モードのための導出処理

【０１８４】

この処理への入力は以下の通りである。

－　現在のピクチャの左上の輝度サンプルに対する現在の輝度符号化ブロックの左上のサンプルを規定する輝度位置（ｘＣｂ，ｙＣｂ）

－　輝度サンプルにおける現在の符号化ブロックの幅を規定する変数ｃｂＷｉｄｔｈ、

－　輝度サンプルにおける現在の符号化ブロックの高さを規定する変数ｃｂＨｅｉｇｈｔ。

【０１８５】

この処理において、輝度イントラ予測モードＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］が導出される。

表８－１に、イントラ予測モードＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］の値とその関連名称を示す。

【０１８６】

【表１０】

表８－１：イントラ予測モードの指定とその関連名称の仕様

イントラ予測モード	関連名称
0	INTRA_PLANAR
1	INTRA_DC
2..66	INTRA_ANGULAR2..INTRA_ANGULAR66
81..83	INTRA_LT_CCLM, INTRA_L_CCLM, INTRA_T_CCLM

【０１８７】

注－：イントラ予測モードＩＮＴＲＡ＿ＬＴ＿ＣＣＬＭ、ＩＮＴＲＡ＿Ｌ＿ＣＣＬＭ、ＩＮＴＲＡ＿Ｔ＿ＣＣＬＭは、クロマ成分にのみ適用可能である。

【０１８８】

ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］は、以下のように導出される。

【０１８９】

－　ｉｎｔｒａ＿ｌｕｍａ＿ｎｏｔ＿ｐｌａｎａｒ＿ｆｌａｇ［ｘＣｂ］［ｙＣｂ］が１に等しい場合、以下の順序付けられたステップである。

【０１９０】

１．近傍位置（ｘＮｂＡ、ｙＮｂＡ）および（ｘＮｂＢ、ｙＮｂＢ）は、それぞれ、（ｘＣｂ－１，ｙＣｂ＋ｃｂＨｅｉｇｈｔ－１）および（ｘＣｂ＋ｃｂＷｉｄｔｈ－１，ｙＣｂ－１）に等しく設定される。

【０１９１】

２．ＸをＡまたはＢのいずれかに置き換える場合、変数ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＸは、以下のように導出される。

－　６．４．Ｘ項［Ｅｄ．（ＢＢ）：近傍ブロックの可用性導出処理ｔｂｄ］で規定されるブロックの可用性導出処理は、入力として、（ｘＣｂ，ｙＣｂ）に等しく設定された位置（ｘＣｕｒｒ，ｙＣｕｒｒ）と、（ｘＮｂＸ，ｙＮｂＸ）に等しく設定した近傍位置（ｘＮｂＹ，ｙＮｂＹ）で呼び出され、出力をａｖａｉｌａｂｌｅＸに割り当てる。

－　候補イントラ予測モードｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＸは、以下のように導出される。

－ 以下の条件の１つ以上がＴＲＵＥである場合、ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＸをＩＮＴＲＡ＿ＰＬＡＮＡＲに等しく設定する。
－ 変数ａｖａｉｌａｂｌｅＸはＦＡＬＳＥに等しい。
－ ＣｕＰｒｅｄＭｏｄｅ［ｘＮｂＸ］［ｙＮｂＸ］はＭＯＤＥ＿ＩＮＴＲＡに等しくなく、かつｃｉｉｐ＿ｆｌａｇ［ｘＮｂＸ］［ｙＮｂＸ］は１に等しくない。
－ ｐｃｍ＿ｆｌａｇ［ｘＮｂＸ］［ｙＮｂＸ］は１に等しい。
－ ＸがＢに等しく、ｙＣｂ＿１が（（ｙＣｂ＞＞ＣｔｂＬｏｇ２ＳｉｚｅＹ）＜＜ＣｔｂＬｏｇ２ＳｉｚｅＹ）未満である。
－ そうでない場合、ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＸをＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＮｂＸ］［ｙＮｂＸ］に等しく設定する。

【０１９２】

３．ｘ＝０．．４の場合のｃａｎｄＭｏｄｅＬｉｓｔ［ｘ］は、以下のように導出される。

－ ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＢがｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡに等しく、ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡがＩＮＴＲＡ＿ＤＣよりも大きい場合、ｃａｎｄＭｏｄｅＬｉｓｔ［ｘ］（ｘ＝０．．４）は、以下のように導出される。

ｃａｎｄＭｏｄｅＬｉｓｔ［０］＝ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡ　（８－１０）

ｃａｎｄＭｏｄｅＬｉｓｔ［１］＝２＋（（ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡ＋６１）％６４）　（８－１２）

ｃａｎｄＭｏｄｅＬｉｓｔ［２］＝２＋（（ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡ－１）％６４）　（８－１３）

ｃａｎｄＭｏｄｅＬｉｓｔ［３］＝ＩＮＴＲＡ＿ＤＣ　（８－１１）

ｃａｎｄＭｏｄｅＬｉｓｔ［４］＝２＋（（ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡ＋６０）％６４）　（８－１４）

－ ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＢがｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡに等しくなく、ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡまたはｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＢがＩＮＴＲＡ＿ＤＣよりも大きい場合、以下が適用される。
－ 変数ｍｉｎＡＢおよびｍａｘＡＢは、以下のように導出される。

ｍｉｎＡＢ＝Ｍｉｎ（ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡ，ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＢ）　（８－２４）

ｍａｘＡＢ＝Ｍａｘ（ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡ，ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＢ）　（８－２５）

－ ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡ及びｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＢの両方がＩＮＴＲＡ＿ＤＣよりも大きい場合、ｘ＝０．．４の場合のｃａｎｄＭｏｄｅＬｉｓｔ［ｘ］は、以下のように導出される。

ｃａｎｄＭｏｄｅＬｉｓｔ［０］＝ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡ　（８－２７）

ｃａｎｄＭｏｄｅＬｉｓｔ［１］＝ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＢ　（８－２９）

ｃａｎｄＭｏｄｅＬｉｓｔ［２］＝ＩＮＴＲＡ＿ＤＣ　（８－２９）

－ ｍａｘＡＢ－ｍｉｎＡＢが２～６２の範囲内にある場合、以下が適用される。

ｃａｎｄＭｏｄｅＬｉｓｔ［３］＝２＋（（ｍａｘＡＢ＋６１）％６４）　（８－３０）

ｃａｎｄＭｏｄｅＬｉｓｔ［４］＝２＋（（ｍａｘＡＢ－１）％６４）　（８－３１）

－ そうでない場合、以下が適用される。

ｃａｎｄＭｏｄｅＬｉｓｔ［３］＝２＋（（ｍａｘＡＢ＋６０）％６４）　（８－３２）

ｃａｎｄＭｏｄｅＬｉｓｔ［４］＝２＋（（ｍａｘＡＢ）％６４）　（８－３３

)

－ そうでない場合（ｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＡまたはｃａｎｄＩｎｔｒａＰｒｅｄＭｏｄｅＢ＞ＩＮＴＲＡ＿ＤＣ）、ｘ＝０．．４のｃａｎｄＭｏｄｅＬｉｓｔ［ｘ］は、以下のように導出される。

ｃａｎｄＭｏｄｅＬｉｓｔ［０］＝ｍａｘＡＢ　（８－６５）

ｃａｎｄＭｏｄｅＬｉｓｔ［１］＝ＩＮＴＲＡ＿ＤＣ　（８－６６）

ｃａｎｄＭｏｄｅＬｉｓｔ［２］＝２＋（（ｍａｘＡＢ＋６１）％６４）　（８－６６）

ｃａｎｄＭｏｄｅＬｉｓｔ［３］＝２＋（（ｍａｘＡＢ－１）％６４）　（８－６７）

ｃａｎｄＭｏｄｅＬｉｓｔ［４］＝２＋（（ｍａｘＡＢ＋６０）％６４）　（８－６８）

－ そうでない場合、以下が適用される。

ｃａｎｄＭｏｄｅＬｉｓｔ［０］＝ＩＮＴＲＡ＿ＤＣ　（８－７１）

ｃａｎｄＭｏｄｅＬｉｓｔ［１］＝ＩＮＴＲＡ＿ＡＮＧＵＬＡＲ５０　（８－７２）

ｃａｎｄＭｏｄｅＬｉｓｔ［２］＝ＩＮＴＲＡ＿ＡＮＧＵＬＡＲ１８　（８－７３）

ｃａｎｄＭｏｄｅＬｉｓｔ［３］＝ＩＮＴＲＡ＿ＡＮＧＵＬＡＲ４６　（８－７４）

ｃａｎｄＭｏｄｅＬｉｓｔ［４］＝ＩＮＴＲＡ＿ＡＮＧＵＬＡＲ５４　（８－７５）

【０１９３】

４．ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］は、以下の順のステップを適用することによって導出される。

－ ｉｎｔｒａ＿ｌｕｍａ＿ｍｐｍ＿ｆｌａｇ［ｘＣｂ］［ｙＣｂ］が１に等しい場合、ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］は、ｃａｎｄＭｏｄｅＬｉｓｔ［ｉｎｔｒａ＿ｌｕｍａ＿ｍｐｍ＿ｉｄｘ［ｘＣｂ］［ｙＣｂ］］と等しく設定される。

－ そうでない場合、ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］は以下の順のステップを適用することにより、導出される。

１．ｃａｎｄＭｏｄｅＬｉｓｔ［ｉ］がｃａｎｄＭｏｄｅＬｉｓｔ［ｊ］よりも大きい（ｉ＝０．．３、各ｉ，ｊ＝（ｉ＋１）．．４）場合、両方の値は、以下のようにスワップされる。

（ｃａｎｄＭｏｄｅＬｉｓｔ［ｉ］，ｃａｎｄＭｏｄｅＬｉｓｔ［ｊ］）＝Ｓｗａｐ（ｃａｎｄＭｏｄｅＬｉｓｔ［ｉ］，ｃａｎｄＭｏｄｅＬｉｓｔ［ｊ］）　（８－９４）

２．ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］は以下の順のステップで導出される。

ｉ．ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］はｉｎｔｒａ＿ｌｕｍａ＿ｍｐｍ＿ｒｅｍａｉｎｄｅｒ［ｘＣｂ］［ｙＣｂ］と等しく設定される。

ｉｉ．ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］の値が１増加する。

ｉｉｉ．ｉ＝０～４の場合、ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］がｃａｎｄＭｏｄｅＬｉｓｔ［ｉ］以上である場合、ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］の値が１増加する。

【０１９４】

－ そうでない場合（ｉｎｔｒａ＿ｌｕｍａ＿ｎｏｔ＿ｐｌａｎａｒ＿ｆｌａｇ［ｘＣｂ］［ｙＣｂ］は０と等しい）、ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］はＩＮＴＲＡ＿ＰＬＡＮＡＲと等しく設定される。

【０１９５】

ｘ＝ｘＣｂ．．ｘＣｂ＋ｃｂＷｉｄｔｈ－１およびｙ＝ｙＣｂ．．ｙＣｂ＋ｃｂＨｅｉｇｈｔ－１の場合、変数ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘ］［ｙ］は、ＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］と等しく設定される。

【０１９６】

クロマイントラモード符号化の場合、クロマイントラモード符号化の場合、合計８つのイントラモードが許可される。これらのモードには、５つの伝統的なイントラモードと６つ

の構成要素共通の線形モデルモードが含まれる（ＣＣＬＭ、ＬＭ＿ＡおよびＬＭ＿Ｌ）。クロマモード信号通知および導出処理を表３に示す。クロマモード符号化は、対応する輝度ブロックのイントラ予測モードに直接依存する。Ｉスライスにおいて、輝度成分とクロマ成分に対するブロック分割構造の分離が有効化されているため、１つのクロマブロックは複数の輝度ブロックに対応してもよい。よって、クロマＤＭモードの場合、現在のクロマブロックの中心位置を含む、対応する輝度ブロックのイントラ予測モードは直接継承される。

【０１９７】

【表１１】

表４：ｃｃｌｍ＿が有効な時の輝度モードからのクロマ予測モードの導出

クロマ予測モード	対応する輝度イントラ予測モード				
	0	50	18	1	X (0<=X<=66)
0	66	0	0	0	0
1	50	66	50	50	50
2	18	18	66	18	18
3	1	1	1	66	1
4	81	81	81	81	81
5	82	82	82	82	82
6	83	83	83	83	83
7	0	50	18	1	X

【０１９８】

２．１３　履歴に基づくマージ候補の導出

【０１９９】

履歴に基づくＭＶＰ（ＨＭＶＰ）マージ候補は、空間ＭＶＰおよびＴＭＶＰの後にマージリストに追加される。この方法では、あらかじめ符号化されたブロックの動き情報がテーブルに保存され、現在のＣＵのＭＶＰとして使用される。符号化／復号化処理中、複数のＨＭＶＰ候補を有するテーブルが維持される。新しいＣＴＵ行が検出されると、テーブルはリセット（空）される。非サブブロックインター符号化ＣＵがある場合は常に、関する動き情報が新しいＨＭＶＰ候補としてテーブルの最後のエントリに追加される。

【０２００】

ＶＴＭ５では、ＨＭＶＰテーブルサイズＳは５に設定されている。これは、最大５つの履歴に基づくＭＶＰ（ＨＭＶＰ）候補をテーブルに追加できることを示している。新しい動き候補をテーブルに挿入する際には、制約付き先入れ先出し（ＦＩＦＯ）規則が使用され、この規則では、冗長性チェックが最初に適用され、テーブルに同一のＨＭＶＰがあるかどうかが確認される。見つかった場合、同一のＨＭＶＰがテーブルから削除され、その後、すべてのＨＭＶＰ候補が前方に移動され、ＨＭＶＰ候補は、マージ候補リスト構築処理において使用され得る。表内の最新のいくつかのＨＭＶＰ候補が順番にチェックされ、ＴＭＶＰ候補の後に候補リストに挿入される。冗長性チェックは、ＨＭＶＰ候補から空間的マージ候補に適用される。

【０２０１】

２．１４　スケーリングリスト

【０２０２】

スケーリングリスト（参照、量子化行列）は、（逆）量子化処理中に適用できる。ＨＥＶＣでは、ユーザ定義のスケーリング値がＰＰＳで信号通知され、可能なＴＢサイズ、色成分、予測タイプ（イントラ／インター）ごとに、独自のスケーリングリストを持つことができる。ただし、４：４：４　ＲＥｘｔクロマフォーマットにのみ使用される３２×３２クロマブロックは除かれる。１６×１６および３２×３２スケーリングリストの場合、スケーリングリストは、ＤＣ周波数位置に対応するエントリに使用される値とともに、必要なサイズに値が繰り返される８×８グリッドの値で指定される。

【０２０３】

２．１５　インター予測のための三角形分割モード（ＴＰＭ）

【０２０４】

ＶＴＭ４において、インター予測のために三角形分割モードがサポートされる。三角形分割モードは、８×８以上であり、スキップまたはマージモードで符号化されるＣＵにのみ適用され、ＭＭＶＤまたはＣＩＩＰモードでは適用されない。これらの条件を満たすＣＵの場合、ＣＵレベルフラグを信号通知し、三角形分割モードが適用されるかどうかを示す。

【０２０５】

このモードを使用する場合（図１４）、対角分割または逆対角分割のいずれかを使用して、１つのＣＵを２つの三角形のパーティションに等分する。ＣＵにおける各三角形のパーティションは、それ自体の動きを使用してインター予測され、各パーティションに対して単一予測のみが許可される。すなわち、各パーティションは、１つの動きベクトル及び１つの参照インデックスを有する。従来の双予測と同様にＣＵごとに２つの動き補償予測しか必要としないようにするために、単一予測動き制約を適用する。３．４．１０．１の処理を用いて構築された単一予測候補リストから、各パーティションの単一予測の動きを導出する。

【０２０６】

ＣＵレベルフラグが、現在のＣＵが三角形分割モードで符号化されていることを示す場合。三角形分割モードを使用する場合、三角形パーティションの方向（対角または逆対角）を示すフラグ、及び２つのマージインデックス（各パーティションに１つ）をさらに信号通知する。各三角形パーティションの各々を予測した後、適応重み付きブレンド処理を使用して、対角または逆対角の縁部に沿ったサンプル値を調整する。これがＣＵ全体の予測信号であり、他の予測モードと同様に、ＣＵ全体に対して変換及び量子化処理を適用する。最後に、２．１６．３で示すように、三角形分割モードを使用して予測されたＣＵの動きフィールドを４×４ユニットで記憶する。

【０２０７】

２．１５．１　単一予測候補リスト構築

【０２０８】

この単一予測候補リストは、５つの単一予測動きベクトル候補からなる。それは、５つの空間的に近傍のブロック（図１５において１～５とラベル付けされる）と、２つの時間的に同一位置にあるブロック（図１５において６～７とラベル付けされる）とを含む７つの近傍のブロックから導出される。７つの近傍のブロックを収集し、下記の順に従って、単一予測候補リストに入れる。まず、単一予測された近傍のブロックの動きベクトルを収集し、単一予測候補リストに入れる。次に、単一予測された近傍のブロックの動きベクトルを収集し、次に、双予測された近傍のブロックについて、Ｌ０動きベクトル（すなわち、双予測ＭＶのＬ０動きベクトル部分）、Ｌ１動きベクトル（すなわち、双予測ＭＶのＬ１動きベクトル部分）、及び双予測ＭＶのＬ０、Ｌ１動きベクトルの平均化された動きベクトルを収集する。候補の数が５未満である場合、リストの末端にゼロの動きベクトルを加える。

【０２０９】

２．１５．２　三角形の分割エッジに沿ったブレンド

【０２１０】

各三角形パーティションをそれ自身の動きで予測した後、２つの予測信号にブレンドを適用し、対角線または逆対角の縁部付近のサンプルを導出する。ブレンド処理において、以下の重みを用いる。

【０２１１】

図１６の例に示すように、輝度に関しては○７／８，６／８，５／８，４／８，３／８，２／８，１／８｝、クロマに関しては｛６／８，４／８，２／８｝である。

【０２１２】

２．１５．３　動きフィールド記憶域

【０２１３】
　三角形分割モードで符号化されたＣＵの動きベクトルは、４×４単位で記憶される。各４×４ユニットの位置に基づいて、１つの予測動きベクトルまたは２つの予測動きベクトルのいずれかが記憶される。Ｍｖ１及びＭｖ２をそれぞれパーティション１及びパーティション２の単一予測動きベクトルとする。図１６に示されるように、非重み付け領域に４×４個のユニットが位置する場合、その４×４個のユニットに対してＭｖ１またはＭｖ２のいずれかを記憶する。そうではなく、４×４ユニットが重み付け領域に位置する場合、双予測動きベクトルを記憶する。以下の処理に従って、Ｍｖ１及びＭｖ２から双予測動きベクトルを導出する。
【０２１４】
　１）　Ｍｖ１及びＭｖ２が異なる参照ピクチャリストから（一方はＬ０から、他方はＬ１から）のものである場合、Ｍｖ１及びＭｖ２を単に組み合わせて双予測動きベクトルを形成する。
【０２１５】
　２）　そうでない場合、Ｍｖ１とＭｖ２が同じリストに由来し、一般性を損なわない場合、両方ともＬ０に由来すると仮定する。この場合、以下である。
【０２１６】
　　ａ）　Ｍｖ２（またはＭｖ１）のいずれかの参照ピクチャがＬ１に現れる場合、そのＭｖ２（またはＭｖ１）は、Ｌ１におけるその参照ピクチャを使用してＬ１動きベクトルに変換される。次に、２つの動きベクトルを組み合わせ、双予測動きベクトルを形成する。
【０２１７】
　　ｂ）　そうでない場合、双予測動きの代わりに、単一予測動きＭｖ１のみが記憶される。
【０２１８】
３　従来技術の技術的制約の例
【０２１９】
　パレットモードまたはその変形例（ＰＣＩＰやＣＰＭなど）で符号化されたブロックでは、いくつかの問題が発生する。
１．パレットモードは、非ブロック化処理で特別に処理する必要がある場合がある。
２．ＣＰＭおよび／またはＰＣＩＰは、符号化効率を高めるためのイントラ予測モードとして扱ってもよい。
３．ＣＰＭで符号化されたブロックのＭＶ／ＢＶは、符号化効率を高めるために、連続するブロックのＭＶ／ＢＶ予測として使用してもよい。
４．ＰＣＩＰで符号化されたブロックのイントラ予測方向は、符号化効率のための連続するブロックのイントラモード予測として使用してもよい。
５．ＶＴＭと同様に、クロマＣＰＭはサブブロックレベルで動きベクトルの導出を行う必要があるため、複雑さは増加する。
６．クロマＰＣＩＰの信号通知は、クロマＩＢＣモードの可用性を考慮していない。
７．非変換モード（すなわち、１つのブロックが変換で符号化されていない、または恒等変換で符号化されている）、例えば、変換スキップ、ｂｄｐｃｍモードが使用される場合、いくつかのＱＰは非効率的でもよい。
８．モードを導出するために使用される対応する輝度ブロックがＩＢＣで符号化される場合、クロマのためにＤＭモードをどのように扱うかは、不明瞭である。
【０２２０】
４　例示的な実施形態
【０２２１】
　以下に列記される実施形態は、一般的な概念を説明するための例であると考えられるべきである。これらの実施形態は狭い意味で解釈されるべきではない。さらに、列挙された技術は、任意の形で組み合わせることができ、例えば、符号化効率を改善し、復号化また

は符号化の複雑さを低減する。

【０２２２】

以下に説明する方法は、現在のパレットモード、ＰＣＩＰ、ＣＰＭ、または残差符号化のためにパレットに依存する可能性のある他の符号化方法に適用され得る。

【０２２３】

１．パレットモードは、非ブロック化処理で既存の符号化モード（たとえば、イントラ、インター、ＩＢＣ）に加えて、別個のモード（たとえば、ＭＯＤＥ＿ＰＬＴ）として扱われ得る。

ａ．一例として、Ｐ側およびＱ側のブロックが両方ともパレットモードで符号化されている場合、境界強度を０に設定してもよい。

ｂ．一例として、一方の側のブロックＡがパレットモードで符号化され、他方の側のブロックＢがパレットモードで符号化されていない場合、境界強度は、ブロックＢの情報にのみ依存してもよい。

ｉ．一例として、ブロックＢはイントラモードで符号化されている。このような場合、境界強度を２に設定してもよい。

ｉｉ．一例において、ブロックＢはＩＢＣモードで符号化されている。このような場合、ブロックＢが少なくとも１つの非ゼロ係数を有する場合、境界強度は１に設定してもよい。

ｉｉｉ．一例において、ブロックＢはＩＢＣモードで符号化されている。このような場合、ブロックＢのすべての係数が０であれば、境界強度を０に設定してもよい。

ｃ．一例として、上記の方法は、ＰＣＩＰ、ＣＰＭまたは他のパレット関連の方法にも適用されてもよい。

【０２２４】

２．ＣＰＭは、非ブロック化処理において（たとえば、境界強度の導出など）、ＩＢＣモードと同様に扱ってもよい。

ａ．あるいは、ＣＰＭは、非ブロック化処理において（たとえば、境界強度の導出など）、パレットモードと同様に扱ってもよい。

ｂ．あるいは、ＣＰＭは、非ブロック化処理において（たとえば、境界強度の導出など）、イントラモードと同様に扱ってもよい。

ｃ．あるいは、非ブロック化フィルタを適用する方法は、ＣＰＭの使用法に依存してもよい。

【０２２５】

３．ＰＣＩＰは、非ブロック化処理において（たとえば、境界強度の導出など）、イントラモードと同様に扱ってもよい。

ａ．あるいは、ＰＣＩＰは、非ブロック化処理において（たとえば、境界強度の導出など）、パレットモードと同様に扱ってもよい。

ｂ．あるいは、非ブロック化フィルタを適用する方法は、ＰＣＩＰの使用法に依存してもよい。

【０２２６】

４．パレット符号化ブロック／ＣＰＭ／ＰＣＩＰの符号化ブロックフラグは、固定値に設定しても、ブロック内のエスケープ画素の存在に依存してもよい。

ａ．一例として、符号化されたブロックフラグを０に設定してもよい。

ｂ．一例として、符号化されたブロックフラグを１に設定してもよい。

ｃ．一例として、現在のブロックにエスケープ画素がない場合、符号化されたブロックフラグを０に設定してもよい。

ｄ．一例として、現在のブロックに少なくとも１つのエスケープ画素がある場合、符号化されたブロックフラグを１に設定してもよい。

【０２２７】

５．パレットモード、および／またはＣＰＭおよび／またはＰＣＩＰで符号化されたサンプルはブロック解除されない場合がある。

ａ．一例として、Ｐ側またはＱ側のブロックがＣＰＭおよび／またはＰＣＩＰで符号化されている場合、Ｐ側および／またはＱ側のサンプルはブロック解除されない可能性がある。

ｂ．一例として、Ｐ側のブロックがＣＰＭおよび／またはＰＣＩＰで符号化され、Ｑ側のブロックがこれらのモードで符号化されていない場合、Ｐ側のサンプルはブロック解除されない可能性があり、逆もまた同様である。

ｃ．一例として、ＣＰＭモードの場合、非ブロック化フィルタは、ＩＢＣ予測からコピーされたサンプルに対してのみ行われ得る。パレットエントリの他のサンプルはブロック解除されない場合がある。

ｄ．一例として、ＰＣＩＰモードの場合、非ブロック化フィルタは、イントラ予測からコピーされたサンプルに対してのみ行われ得る。パレットエントリの他のサンプルはブロック解除されない場合がある。

【０２２８】

６．変換ユニット／予測ユニット／符号化ブロック／領域に対するＣＰＭの使用の指示は、イントラモード、インターモード、ＩＢＣモードおよびパレットモードを含み得る既存の予測モードとは別に符号化され得る。

ａ．一例として、予測モードに関する構文要素（例えば、ｃｕ＿ｓｋｉｐ＿ｆｌａｇ、ｐｒｅｄ＿ｍｏｄｅ＿ｆｌａｇ、ｐｒｅｄ＿ｍｏｄｅ＿ｉｂｃ＿ｆｌａｇ、およびｃｕ＿ｐａｌｅｔｔｅ＿ｆｌａｇ）は、ＣＰＭの使用の指示の前に符号化することができる。

ｉ．代替的に、さらに、ＣＰＭの使用の指示は、予測モードに基づいて条件付きで信号通知されてもよい。

１．一例では、ＣＰＭモードがブロック内コピーモード（すなわち、ＭＯＤＥ＿ＩＢＣ）である場合、ＣＰＭモードの使用の指示の信号通知はスキップされ得る。さらに、現在の予測モードがＭＯＤＥ＿ＩＢＣである場合、ＣＰＭの使用の指示がＦＡＬＳＥであると推測されてもよい。

【０２２９】

７．色成分の信号ＣＰＭおよび／またはＰＣＩＰ関連の構文を通知するかどうかは、パーティション構造および／またはカラーフォーマットおよび／または別個のプレーン符号化に依存してもよい。

ａ．一例として、クロマ成分の場合、単一のツリーが適用されると、ＣＰＭおよび／またはＰＣＩＰ関連の構文の信号通知がスキップされてもよい。

ｂ．あるいは、クロマ成分の場合、デュアルツリーが適用されている場合、ＣＰＭおよび／またはＰＣＩＰ関連の構文が通知されてもよい。

ｃ．あるいは、カラーフォーマットが４：０：０の場合、クロマブロックのＣＰＭおよび／またはＰＣＩＰ関連の構文の信号通知をスキップできる。

ｉ．あるいは、さらに、それらは依然として信号通知されてもよいが、復号化プロセスでは使用されない。

【０２３０】

８．クロマブロックのＣＰＭモードの使用の指示の信号通知は、対応する輝度ブロック内の１つ以上の選択されたブロックの符号化された情報（例えば、予測モード）に基づくことができる。

ａ．一例として、選択された輝度ブロックの少なくとも１つがＩＢＣモードで符号化されている場合、クロマブロックのＣＰＭの指示が信号通知されてもよい。

ｉ．あるいは、選択されたすべての輝度ブロックがＩＢＣモードで符号化されている場合、クロマブロックのＣＰＭの指示が信号通知されてもよい。

ｉｉ．あるいは、選択されたブロックがイントラブロックコピーモードで符号化されていない（すなわち、ＭＯＤＥ＿ＩＢＣである）場合、クロマＣＰＭの使用の指示は、信号通知されなくてもよく、さらに、選択されたブロックの予測モードがＭＯＤＥ＿ＩＢＣでない場合、ＣＰＭの使用の指示は、デフォルト値（例えば、ｆａｌｓｅ）であると推測されてもよい。

ｉｉｉ．あるいは、選択されたすべてのブロックがイントラブロックコピーモード（すなわち、ＭＯＤＥ＿ＩＢＣ）で符号化されていない場合、クロマＣＰＭの使用の指示が信号通知されない場合がある。さらに、選択されたすべてのブロックがＩＢＣモードで符号化されていない場合、ＣＰＭの使用の指示がデフォルト値（たとえば、ｆａｌｓｅ）であると推測されてもよい。

ｂ．一例において、選択された輝度ブロックのサイズは、最小のＣＵ／ＰＵ／ＴＵサイズまたはモーション／モードストレージ用のユニット（４×４など）であり得る。

ｃ．一例として、選択された輝度ブロックは、対応する輝度領域の中央、左上、右上、左下、または右下の位置を含むＣＵ／ＰＵ／ＴＵであってもよい。対応する輝度領域の例を図４－１に示す。

ｉ．一例において、現在のクロマブロックの左上の座標が（ｘ０、ｙ０）であり、現在のクロマブロックの幅と高さがそれぞれｗ０とｈ０であると仮定する。対応する輝度領域の左上のサンプルの座標、対応する輝度領域の幅と高さは、カラーフォーマットに従ってスケーリングしてもよい。

ａ）４：２：０カラーフォーマットの場合、同一位置に配置された輝度領域の左上の座標は（２＊ｘ０、２＊ｙ０）であり、その幅と高さはそれぞれ２＊ｗ０と２＊ｈ０である。

ｂ）４：４：４カラーフォーマットの場合、同一位置に配置された輝度領域の左上の座標は（ｘ０、ｙ０）であり、その幅と高さはそれぞれｗ０とｈ０である。

ｉｉ．同一位置に配置された輝度領域の左上の座標が（ｘ、ｙ）であり、その幅と高さがＷとＨであるとすると、中心位置の座標は次のようになり得る。

ａ）　（ｘ＋Ｗ／２，ｙ＋Ｈ／２）
ｂ）　（ｘ＋Ｗ／２－１，ｙ＋Ｈ／２－１）
ｃ）　（ｘ＋Ｗ／２，ｙ＋Ｈ／２－１）
ｄ）　（ｘ＋Ｗ／２－１，ｙ＋Ｈ／２）

ｉｉｉ．同一位置に配置された輝度領域の左上の座標が（ｘ、ｙ）であり、その幅と高さがＷとＨであるとすると、左上の位置の座標は（ｘ、ｙ）になり得る。

ｉｖ．同一位置に配置された輝度領域の左上の座標が（ｘ、ｙ）であり、その幅と高さがＷとＨであるとすると、右上の位置の座標は次のようになり得る。

ａ）　（ｘ＋Ｗ，ｙ）
ｂ）　（ｘ＋Ｗ－１，ｙ）

ｖ．同一位置に配置された輝度領域の左上の座標が（ｘ、ｙ）であり、その幅と高さがＷとＨであるとすると、左下の位置の座標は次のようになる。

ａ）　（ｘ，ｙ＋Ｈ）
ｂ）　（ｘ，ｙ＋Ｈ－１）

ｖｉ．同一位置に配置された輝度領域の左上の座標が（ｘ、ｙ）であり、その幅と高さがＷとＨであるとすると、右下の位置の座標は次のようになり得る。

ａ）　（ｘ＋Ｗ，ｙ＋Ｈ）
ｂ）　（ｘ＋Ｗ，ｙ＋Ｈ－１）
ｃ）　（ｘ＋Ｗ－１，ｙ＋Ｈ）
ｄ）　（ｘ＋Ｗ－１，ｙ＋Ｈ－１）

【０２３１】

図１３は、対応する輝度領域の例を示している。

【０２３２】

９．クロマＰＣＩＰモードの使用の信号通知は、１つ以上の選択された輝度ブロックの符号化された情報に依存してもよい。

ａ．一例として、対応する輝度ブロック内の選択されたブロックがイントラブロックコピーモード（すなわち、ＭＯＤＥ＿ＩＢＣ）で符号化されている場合、クロマＰＣＩＰの使用の指示が信号通知されない場合がある。さらに、対応する輝度ブロックの選択されたブロックの予測モードがＭＯＤＥ＿ＩＢＣである場合、ＰＣＩＰの使用の指示はｆａｌｓ

ｅであると推測され得る。

ｉ．あるいは、選択されたすべての輝度ブロックがＩＢＣモードで符号化されている場合、クロマブロックのＰＣＩＰの指示が信号通知されない場合がある。

ｉｉ．あるいは、対応する輝度領域内のすべてのサンプルがＩＢＣモードで符号化されている場合、クロマブロックのＰＣＩＰの指示が信号通知されない場合がある。

ｂ．一例において、選択された輝度ブロックのサイズは、最小のＣＵ／ＰＵ／ＴＵサイズまたはモーション／モードストレージ用のユニット（４×４など）であり得る。

ｃ．一例として、選択された輝度ブロックは、図４－１に示されるように、対応する輝度領域の中央、左上、右上、左下、または右下の位置を含むＣＵ／ＰＵ／ＴＵであってもよい。

ｉ．一例において、現在のクロマブロックの左上の座標が（ｘ０、ｙ０）であり、現在のクロマブロックの幅と高さがそれぞれｗ０とｈ０であると仮定する。対応する輝度領域の左上のサンプルの座標、対応する輝度領域の幅と高さは、カラーフォーマットに従ってスケーリングしてもよい。

ａ）４：２：０カラーフォーマットの場合、同一位置に配置された輝度領域の左上の座標は（２＊ｘ０、２＊ｙ０）であり、その幅と高さはそれぞれ２＊ｗ０と２＊ｈ０である。

ｂ）４：４：４カラーフォーマットの場合、同一位置に配置された輝度領域の左上の座標は（ｘ０、ｙ０）であり、その幅と高さはそれぞれｗ０とｈ０である。

ｉｉ．同一位置に配置された輝度領域の左上の座標が（ｘ、ｙ）であり、その幅と高さがＷとＨであるとすると、中心位置の座標は次のようになり得る。

ａ）　（ｘ＋Ｗ／２，ｙ＋Ｈ／２）

ｂ）　（ｘ＋Ｗ／２－１，ｙ＋Ｈ／２－１）

ｃ）　（ｘ＋Ｗ／２，ｙ＋Ｈ／２－１）

ｄ）　（ｘ＋Ｗ／２－１，ｙ＋Ｈ／２）

ｉｉｉ．同一位置に配置された輝度領域の左上の座標が（ｘ、ｙ）であり、その幅と高さがＷとＨであるとすると、左上の位置の座標は（ｘ、ｙ）になり得る。

ｉｖ．同一位置に配置された輝度領域の左上の座標が（ｘ、ｙ）であり、その幅と高さがＷとＨであるとすると、右上の位置の座標は次のようになり得る。

ａ）　（ｘ＋Ｗ，ｙ）

ｂ）　（ｘ＋Ｗ－１，ｙ）

ｖ．同一位置に配置された輝度領域の左上の座標が（ｘ、ｙ）であり、その幅と高さがＷとＨであるとすると、左下の位置の座標は次のようになる。

ａ）　（ｘ，ｙ＋Ｈ）

ｂ）　（ｘ，ｙ＋Ｈ－１）

ｖｉ．同一位置に配置された輝度領域の左上の座標が（ｘ、ｙ）であり、その幅と高さがＷとＨであるとすると、右下の位置の座標は次のようになり得る。

ａ）　（ｘ＋Ｗ，ｙ＋Ｈ）

ｂ）　（ｘ＋Ｗ，ｙ＋Ｈ－１）

ｃ）　（ｘ＋Ｗ－１，ｙ＋Ｈ）

ｄ）　（ｘ＋Ｗ－１，ｙ＋Ｈ－１）

【０２３３】

１０．ＣＰＭモードを、既存の予測モードに加えて、別個の予測モード（たとえば、ＭＯＤＥ＿ＰＬＴ＿ＩＢＣで示される）として扱うことを提案する。

ａ．一例において、予測モードは、イントラスライス／Ｉピクチャ／イントラタイルグループのためのイントラ、イントラブロックコピー、パレットモード、およびＣＰＭモードを含んでもよい。

ｂ．代替的に、予測モードは、イントラスライス／Ｉピクチャ／イントラタイルグループのためのイントラ、パレットモード、およびＣＰＭモードを含んでもよい。

ｃ．一例において、予測モードは、スライス間／および／またはＢピクチャ／タイルグ

ループのためのイントラ、インター、イントラブロックコピー、パレットモード、およびＣＰＭモードを含んでもよい。

ｄ．代替的に、予測モードは、イントラ、インター、イントラブロックコピー、パレットモード、およびＣＰＭモードのうちの少なくとも２つを含んでもよい。

ｅ．一例において、予測モードがインターモード（すなわち、ＭＯＤＥ＿ＩＮＴＥＲ）である場合、ＣＰＭモードの使用の指示の信号通知はスキップされてもよい。さらに、現在の予測モードがＭＯＤＥ＿ＩＮＴＥＲである場合、ＣＰＭモードの使用の指示がＦＡＬＳＥであると推測されてもよい。

【０２３４】

１１．ＣＰＭは、既存の予測モードの特殊なケースとして扱われてもよい。この場合、ＣＰＭモードに関する構文は、既存の予測モードの特定の使用条件下でさらに信号通知されてもよい。

ａ．一例として、予測モードがイントラモード（すなわち、ＭＯＤＥ＿ＩＮＴＲＡ）である場合、ＣＰＭ関連の構文の信号通知はスキップされ得る。さらに、現在の予測モードがＭＯＤＥ＿ＩＮＴＲＡである場合、ＣＰＭモードの使用の指示がＦＡＬＳＥであると推測されてもよい。

ｂ．一例として、予測モードがスキップモードである場合（すなわち、スキップフラグが１である場合）、ＣＰＭ関連構文の信号通知をスキップすることができる。さらに、現在のＣＵでスキップモードが採用されている場合、ＣＰＭの使用状況の指示は誤っていると推測されてもよい。

ｃ．一例として、予測モードがパレットモード（例えば、ＭＯＤＥ＿ＰＬＴ）である場合、ＣＰＭおよびＣＰＭ関連の構文の使用の指示が信号通知されてもよい。

ｄ．あるいは、予測モードがイントラモード、インターモード、またはイントラブロックコピーモードである場合、ＣＰＭ関連の構文をスキップすることができる。

ｅ．あるいは、予測モードがパルス符号変調（ＰＣＭ）モードではなくイントラモードである場合、ＣＰＭ関連の構文が信号通知されてもよい。

ｆ．一例として、予測モードがパレットモード（例えば、ＭＯＤＥ＿ＰＬＴ）である場合、ＣＰＭおよびＣＰＭ関連構文の使用の指示は、ＰＣＭモードの使用の指示の前または後に信号通知されてもよい。一例として、ＣＰＭモードが適用される場合、ＣＰＭ関連の構文の信号通知が行われてもよい。

ｇ．　予測モードがイントラモード、インターモード、またはイントラブロックコピーモードの場合、ＣＰＭ関連の構文の信号通知がスキップされてもよい。

ｈ．あるいは、予測モードがパルス符号変調（ＰＣＭ）モードではなくイントラモードである場合、ＣＰＭ関連の構文が信号通知されてもよい。

ｉ．一例として、予測モードがイントラブロックコピーモードである場合、ＣＰＭ関連の構文が信号通知されてもよい。

ｊ．あるいは、予測モードがパレットモード、インターモード、またはイントラモードである場合、ＣＰＭ関連の構文の信号通知をスキップされてもよい。

ｋ．上記の方法は、ＰＣＩＰまたは他のパレット関連の方法にも適用できる。

【０２３５】

１２．ＣＰＭで符号化されたブロックのＭＶは、後続のブロックのＭＶ予測としても使用されてもよい。

ａ．一例として、ＣＰＭで符号化されたブロックのＭＶは、他のブロックのマージ候補として使用されてもよい。

ｂ．一例として、ＣＰＭで符号化されたブロックのＭＶは、他のブロックのＨＭＶＰ候補として使用されてもよい。

ｉ．あるいは、さらに、ＣＰＭで符号化されたブロックを符号化した後に、ＨＭＶＰテーブルを更新してもよい。

ｃ．一例として、ＣＰＭで符号化されたブロックのＭＶは、他のブロックのＡＭＶＰ候補として使用されてもよい。

ｄ．あるいは、１つのＣＰＭ符号化ブロックを符号化した後、ＨＭＶＰテーブル更新プロセスは許可されない。

ｅ．あるいは、ＣＰＭで符号化されたブロックのＢＶを、後続のブロックの動きベクトル予測子として使用することを許可しない場合がある。

【０２３６】

１３．ＰＣＩＰで符号化されたブロックのイントラ予測方向は、後続のブロックのイントラモード予測子としても使用されてもよい。

ａ．一例として、ＰＣＩＰで符号化されたブロックのイントラ予測方向は、後続のブロックのＭＰＭモードとして使用してもよい。

ｉ．一例として、ＰＣＩＰで符号化されたブロックが水平方向を選択する場合、イントラモードはＶＥＲ＿ＩＤＸに設定されてもよく、後続のブロックのためのＭＰＭモードとして使用してもよい。

ｉｉ．一例として、ＰＣＩＰで符号化されたブロックが垂直方向を選択する場合、イントラモードはＨＯＲ＿ＩＤＸに設定されてもよく、後続のブロックのためのＭＰＭモードとして使用してもよい。

ｂ．あるいは、ＣＰＭ符号化ブロックのイントラ予測方向を後続のブロックのＭＰＭとして使用することを許可しない場合がある。

【０２３７】

１４．使用するパレットの最大エントリ数は、１つの映像ユニット（ＣＴＢ、ＣＵ、ＰＵ、ＴＵなど）から別の映像ユニットに変更できる。二値化は、使用するパレットの最大エントリ数に応じて変更され得る。

ａ．一例として、それは符号化された情報に依存してもよい。

ｉ．一例として、それはブロックの寸法に依存してもよい。

ａ）一例として、パレットモードで符号化されたより大きなブロックのサイズは、より大きい最大エントリ数を有する可能性があり、一方、パレットモードで符号化されたより小さなブロックのサイズは、より小さい最大エントリ数を有する可能性がある。

ｂ）１つの例では、４×４ブロックの場合、Ｎ０パレットエントリのみが許可される。

ｃ）一例として、８×４ブロックおよび／または４×８ブロックの場合、Ｎ１パレットエントリのみが許可され得る。

ｄ）一例として、８×８ブロックの場合、Ｎ２パレットエントリのみが許可される。

ｅ）一例として、１６×１６ブロックの場合、Ｎ３パレットエントリのみが許可される。

ｆ）一例として、Ｎ０、Ｎ１、Ｎ２およびＮ３は固定数（例えば、３、７または１５）であり、それらは同じであっても異なっていてもよい。

ｇ）一例として、パレットエントリの最大値とブロック寸法の関係を次のように表にしている。ここで、ＷとＨはそれぞれ現在のブロック幅とブロック高さを示し、Ｍａｘは２つの入力間で大きい方を取得する操作である。

【０２３８】

【表１２】

表４－１　最大パレットエントリとブロック寸法の関係の一例

最大（W、H）	パレットエントリの最大値
4	3
8	7
16	15
32	31
64	31
128	31

【０２３９】
　　ｈ）一例として、パレットエントリの最大値とブロック寸法の関係を次のように表にしている。ここで、ＷとＨはそれぞれ現在のブロック幅とブロック高さを示し、Ｍａｘは２つの入力間で大きい方を取得する操作である。
【０２４０】
【表１３】

表４－２　最大パレットエントリとブロック寸法の関係の一例

最大（W、H）	パレットエントリの最大値
4	3
8	15
16	15
32	31
64	31
128	31

【０２４１】
　ｂ．一例において、それは、色成分に依存してもよい。
　　ｉ．一例として、パレットモードで符号化された輝度ブロックは、最大エントリ数が大きい可能性があり、パレットモードで符号化されたクロマブロックは、最大エントリ数が小さい可能性がある。
　　ａ）一例として、輝度ブロックの最大エントリ数はＮであり得、したがって、クロマブロックの最大エントリ数はＭであり得る。
　　ｉ．一例として、Ｍは固定数（例えば、３、７または１５）であり、ＭはＮより小さい場合がある。
　ｃ．一例として、それは、デュアルツリーまたはシングルツリーなどの分割構造に依存してもよい。
【０２４２】
１５．パレット／ＣＰＭ／ＰＣＩＰ符号化ブロックのスケーリング行列（例えば、ユーザー定義の
スケーリング行列など）を禁止することを提案している。
　ａ．あるいは、スケーリング行列をパレット／ＣＰＭ／ＰＣＩＰ符号化ブロックに適用することもできる。
　ｂ．一例として、パレット／ＣＰＭ／ＰＣＩＰ符号化ブロックのスケーリング行列を選択する方法は、変換スキップ符号化ブロックと同じ方法で行われ得る。
【０２４３】
１６．ＩＢＣモードの使用を符号化するためのコンテクストモデリング（例えば、ｃｕ＿ｓｋｉｐ＿ｆｌａｇおよび／またはｐｒｅｄ＿ｍｏｄｅ＿ｉｂｃ＿ｆｌａｇ）は、現在のブロックの符号化された情報、例えば、ブロックの寸法／ブロックの形状／分割ツリーの深さ（例えば、ＱＴ／ＢＴ／ＴＴ／ＭＴＴの深さ）および／またはスライスタイプに依存してもよい。
　ａ．一例において、ＩＢＣフラグの指示を符号化するためのコンテクストは、ブロックサイズに基づいて導出されてもよい。
　　ｉ．一例において、異なるブロックサイズの下でのＩＢＣモードの使用の指示は、異なるコンテクストを使用することによって信号通知されてもよい。
　　ｉｉ．あるいは、Ｔより小さいまたはＴより大きくないブロックサイズを有するブロックのＩＢＣモードの使用の指示は、１つのコンテクストを使用することができる。他のブロックサイズのブロックは、他のコンテクスト（例えば、１つのみ）を使用してもよい。
　　ｉｉｉ．あるいは、Ｔより大きいまたはＴより小さくないブロックサイズを有するブロックのＩＢＣモードの使用の指示は、１つのコンテクストを使用することができる。他

のブロックサイズのブロックは、他のコンテクスト（例えば、１つのみ）を使用してもよい。

ｉｖ．一例において、Ｔは、以下に基づいてもよい。

１．映像コンテンツ（例えば、スクリーンコンテンツまたは自然コンテンツ）

２．ＤＰＳ／ＳＰＳ／ＶＰＳ／ＰＰＳ／ＡＰＳ／ピクチャヘッダ／スライスヘッダ／タイルグループヘッダ／最大符号化ユニット（ＬＣＵ）／符号化ユニット（ＣＵ）／ＬＣＵ行／ＬＣＵグループ／ＴＵ／ＰＵにおいて信号通知されるメッセージ

３．カラーフォーマットの表示（例えば、４：２：０、４：４：４）

４．符号化ツリー構造

５．スライス／タイルグループのタイプおよび／またはピクチャのタイプ

６．色成分（例えば、クロマ成分または輝度成分にのみ適用されてもよい）

７．時間層ＩＤ

８．標準のプロファイル／レベル／層

９．デコーダに信号が送信される。

ｂ．一例において、ＩＢＣモードの使用を符号化するためのコンテクストモデリングは、ブロック形状（例えば、正方形または非正方形、幅と高さの比）に基づいて導出されてもよい。

ｉ．あるいは、異なるブロック形状の下でのＩＢＣフラグの指示は、異なるコンテクストを使用することによって信号通知されてもよい。

ｃ．一例において、Ｉスライスのためのｃｕ＿ｓｋｉｐ＿ｆｌａｇのコンテクストモデリングは、Ｐ／Ｂスライスのためのコンテクストモデリングとは異なってもよい。

ｉ．一例において、ｃｕ＿ｓｋｉｐ＿ｆｌａｇのコンテクストモデリングは、近傍のブロックのｃｕ＿ｓｋｉｐ＿ｆｌａｇ情報に依存していてもよい。ＰＢスライスの場合、ｃｕ＿ｓｋｉｐ＿ｆｌａｇの符号化には１つのコンテクストのみを用いることができる。またはその逆である。

ｄ．一例において、１つのコンテクストのみが、ＩＢＣモードの使用を符号化するために利用されてもよい。

ｅ．上記方法は、特定のスライス／ピクチャタイプに使用されるものにのみ適用されてもよい。

【０２４４】

１７．ｃｕ＿ｓｋｉｐ＿ｆｌａｇのコンテクストモデリングは、近傍のブロックがＩＢＣモードで符号化されているか否かに依存してもよい。

ａ．一例において、近傍のブロックのうちの少なくとも１つまたは両方がＩＢＣモードで符号化される場合、１つのコンテクストを使用してもよい。

【０２４５】

１８．第１の構文要素に関連付けられたビンのコンテクスト符号化は、近傍のブロック（隣接または非隣接）に関連付けられた第２の構文要素の値に依存してもよい。

ａ．一例において、第１の構文要素は、イントラ－インター予測モード／三角形予測モードを組み合わせたことを示すためのものであってもよい。

ｂ．一例において、第２の構文要素は、ＩＢＣの使用に関するものであってもよいし、または第１の構文要素に関連付けられたビンのコンテクスト符号化は、近傍のブロックのＩＢＣモードの使用に依存してもよい。

ｃ．あるいは、このメッセージは、現在のスライス／タイル／シーケンスがスクリーンコンテンツであるかどうかを示してもよい。

【０２４６】

１９．符号化ブロックフラグ（ｃｂｆ）は、信号通知される代わりに、ＱＲ－ＢＤＰＣＭモードで符号化されたブロックについて推測されてもよい。

ａ．あるいは、一例において、ｃｂｆは、ＱＲ－ＢＤＰＣＭモードで符号化されたブロックにおいて、１と推測されてもよい。

ｂ．上記方法は、輝度ｃｂｆおよび／またはクロマｃｂｆに適用される。

【０２４７】
２０．現在のクロマブロックの中心位置を覆い、且つ／またはＤＭモードを導出するために使用される対応する輝度ブロックがイントラモード（例えば、ＭＯＤＥ＿ＩＮＴＲＡ）で符号化されていない場合、クロマＤＭモードは許可されないことがある。Ａを、ＤＭモード導出処理に用いられる対応する輝度ブロック（例えば、現在のクロマブロックの対応する中心位置を覆うもの）とする。
　ａ．一例において、ＡがＩＢＣおよび／またはＩＢＣモードの変形で符号化される場合、クロマＤＭモードは許可されなくてもよい。
　ｂ．一例において、Ａがパレットおよび／またはパレットモードの変形で符号化される場合、クロマＤＭモードは許可されなくてもよい。
　ｃ．一例において、ＡがＰＣＭおよび／またはＰＣＭモードの変形で符号化される場合、クロマＤＭモードは許可されなくてもよい。
　ｄ．上記例において、クロマＤＭモードが許可されない場合（例えば、対応する輝度ブロックの予測モードが特定のモードであることに起因して）、クロマＤＭモードの指示は、通知されてもよいし、または信号通知されなくてもよい。
　　ｉ．あるいは、クロマＤＭモードの指示が信号通知されてもよい。しかしながら、適合ビットストリームは、クロマＤＭモードが許可されない場合（例えば、対応する輝度ブロックの予測モードが特定のモードであるため）、復号化されたｉｎｔｒａ＿ｃｈｒｏｍａ＿ｐｒｅｄ＿ｍｏｄｅが特定の値でない（例えば、ＣＣＬＭがオフである場合は４、ＣＣＬＭがオンである場合は７）ことを満たすべきである。
　ｅ．一例において、ＡがＱＲ－ＢＤＰＣＭモードで符号化されている場合、クロマＤＭモードにおけるイントラ予測モードは、Ａのイントラ予測方向（例えば、ｂｄｐｃｍ＿ｄｉｒ＿ｆｌａｇ）に基づいて導出されてもよい。
　　ｉ．一例において、Ａのイントラ予測方向が垂直である（例えば、ｂｄｐｃｍ＿ｄｉｒ＿ｆｌａｇが１である）場合、クロマＤＭモードにおけるイントラ予測モードは、垂直モード（例えば、ＶＥＲ＿ＩＤＸ）でもよい。
　　ｉｉ．一例において、Ａのイントラ予測方向が水平である（例えば、ｂｄｐｃｍ＿ｄｉｒ＿ｆｌａｇが０である）場合、クロマＤＭモードにおけるイントラ予測モードは、水平モード（例えば、ＨＯＲ＿ＩＤＸ）でもよい。
　ｆ．一例において、クロマＤＭモードにおけるイントラ予測モードは、複合ＩＢＣ及びパレットモードの使用に基づいて導出されてもよい。
　　ｉ．一例において、ＤＭは、平面または水平／垂直予測モードに設定してもよい。
【０２４８】
２１．現在のクロマブロックの中心位置を含み、且つ／またはＤＭモードを導出するために用いられる対応する輝度ブロックがイントラモード（例えば、ＭＯＤＥ＿ＩＮＴＲＡ）で符号化されていない場合、イントラ予測方法で生成された予測ブロックによって、ＤＭモードを有するクロマブロックの予測ブロックを埋めてもよい。Ａを、ＤＭモード導出処理に用いられる対応する輝度ブロック（例えば、現在のクロマブロックの対応する中心位置を覆うもの）とする。
　ａ．一例において、Ａがパレットおよび／またはパレットモードの変形で符号化される場合、イントラ予測方法は、垂直イントラ予測に使用される方法でもよい。
　　ｉ．あるいは、一例において、Ａがパレットおよび／またはパレットモードの変形で符号化される場合、クロマＤＭモードの予測は、水平予測によって埋めてもよい。
　　ｉｉ．あるいは、一例において、Ａがパレットおよび／またはパレットモードの変形で符号化される場合、クロマＤＭモードの予測は、平面予測によって埋めてもよい。
　　ｉｉｉ．あるいは、一例において、Ａがパレットおよび／またはパレットモードの変形で符号化される場合、クロマＤＭモードの予測は、ＤＣ予測によって埋めてもよい。
　　ｉｖ．あるいは、一例において、Ａがパレットおよび／またはパレットモードの変形で符号化される場合、クロマＤＭモードの予測は、任意のイントラ予測モードによって埋めてもよい。

ｂ．一例において、クロマブロックの予測は、現在の色成分または他の色成分における近傍（隣接または非隣接）輝度ブロックのイントラ予測モードおよび／または近傍（隣接または非隣接）のクロマブロックのイントラ予測モードに基づいて生成されてもよい。

ｃ．一例において、イントラ予測方法は、ＭＰＭリスト構築時にＰＣＭブロックに割り当てられるイントラ予測モードに対応していてもよい。

ｄ．一例において、イントラ予測方法は、輝度ＰＣＭブロックに割り当てられたイントラ予測モードに対応していてもよい。

ｅ．一例において、イントラ予測方法は、クロマＰＣＭブロックに割り当てられたイントラ予測モードに対応していてもよい。

ｆ．いくつかの実装形態において、イントラ予測方法は、ＡのＭＰＭ一覧における第１のイントラ予測方式に対応していてもよい。

ｇ．上記方法は、ＡがＩＢＣおよび／またはＩＢＣモードの変形で符号化される場合に適用されてもよい。

ｉ．あるいは、上記方法は、ＡがＰＣＭおよび／またはＰＣＭモードの変形で符号化される場合に適用されてもよい。

ｉｉ．あるいは、上記方法は、ＡがＱＲ－ＢＤＰＣＭおよび／またはＱＲ－ＢＤＰＣＭモードの変形で符号化される場合に適用されてもよい。

【０２４９】

２２．無効なブロックベクトルを有するクロマブロックを処理する方法に従って、ＤＭモードを有するクロマブロックの予測ブロックを導出してもよい。

ａ．一例において、ＤＭモードを有するクロマブロックの予測ブロックは、デフォルト値で埋めてもよい。

ｉ．あるいは、デフォルト値はＭに固定されてもよい。

ｉｉ．あるいは、デフォルト値は、入力および／または内部ビット深度に依存してもよい。

ｂ．一例において、上記方法は、対応する輝度ブロックの予測モードがイントラモードでない場合に、適用されてもよい。

ｉ．一例において、上記方法は、対応する輝度ブロックの予測モードがＩＢＣモードおよび／またはＩＢＣモードの変形である場合に、適用されてもよい。

ｉｉ．一例において、上記方法は、対応する輝度ブロックの予測モードがパレットモードおよび／またはパレットモードの変形である場合に、適用されてもよい。

【０２５０】

２３．１つのブロックは、デフォルトのイントラモードを有してもよい。

ａ．一例において、デフォルトモードは、ＤＣ／ＰＬＡＮＡＲ／ＶＥＲ／ＨＯＲモードであってもよい。

ｂ．一例において、デフォルトモードは、任意のイントラモードであってもよい。

ｃ．一例において、デフォルトモードは、イントラモード符号化処理における最大確率モード（ＭＰＭ）として用いられてもよい。

ｉ．あるいは、一例において、近傍のブロックからイントラモードをフェッチするとき、このブロックと現在のブロックとが異なる映像符号化ユニット（例えば、タイル、スライス、ブリック）にある場合、近傍のブロックのデフォルトモードが使用される。

ｄ．一例において、クロマＤＭモードにおいて、対応するルミナンスブロックのデフォルトモードは、ＤＭモードを導出するために使用されてもよい。

ｅ．ブロックがイントラモードで符号化されていない場合、上記方法が使用されてもよい。

ｉ．あるいは、上記方法は、ブロックがＩＢＣ／インター／ＱＲＤＰＣＭ／ＰＣＭ／パレットモードで符号化される場合に使用されてもよい。

ｉｉ．あるいは、現在のブロックのイントラモードを復号化しない場合、上記方法が使用されてもよい。

【０２５１】

２４．ＭＰＭリスト構築中、近傍のブロックがＰＣＭモードで符号化される場合、それをＤＣ予測モードと見なしてもよい。

【０２５２】

２５．ＴＰＭおよび／またはＴＰＭブレンド処理および／または動き情報の記憶をどのように適用するかは、近傍のブロックの符号化された情報に基づいて決定されてもよい。

ａ．一例において、その近傍のブロック（例えば、図１２における左側及び上側の近傍のブロック）がＩＢＣ／パレット／ＴＳ／ＱＲ－ＢＤＰＣＭモードで符号化される場合、ＴＰＭブレンド処理は、現在のブロックに対して無効化されてもよい。

ｂ．一例において、その近傍のブロック（例えば、図１２における左側及び上側の近傍のブロック）がＩＢＣ／パレット／ＴＳ／ＱＲ－ＢＤＰＣＭモードで符号化される場合、現在のブロックに対して別のブレンド処理／動き記憶処理を適用してもよい。

【０２５３】

<u>恒等変換を用いて符号化されたブロックのための量子化パラメータ（例えば、変換スキップモード、ＱＲ－ＢＤＰＣＭ等）</u>

【０２５４】

２６．恒等変換を使用する符号化ツールの指示の信号通知は、ビットストリームからの復号化された量子化パラメータに依存してもよい。

ａ．一例において、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇは、いくつかのＱＰに信号通知されなくてもよく、この場合、このフラグは０であると推測される。

ｉ．一例において、ＱＰが（４＋ＱｐＢｄＯｆｆｓｅｔＹ）より小さい場合、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇは通知されなくてもよい。

ｂ．ｉｎｔｒａ＿ｂｄｐｃｍ＿ｆｌａｇは、いくつかのＱＰに信号通知されなくてもよく、この場合、このフラグは０であると推測される。

ｉ．一例において、ＱＰが（４＋ＱｐＢｄＯｆｆｓｅｔＹ）より小さい場合、ｉｎｔｒａ＿ｂｄｐｃｍ＿ｆｌａｇは信号通知されなくてもよい。

【０２５５】

２７．恒等変換符号化されたブロックのビットストリームから復号化された量子化パラメータは、輝度／クロマ量子化パラメータの範囲のオフセット（ＱｐＢｄＯｆｆｓｅｔＹ，ＱｐＢｄＯｆｆｓｅｔＣ）に基づいて変更されてもよい。

ａ．一例において、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、ＱＰクリッピングが適用してもよい。

ｉ．一例において、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、ＱＰ値が（４＋ＱｐＢｄＯｆｆｓｅｔＹ）より小さい場合、ＱＰ値を（４＋ＱｐＢｄＯｆｆｓｅｔＹ）に設定してもよい。

ｉｉ．一例において、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、ＱＰ値は、ＱＴより小さい場合、ある値ＱＴに設定してもよい。

ｂ．ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、復号化されたＱＰ値にオフセットを加えることができ、これをＱＰｄｅｃとする。

ｉ．一例において、ＱＰｄｅｃの代わりにＱＰｄｅｃ＋４＋ＱｐＢｄＯｆｆｓｅｔＹをｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１であるブロックに適用してもよい。

ｃ．上記方法は、例えば、ｉｎｔｒａ＿ｂｄｐｃｍ＿ｆｌａｇが１である場合、ＱＲ－ＤＰＣＭ符号化ブロックにも適用可能であってもよい。

ｄ．一例において、修正された量子化パラメータは、後続のブロックのデルタＱＰ信号通知の量子化／逆量子化／フィルタリング／予測に使用されてもよい。

【０２５６】

２８．固定オフセットに基づいて、恒等変換符号化されたブロックのビットストリームから復号化された量子化パラメータを変更してもよい。

ａ．一例において、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、ＱＰクリッピングが適用してもよい。

ｉ．一例において、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、ＱＰ

値が４より小さい場合、ＱＰ値を４に設定してもよい。

ｉｉ．一例において、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、ＱＰ値は、ＱＴより小さい場合、ある値ＱＴに設定してもよい。

ｂ．ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、復号化されたＱＰ値にオフセットを加えることができ、これをＱＰｄｅｃとする。

ｉ．一例において、ＱＰｄｅｃの代わりにＱＰｄｅｃ＋４を、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１であるブロックに適用されてもよい。

ｃ．上記方法は、例えば、ｉｎｔｒａ＿ｂｄｐｃｍ＿ｆｌａｇが１である場合、ＱＲ－ＤＰＣＭ符号化ブロックにも適用可能であってもよい。

ｄ．一例において、修正された量子化パラメータは、後続のブロックのデルタＱＰ信号通知の量子化／逆量子化／フィルタリング／予測に使用されてもよい。

【０２５７】

２９．恒等変換で符号化されたブロックのためのビットストリームからの復号化された量子化パラメータは、入力ビット深度に基づいて修正されてもよく、ｂｄｉと表される。

ａ．一例において、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、ＱＰクリッピングが適用してもよい。

ｉ．一例において、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、ＱＰ値が（４＋６＊（ｂｄｉ－８））より小さい場合には、ＱＰ値を（４＋６＊（ｂｄｉ－８））に設定してもよい。

ｂ．ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、復号化されたＱＰ値にオフセットを加えることができ、これをＱＰｄｅｃとする。

ｉ．一例において、ＱＰｄｅｃの代わりにＱＰｄｅｃ＋（４＋６＊（ｂｄｉ－８））をｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１であるブロックに適用してもよい。

ｃ．上記方法は、例えば、ｉｎｔｒａ＿ｂｄｐｃｍ＿ｆｌａｇが１である場合、ＱＲ－ＤＰＣＭ符号化ブロックにも適用可能であってもよい。

ｄ．一例において、修正された量子化パラメータは、後続のブロックのデルタＱＰ信号通知の量子化／逆量子化／フィルタリング／予測に使用されてもよい。

【０２５８】

３０．恒等変換で符号化されたブロックのためのビットストリームからの復号化された量子化パラメータは、内部ビット深度（ｂｄ）および入力ビット深度に基づいて修正されてもよく、ｂｄｉと表される。

ａ．一例において、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、ＱＰクリッピングが適用してもよい。

ｉ．一例において、ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、ＱＰ値が（４＋６＊（ｂｄ－ｂｄｉ））より小さい場合、ＱＰ値を（４＋６＊（ｂｄ－ｂｄｉ））に設定されてもよい。

ｂ．ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１である場合、復号化されたＱＰ値にオフセットを加えることができ、これをＱＰｄｅｃとする。

ｉ．一例において、ＱＰｄｅｃの代わりにＱＰｄｅｃ＋（４＋６＊（ｂｄ－ｂｄｉ））をｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇが１であるブロックに適用してもよい。

ｃ．上記方法は、例えば、ｉｎｔｒａ＿ｂｄｐｃｍ＿ｆｌａｇが１である場合、ＱＲ－ＤＰＣＭ符号化ブロックにも適用可能であってもよい。

ｄ．一例において、修正された量子化パラメータは、後続のブロックのデルタＱＰ信号通知の量子化／逆量子化／フィルタリング／予測に使用されてもよい。

【０２５９】

３１．上述した復号化された量子化パラメータは、内部ビット深度ＱＰオフセットを考慮しない（例えば、ＱｐＢｄＯｆｆｓｅｔＹを付加しない）量子化パラメータであってもよい。

ａ．あるいは、上記復号化された量子化パラメータは、内部ビット深度ＱＰオフセットを考慮した量子化パラメータであってもよい。

【０２６０】
３２．ビットストリームにおける入力ビット深度を示すことが提案される。
　ａ．一例において、ＳＰＳ／ＶＰＳにおいて、入力映像のビット深度を示すために、（ｉｎｐｕｔ＿ｂｉｔｄｅｐｔｈ＿８）である構文要素ｉｎｐｕｔ＿ｂｉｔｄｅｐｔｈ＿ｍｉｎｕｓ８を信号通知してもよい。
　　ｉ．一例において、構文要素は、０．．ｄＱＰの範囲を有し、ここで、ｄＱＰは固定値である。
　ｂ．一例において、ＳＰＳ／ＶＰＳにおいて、（ｉｎｔｅｒｎａｌ＿ｂｉｔｄｅｐｔｈ－ｉｎｐｕｔ＿ｂｉｔｄｅｐｔｈ）である構文要素ｂｉｔｄｅｐｔｈ＿ｍｉｎｕｓ＿ｉｎｐｕｔ＿ｂｉｔｄｅｐｔｈを信号通知し、入力映像のビット深度を示すことができる。
　　ｉ．一例において、構文要素は、０．．ｄＱＰの範囲を有し、ここで、ｄＱＰは固定値である。
【０２６１】
３３．上記方法を適用するかどうか、および／またはどのように適用するかは、以下に基づいてもよい。
　ａ．映像コンテンツ（例えば、スクリーンコンテンツまたは自然コンテンツ）
　ｂ．ＤＰＳ／ＳＰＳ／ＶＰＳ／ＰＰＳ／ＡＰＳ／ピクチャヘッダ／スライスヘッダ／タイルグループヘッダ／最大符号化ユニット（ＬＣＵ）／符号化ユニット（ＣＵ）／ＬＣＵ行／ＬＣＵグループ／ＴＵ／ＰＵブロック／映像符号化ユニットにおいて信号通知されるメッセージ
　ｃ．ＣＵ／ＰＵ／ＴＵ／ブロック／映像符号化ユニットの位置
　ｄ．現在のブロックおよび／またはその近傍のブロックのブロック寸法
　　ｉ．一例において、上記方法は、現在のブロックの幅及び高さの両方がＴより小さい（例えば、３２または６４）場合にのみ適用してもよい。
　ｅ．現在のブロックおよび／またはその近傍のブロックのブロック形状
　ｆ．現在のブロックおよび／またはその近傍のブロックの予測モード（例えば、ｐｒｅｄ＿ｍｏｄｅ＿ｉｂｃ＿ｆｌａｇ）
　ｇ．現在のブロック及び／またはその近傍のブロックのイントラモード
　ｈ．現在のブロックおよび／またはその近傍のブロックの動き／ブロックベクトル
　ｉ．カラーフォーマットの表示（例えば、４：２：０、４：４：４）
　ｊ．符号化ツリー構造
　ｋ．スライス／タイルグループのタイプおよび／またはピクチャのタイプ
　ｌ．色成分（例えば、クロマ成分または輝度成分にのみ適用されてもよい）。
　ｍ．時間層ＩＤ
　ｎ．標準のプロファイル／レベル／層
【０２６２】
５　ＪＶＥＴ－Ｎ１００１－ｖ６に加えた実施例
【０２６３】
【化１】
　ＪＶＥＴ－Ｎ１００１－ｖ６によって提供されたドラフト上の以下の変更は、*太字のイタリック体*で強調表示されている。削除されたテキストには二重括弧で囲んだ印が付けられている（例えば、［［ａ］］は文字「ａ」の削除を意味する）。

【０２６４】
シーケンスパラメータセットＲＢＳＰ構文
【０２６５】

【表１４】

seq_parameter_set_rbsp( ) {	Descriptor
sps_max_sub_layers_minus1	u(3)
…	
sps_ibc_enabled_flag	u(1)
sps_palette_enabled_flag	u(1)
…	u(1)
}	

【０２６６】
符号化ユニット構文
【０２６７】

【表１５】

coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	Descriptor
if(slice_type != I || sps_ibc_enabled_flag) {	
if(treeType != DUAL_TREE_CHROMA && !(cbWidth = = 4 && cbHeight = = 4 && !sps_ibc_enabled_flag))	
cu_skip_flag[x0][y0]	ae(v)
[[if(cu_skip_flag[x0][y0] = = 0 && slice_type != I && !(cbWidth = = 4 && cbHeight = = 4))	
pred_mode_flag	ae(v)
if(((slice_type = = I && cu_skip_flag[x0][y0] = =0) || (slice_type != I && (CuPredMode[x0][y0] != MODE_INTRA || (cbWidth = = 4 && cbHeight = = 4 && cu_skip_flag[x0][y0] = = 0)))) && sps_ibc_enabled_flag && (cbWidth != 128 || cbHeight != 128))	
pred_mode_ibc_flag	ae(v)]]
}	
if(cu_skip_flag[x0][y0] == 0)	
pred_modes(x0, y0, cbWidth, cbHeight)	
if(cu_palette_flag [x0][y0])· {	
cu_palette_ibc_mode	
if(treeType = = SINGLE_TREE) {	
palette_coding(x0, y0, , nWidth, nHeight, 0, 3)	
} else {	
if(treeType = = DUAL_TREE_LUMA)	
palette_coding(x0, y0, , nWidth, nHeight, 0, 1)	
if(treeType = = DUAL_TREE_ CHROMA)	
palette_coding(x0, y0, , nWidth/SubWidthC, nHeight/SubHeightC , 1, 2)	
}	
} else {	

【表１５】

if(CuPredMode[x0][y0] == MODE_INTRA) {	
if(sps_pcm_enabled_flag && cbWidth >= MinIpcmCbSizeY && cbWidth <= MaxIpcmCbSizeY && cbHeight >= MinIpcmCbSizeY && cbHeight <= MaxIpcmCbSizeY)	
pcm_flag[x0][y0]	
…	
}	
}	

【０２６８】
予測モード構文
以降の全文と表は、ＶＶＣの現在のバージョンに新たに追加されるものとして提案されている。
【０２６９】

【表１７】

pred_modes (x0, y0, cbWidth, cbHeight) {	Descriptor
if(sps_ibc_enabled_flag) {	
if(slice_type == I ||(cbWidth == 4 && cbHeight == 4)){	
if(cbWidth < 128 || cbHeight < 128)	
pred_mode_ ibc_flag	ae(v)
if(!pred_mode_ ibc_flag && sps_plt_enabled_ flag && cbWidth <=64 && cbHeight <= 64)	
cu_palette_flag	
}else{	ae(v)
pred_mode_flag	
if(CuPredMode[x0][y0] == MODE_INTRA){	
if(sps_plt_enabled_flag && cbWidth <=64 && cbHeight <= 64)	
cu_palette_flag	ae(v)
}	
else{	
if(cbWidth < 128 || cbHeight < 128)	
pred_mode_ibc_flag	ae(v)
}	
}	
}	
else{	
if(slice_type == I ||(cbWidth == 4 && cbHeight == 4)){	
if(sps_plt_enabled_flag && cbWidth <=64 && cbHeight <= 64)	
cu_palette_flag	
}else{	
pred_mode_flag	
if(CuPredMode[x0][y0] != MODE_INTRA && sps_plt_enabled_flag && cbWidth <=64 && cbHeight <= 64)	
cu_palette_flag	
}	
}	
}	

【０２７０】
パレット構文
以降の全文と表は、現在のバージョンに新たに追加されるものとして提案されている。
【０２７１】

【表１８】

palette_coding(x0, y0, nWidth, nHeight, startComp, numComps) {	Descriptor
paletteP redictionFinished = 0	
NumPredictedPaletteEntries = 0	
for(predictorEntryIdx = 0; predictorEntryIdx < PredictorPaletteSize [startComp] && !palettePredictionFinished && NumPredictedPaletteEntries [startComp] < palette_max_size; predictorEntryIdx++) {	
palette_predictor_run	ae(v)
if(palette_predictor_run != 1) {	
if(palette_predictor_run > 1)	
predictorEntryIdx += palette_predictor_run − 1	
PalettePredictorEntryReuseFlags[startComp][predictorEntryIdx] = 1	
NumPredictedPaletteEntries[startComp]++	
} else	
palettePredictionFinished = 1	
}	
if(NumPredictedPaletteEntries[startComp] < palette_max_size)	
num_signalled_palette_entries[startComp]	ae(v)
for(cIdx = startComp; cIdx < (startComp + numComps); cIdx++)	
for(i = 0; i < num_signalled_palette_entries[startComp]; i++)	
new_palette_entries[startComp][i]	ae(v)
if(CurrentPaletteSize != 0)	
palette_escape_val_present_flag[startComp]	ae(v)
if(MaxPaletteIndex > 1) {	
num_palette_indices_minus1	ae(v)
adjust = 0	
for(i = 0; i <= num_palette_indices_minus1; i++) {	
if(MaxPaletteIndex − adjust > 1) {	
palette_index_idc	ae(v)
PaletteIndexIdc [i] = palette_index_idc	
}	
adjust = 1	
}	

【表１９】

copy_above_indices_for_final_run_flag	ae(v)
palette_transpose_flag[startComp]	ae(v)
}	
remainingNumIndices = num_palette_indices_minus1 + 1	
PaletteScanPos = 0	
while(PaletteScanPos <nWidth*nHeightt) {	
xC = x0 + ScanOrder[nWidth][nHeight][3][PaletteScanPos][0]	
yC = y0 + ScanOrder[nWidth][nHeight][3][PaletteScanPos][1]	
if(PaletteScanPos > 0) {	
xcPrev = x0 + ScanOrder[nWidth][nHeight][3][PaletteScanPos − 1][0]	
ycPrev = y0 + ScanOrder[nWidth][nHeight][3][PaletteScanPos − 1][1]	
}	
PaletteRun = nWidth * nHeight − PaletteScanPos − 1	
CopyAboveIndicesFlag[xC][yC] = 0	
if(MaxPaletteIndex > 1)	
if(((!palette_transpose_flag && PaletteScanPos>=nWidth) || ((palette_transpose_flag && PaletteScanPos>=nHeight)) && CopyAboveIndicesFlag[xcPrev][ycPrev] = = 0)	
if(remainingNumIndices > 0 && PaletteScanPos < nWidth* nHeight − 1) {	
copy_above_palette_indices_flag	ae(v)
CopyAboveIndicesFlag[xC][yC] = copy_above_palette_indices_flag	
} else	
if(PaletteScanPos = = nWidth * nHeight − 1 && remainingNumIndices > 0)	
CopyAboveIndicesFlag[xC][yC] = 0	
Else	
CopyAboveIndicesFlag[xC][yC] = 1	
if (CopyAboveIndicesFlag[xC][yC] = = 0) {	
currNumIndices = num_palette_indices_minus1 + 1 − remainingNumIndices	
CurrPaletteIndex = PaletteIndexIdc[currNumIndices]	
}	

【表２０】

if(MaxPaletteIndex > 1) {	
if (CopyAboveIndicesFlag[xC][yC] = = 0)	
remainingNumIndices − = 1	
PaletteMaxRun = nCbS * nCbS − PaletteScanPos − 1 − remainingNumIndices − copy_above_indices_for_final_run_flag	
if(remainingNumIndices > 0 || CopyAboveIndicesFlag[xC][yC] != copy_above_indices_for_final_run_flag)	
if(PaletteMaxRun > 0) {	
palette_run_prefix	ae(v)
if((palette_run_prefix > 1) && (PaletteMaxRun != (1 << (palette_run_prefix − 1))))	
palette_run_suffix	ae(v)
}	
}	
runPos = 0	
while (runPos <= PaletteRun) {	
xR = x0 + ScanOrder[nWidth][nHeight][3][PaletteScanPos][0]	
yR = y0 + ScanOrder[nWidth][nHeight][3][PaletteScanPos][1]	
if(CopyAboveIndicesFlag[xC][yC] = = 0) {	
CopyAboveIndicesFlag[xR][yR] = 0	
PaletteIndexMap[startComp][xR][yR] = CurrPaletteIndex	
} else {	
CopyAboveIndicesFlag[xR][yR] = 1	
if (!palette_transpose_flag)	
PaletteIndexMap[startComp] [xR][yR] = PaletteIndexMap[startComp] [xR][yR - 1]	
else	
PaletteIndexMap[startComp] [xR][yR] = PaletteIndexMap[startComp] [xR - 1][yR]	
}	
runPos++	
PaletteScanPos++	
}	
}	

【表２１】

if(palette_escape_val_present_flag) {	
for(cIdx = startComp; cIdx < (startComp + numComps); cIdx++)	
for(sPos = 0; sPos < nWidth* nHeight; sPos++) {	
xC = x0 + ScanOrder[nWidth][nHeight][3][sPos][0]	
yC = y0 + ScanOrder[nWidth][nHeight][3][sPos][1]	
if(PaletteIndexMap[startComp] [xC][yC] = = (MaxPaletteIndex – 1)){	
if(cIdx == 0 | | startComp != 0){	
palette_escape_sign	u(1)
palette_escape_val	ue3 (v)
PaletteEscapeVal[cIdx][xC][yC] = palette_escape_sign > 0 ? palette_escape_val : -palette_escape_val	
}	
if(startComp == 0 && cIdx != 0 && xC % 2 = = 0 && yC %2 = = 0){	
palette_escape_sign	u(1)
palette_escape_val	ue3 (v)
PaletteEscapeVal[cIdx][xC][yC] = palette_escape_sign > 0 ? palette_escape_val : -palette_escape_val	
}	
}	
}	
}	
}	

【０２７２】
パレット意味論
【０２７３】
以下の意味論において、配列インデックスｘ０，ｙ０は、ピクチャの左上輝度サンプルに対する、考慮される符号化ブロックの左上輝度サンプルの位置（ｘ０，ｙ０）を指定する。
【０２７４】
ｃｕ＿ｐａｌｅｔｔｅ＿ｆｌａｇは、現在の符号化ユニットでのパレットモードの使用を指定する。ｃｕ＿ｐａｌｅｔｔｅ＿ｆｌａｇ＝＝１は、パレットモードが現在の符号化ユニットに適用されていることを示す。ｃｕ＿ｐａｌｅｔｔｅ＿ｆｌａｇ＝＝０は、現在の符号化ユニットにパレットモードが適用されていないことを示す。
【０２７５】
ｐａｌｅｔｔｅ＿ｐｒｅｄｉｃｔｏｒ＿ｒｕｎは、配列ＰａｌｅｔｔｅＰｒｅｄｉｃｔｏｒＥｎｔｒｙＲｅｕｓｅＦｌａｇｓの非ゼロエントリの前にあるゼロの数を判定するために使用される。

palette_predictor_runの値が、0～（PredictorPaletteSize－predictorEntryIdx）の範囲であることがビットストリームの適合性の要件であり、predictorEntryIdxは配列PalettePredictorEntryReuseFlagsの現在の位置に対応するものである。変数NumPredictedPaletteEntriesは、予測パレットから再利用される現在のパレットのエントリの数を指定する。NumPredictedPaletteEntriesの値は、0からpalette_max_sizeまでの範囲内にあるべきである。

【0276】

num_signalled_palette_entries[startComp]は、現在のパレットテーブルstartCompの最初の色成分に対して明示的に通知される現在のパレットのエントリの数を指定する。

num_signalled_palette_entries[startComp]が存在しない場合、0であると推測される。

変数CurrentPaletteSizeは、現在のパレットのサイズを指定し、以下のように導出される。

CurrentPaletteSize[startComp]＝NumPredictedPaletteEntries[startComp]＋num_signalled_palette_entries[startComp]（XX）

CurrentPaletteSize[startComp]の値は、0からpalette_max_sizeまでの範囲内にあるべきである。

【0277】

new_palette_entries[cIdx][i]は、色成分cIdxのi番目の信号通知パレットエントリの値を指定する。

変数PredictorPaletteEntries[cIdx][i]は、色成分cIdxのための予測子パレットにおけるi番目の要素を指定する。変数CurrentPaletteEntries[cIdx][i]は、色成分cIdxの現在のパレットのi番目の要素を指定し、次のように導出される。

【0278】

```
numPredictedPaletteEntries＝0
for（i＝0；i＜PredictorPaletteSize[startComp]；i＋＋）
  if（PalettePredictorEntryReuseFlags[i]）{
    for（cIdx＝startComp；cIdx＜（startComp＋numComps）；cIdx＋＋）
      CurrentPaletteEntries[cIdx][numPredictedPaletteEntries]＝
        PredictorPaletteEntries[cIdx][i]
        numPredictedPaletteEntries＋＋
  }
for（i＝0；i＜num_signalled_palette_entries[startComp]；i＋＋）
  for（cIdx＝startComp；cIdx＜（startComp＋numComps）；cIdx＋＋）
    CurrentPaletteEntries[cIdx][numPredictedPaletteEntries＋i]＝
    new_palette_entries[cIdx][i]　（XX）
```

【0279】

palette_escape_val_present_flagが1のとき、現在の符号化ユニットに少なくとも1つのエスケープ符号化されたサンプルが含まれていること

を指定する。ｅｓｃａｐｅ＿ｖａｌ＿ｐｒｅｓｅｎｔ＿ｆｌａｇが０の場合、現在の符号化ユニットにエスケープ符号化されたサンプルがないことを指定する。存在しない場合、ｐａｌｅｔｔｅ＿ｅｓｃａｐｅ＿ｖａｌ＿ｐｒｅｓｅｎｔ＿ｆｌａｇの値は１と推測される。
変数ＭａｘＰａｌｅｔｔｅＩｎｄｅｘは、現在の符号化ユニットのパレットインデックスの可能な最大値を指定する。ＭａｘＰａｌｅｔｔｅＩｎｄｅｘの値は、ｃｕ＿ｐａｌｅｔｔｅ＿ｉｂｃ＿ｍｏｄｅが０の場合、ＣｕｒｒｅｎｔＰａｌｅｔｔｅＳｉｚｅ＋ｐａｌｅｔｔｅ＿ｅｓｃａｐｅ＿ｖａｌ＿ｐｒｅｓｅｎｔ＿ｆｌａｇに設定される。そうではなく、ｃｕ＿ｐａｌｅｔｔｅ＿ｉｂｃ＿ｍｏｄｅが１である場合、ＭａｘＰａｌｅｔｔｅＩｎｄｅｘは、ＣｕｒｒｅｎｔＰａｌｅｔｔｅＳｉｚｅ＋ｐａｌｅｔｔｅ＿ｅｓｃａｐｅ＿ｖａｌ＿ｐｒｅｓｅｎｔ＿ｆｌａｇ＋１に設定される。
【０２８０】
ｎｕｍ＿ｐａｌｅｔｔｅ＿ｉｎｄｉｃｅｓ＿ｍｉｎｕｓ１　ｐｌｕｓ１は、現在のブロックに対して明示的に信号通知されたか、推測されたパレットインデックスの数である。
ｎｕｍ＿ｐａｌｅｔｔｅ＿ｉｎｄｉｃｅｓ＿ｍｉｎｕｓ１が存在しない場合、０に等しいと推測される。
【０２８１】
ｐａｌｅｔｔｅ＿ｉｎｄｅｘ＿ｉｄｃは、ＣｕｒｒｅｎｔＰａｌｅｔｔｅＥｎｔｒｉｅｓによって表現されるアレイへのインデックスを示すものである。ｐａｌｅｔｔｅ＿ｉｎｄｅｘ＿ｉｄｃの値は、ブロックの第１のインデックスについては、０～ＭａｘＰａｌｅｔｔｅＩｎｄｅｘの範囲内にあり、ブロックの残りのインデックスについては、０～（ＭａｘＰａｌｅｔｔｅＩｎｄｅｘ－１）の範囲内にあるものとする。
ｐａｌｅｔｔｅ＿ｉｎｄｅｘ＿ｉｄｃが存在しない場合、それは０に等しいと推論される。
変数ＰａｌｅｔｔｅＩｎｄｅｘＩｄｃ［ｉ］は、ｉ番目のパレット＿ｉｎｄｅｘ＿ｉｄｃを明示的に信号通知または推測して記憶する。配列ＰａｌｅｔｔｅＩｎｄｅｘＩｄｃ［ｉ］のすべての要素を０に初期化する。
【０２８２】
ｃｏｐｙ＿ａｂｏｖｅ＿ｉｎｄｉｃｅｓ＿ｆｏｒ＿ｆｉｎａｌ＿ｒｕｎ＿ｆｌａｇが１の場合、水平トラバーススキャンが使用されている場合は上の行のパレットインデックスから、垂直トラバーススキャンが使用されている場合は左側の列のパレットインデックスから、符号化ユニットの最後の位置のパレットインデックスがコピーされることを指定する。
ｃｏｐｙ＿ａｂｏｖｅ＿ｉｎｄｉｃｅｓ＿ｆｏｒ＿ｆｉｎａｌ＿ｒｕｎ＿ｆｌａｇが０の場合、符号化ユニットの最後の位置のパレットインデックスがＰａｌｅｔｔｅＩｎｄｅｘＩｄｃ［ｎｕｍ＿ｐａｌｅｔｔｅ＿ｉｎｄｉｃｅｓ＿ｍｉｎｕｓ１］からコピーされることを指定する。
ｃｏｐｙ＿ａｂｏｖｅ＿ｉｎｄｉｃｅｓ＿ｆｏｒ＿ｆｉｎａｌ＿ｒｕｎ＿ｆｌａｇが存在しない場合、０と推測される。
【０２８３】
ｐａｌｅｔｔｅ＿ｔｒａｎｓｐｏｓｅ＿ｆｌａｇが１の場合、現在の符号化ユニットの画素のインデックスをスキャンするために垂直方向横断走査が適用されることを指定する。
ｐａｌｅｔｔｅ＿ｔｒａｎｓｐｏｓｅ＿ｆｌａｇが０の場合、現在の符号化ユニットの画素のインデックスをスキャンするために水平方向横断走査が適用されることを指定する。
【０２８４】
ｃｏｐｙ＿ａｂｏｖｅ＿ｐａｌｅｔｔｅ＿ｉｎｄｉｃｅｓ＿ｆｌａｇが１の場合、パレットインデックスは、水平方向横断走査が使用されている場合は上の行の同じ場所、垂直方向横断走査が使用されている場合は左側の列の同じ場所のパレットインデックスと等しいことを指定する。ｃｏｐｙ＿ａｂｏｖｅ＿ｐａｌｅｔｔｅ＿ｉｎｄｉｃｅｓ＿ｆｌａｇが０の場合、サンプルのパレットインデックスの指示がビットストリームに符号化されてい

るか、推測されているかを示す。
変数ＣｏｐｙＡｂｏｖｅＩｎｄｉｃｅｓＦｌａｇ［ｘＣ］［ｙＣ］が１の場合、パレットインデックスが上の行（水平スキャン）または左側の列（垂直スキャン）のパレットインデックスからコピーされることを指定する。ＣｏｐｙＡｂｏｖｅＩｎｄｉｃｅｓＦｌａｇ［ｘＣ］［ｙＣ］が０の場合、パレットインデックスがビットストリームで明示的に符号化されているか、推測されていることを示す。配列インデックスｘＣ、ｙＣは、ピクチャの左上の輝度サンプルに対するサンプルの位置（ｘＣ、ｙＣ）を指定する。
変数ＰａｌｅｔｔｅＩｎｄｅｘＭａｐ［ｘＣ］［ｙＣ］は、ＣｕｒｒｅｎｔＰａｌｅｔｔｅＥｎｔｒｉｅｓで表される配列へのインデックスであるパレットインデックスを指定する。配列インデックスｘＣ、ｙＣは、ピクチャの左上の輝度サンプルに対するサンプルの位置（ｘＣ、ｙＣ）を指定する。ＰａｌｅｔｔｅＩｎｄｅｘＭａｐ［ｘＣ］［ｙＣ］の値は、０からＭａｘＰａｌｅｔｔｅＩｎｄｅｘまでの範囲でなければならない。
変数ＰａｌｅｔｔｅＲｕｎは、ＣｏｐｙＡｂｏｖｅＩｎｄｉｃｅｓＦｌａｇ［ｘＣ］［ｙＣ］が１の場合、上の行（水平走査）または左側の列（垂直走査）の位置と同じパレットインデックスで、連続する位置の数から１を引いた数を指定し、ＣｏｐｙＡｂｏｖｅＩｎｄｉｃｅｓＦｌａｇ［ｘＣ］［ｙＣ］が０の場合、同じパレットインデックスで、連続する位置の数から１を引いた数を指定する。
変数ＰａｌｅｔｔｅＭａｘＲｕｎは、ＰａｌｅｔｔｅＲｕｎの可能な最大値を表す。ＰａｌｅｔｔｅＭａｘＲｕｎの値が０以上であることがビットストリーム適合性の要件である。

【０２８５】
ｐａｌｅｔｔｅ＿ｒｕｎ＿ｐｒｅｆｉｘは、ＰａｌｅｔｔｅＲｕｎの２値化のプレフィックス部分を指定する。

【０２８６】
ｐａｌｅｔｔｅ＿ｒｕｎ＿ｓｕｆｆｉｘは、ＰａｌｅｔｔｅＲｕｎの２値化のサフィックス部分を指定する。ｐａｌｅｔｔｅ＿ｒｕｎ＿ｓｕｆｆｉｘが存在しない場合、ｐａｌｅｔ＿ｒｕｎ＿ｓｕｆｆｉｘの値は０であると推測される。
ＰａｌｅｔｔｅＲｕｎの値は次のように導き出される。
ｐａｌｅｔｔｅ＿ｒｕｎ＿ｐｒｅｆｉｘが２未満の場合、以下が適用される。
ＰａｌｅｔｔｅＲｕｎ＝ｐａｌｅｔｔｅ＿ｒｕｎ＿ｐｒｅｆｉｘ（ＸＸ）
それ以外の場合（ｐａｌｅｔｔｅ＿ｒｕｎ＿ｐｒｅｆｉｘが２以上）、以下が適用される。
ＰｒｅｆｉｘＯｆｆｓｅｔ＝１＜＜（ｐａｌｅｔｔｅ＿ｒｕｎ＿ｐｒｅｆｉｘ－１）
ＰａｌｅｔｔｅＲｕｎ＝ＰｒｅｆｉｘＯｆｆｓｅｔ＋ｐａｌｅｔｔｅ＿ｒｕｎ＿ｓｕｆｆｉｘ　（ＸＸ）

【０２８７】
ｐａｌｅｔｔｅ＿ｅｓｃａｐｅ＿ｖａｌは、コンポーネントの量子化されたエスケープ符号化されたサンプル値を指定する。
変数ＰａｌｅｔｔｅＥｓｃａｐｅＶａｌ［ｃＩｄｘ］［ｘＣ］［ｙＣ］は、ＰａｌｅｔｔｅＩｎｄｅｘＭａｐ［ｘＣ］［ｙＣ］が（ＭａｘＰａｌｅｔｔｅＩｎｄｅｘ－１）であり、ｐａｌｅｔ＿ｅｓｃａｐｅ＿ｖａｌ＿ｐｒｅｓｅｎｔ＿ｆｌａｇが１であるサンプルのエスケープ値を指定する。配列インデックスｃＩｄｘは、色成分を指定する。配列インデックスｘＣ、ｙＣは、ピクチャの左上の輝度サンプルに対するサンプルの位置（ｘＣ、ｙＣ）を指定する。
ｃｌｄｘが０の場合、ＰａｌｅｔｔｅＥｓｃａｐｅＶａｌ［ｃＩｄｘ］［ｘＣ］［ｙＣ］０から（１＜＜（（１＜＜（ＢｉｔＤｅｐｔｈＹ＋１））－１の範囲内であり、ｃＩｄｘが０でない場合、０から（１＜＜（ＢｉｔＤｅｐｔｈＣ＋１））－１までの範囲内であることが、ビットストリーム適合性の要件である。

【０２８８】
イントラ予測モードで符号化されたユニットを符号化するための一般的な復号処理

【０２８９】
この処理への入力は以下の通りである。
－現在のピクチャの左上の輝度サンプルに対する現在の符号化ブロックの左上のサンプルを規定する輝度位置（ｘＣｂ，ｙＣｂ）、
－輝度サンプルにおける現在の符号化ブロックの幅を規定する変数ｃｂＷｉｄｔｈ、
－輝度サンプルにおける現在の符号化ブロックの高さを規定する変数ｃｂＨｅｉｇｈｔ
－単一ツリーを使用するか二重ツリーを使用するかを指定する変数ツリータイプ、および二重ツリーを使用する場合、現在のツリーが輝度成分に対応するか色度成分に対応するかを指定する。
【０２９０】
この処理の出力は、インループ・フィルタリング前の修正された再構成画像である。
輝度位置（ｘＣｂ，ｙＣｂ）、輝度サンプルｃｂＷｉｄｔｈにおける現在の符号化ブロックの幅、輝度サンプルｃｂＨｅｉｇｈｔにおける現在の符号化ブロックの高さ、及び変数ツリータイプを入力として、８．４．１項に規定される量子化パラメータの導出処理を呼び出す。
【０２９１】
ｔｒｅｅＴｙｐｅがＳＩＮＧＬＥ＿ＴＲＥＥに等しい場合、またはｔｒｅｅＴｙｐｅがＤＵＡＬ＿ＴＲＥＥ＿ＬＵＭＡに等しい場合、輝度サンプルの復号化処理は次のように指定される。
【０２９２】
－　ｐｃｍ＿ｆｌａｇ［ｘＣｂ］［ｙＣｂ］が１である場合、再構成画像は以下のように修正される。
　ＳＬ［ｘＣｂ＋ｉ］［ｙＣｂ＋ｊ］＝
　ｐｃｍ＿ｓａｍｐｌｅ＿ｌｕｍａ［（ｃｂＨｅｉｇｈｔ＊ｊ）＋ｉ］＜＜（ＢｉｔＤｅｐｔｈＹ－ＰｃｍＢｉｔＤｅｐｔｈＹ），（８－１）
　ｗｉｔｈ　ｉ＝０．．ｃｂＷｉｄｔｈ－１，ｊ＝０．．ｃｂＨｅｉｇｈｔ－１
【０２９３】
－　そうでない場合、ｃｕ＿ｐａｌｅｔｔｅ＿ｆｌａｇ［ｘＣｂ］［ｙＣｂ］が１と等しい場合、以下が適用される。
－ツリータイプがＳＩＮＧＬＥ＿ＴＲＥＥに等しい場合、以下が適用される。
　１．ＸＸＸ項で指定されているパレットブロックの一般的な復号化プロセスは、輝度位置（ｘＣｂ、ｙＣｂ）、変数ｓｔａｒｔＣｏｍｐを０、変数ｃＩｄｘを０、変数ｎＴｂＷをｃｂＷｉｄｔｈ、変数ｎＴｂＨをｃｂＨｅｉｇｈｔに設定して呼び出される。
－　そうでない場合、以下が適用される。
　１．　ＸＸＸ項で指定されているパレットブロックの一般的な復号化プロセスは、輝度位置（ｘＣｂ、ｙＣｂ）、変数ｓｔａｒｔＣｏｍｐを０、変数ｃＩｄｘを０、変数ｎＴｂＷをｃｂＷｉｄｔｈ、変数ｎＴｂＨをｃｂＨｅｉｇｈｔに設定して呼び出される。
　２．　８．Ｘ．Ｘ項で指定されているパレットブロックの一般的なパレット予測子更新処理は、輝度位置（ｘＣｂ、ｙＣｂ）、変数ｓｔａｒｔＣｏｍｐを０、変数ｎｕｍＣｏｍｐｓを１に設定して呼び出される。
【０２９４】
－　そうでない場合、以下が適用される。
　１．　８．２．２項で指定されている輝度イントラ予測モードの導出処理は、輝度位置（ｘＣｂ、ｙＣｂ）、輝度サンプルの現在の符号化ブロックの幅ｃｂＷｉｄｔｈ、および輝度サンプルの現在の符号化ブロックの高さｃｂＨｅｉｇｈｔを入力として使用して呼び出される。
　２．　８．２．４．１項で指定されているイントラブロックの一般的な復号化処理は、輝度位置（ｘＣｂ、ｙＣｂ）、ツリータイプはｔｒｅｅＴｙｐｅ、変数ｎＴｂＷはｃｂＷｉｄｔｈに等しく、変数ｎＴｂＨはｃｂＨｅｉｇｈｔに等しく、変数ｐｒｅｄＭｏｄｅＩｎｔｒａはＩｎｔｒａＰｒｅｄＭｏｄｅＹ［ｘＣｂ］［ｙＣｂ］に等しく、変数ｃＩｄｘ

は０に等しく、入力設定して呼び出され、出力はループ内フィルタリングの前の修正された再構成ピクチャである。

【０２９５】

ｔｒｅｅＴｙｐｅがＳＩＮＧＬＥ＿ＴＲＥＥに等しい場合、またはｔｒｅｅＴｙｐｅがＤＵＡＬ＿ＴＲＥＥ＿ＣＨＲＯＭＡに等しい場合、クロマサンプルの復号化処理は次のように指定される。

【０２９６】

－ ｐｃｍ＿ｆｌａｇ［ｘＣｂ］［ｙＣｂ］が１に等しい場合、再構成されたピクチャは次のように修正される。

S_{Cb}［ｘＣｂ／ＳｕｂＷｉｄｔｈＣ＋ｉ］［ｙＣｂ／ＳｕｂＨｅｉｇｈｔＣ＋ｊ］＝ｐｃｍ＿ｓａｍｐｌｅ＿ｃｈｒｏｍａ［（ｃｂＨｅｉｇｈｔ／ＳｕｂＷｉｄｔｈＣ＊ｊ）＋ｉ］＜＜（$BitDepth_C$－$PcmBitDepth_C$），

ｗｉｔｈ ｉ＝０．．ｃｂＷｉｄｔｈ／ＳｕｂＷｉｄｔｈＣ－１ ａｎｄ ｊ＝０．．ｃｂＨｅｉｇｈｔ／ＳｕｂＨｅｉｇｈｔＣ－１ （８－２）

S_{Cr}［ｘＣｂ／ＳｕｂＷｉｄｔｈＣ＋ｉ］［ｙＣｂ／ＳｕｂＨｅｉｇｈｔＣ＋ｊ］＝ｐｃｍ＿ｓａｍｐｌｅ＿ｃｈｒｏｍａ［（ｃｂＨｅｉｇｈｔ／ＳｕｂＷｉｄｔｈＣ＊（ｊ＋ｃｂＨｅｉｇｈｔ／ＳｕｂＨｅｉｇｈｔＣ））＋ｉ］＜＜

（$BitDepth_C$－$PcmBitDepth_C$），

ｗｉｔｈ ｉ＝０．．ｃｂＷｉｄｔｈ／ＳｕｂＷｉｄｔｈＣ－１ ａｎｄ ｊ＝０．．ｃｂＨｅｉｇｈｔ／ＳｕｂＨｅｉｇｈｔＣ－１ （８－３）

【０２９７】

－ そうでない場合、ｃｕ＿ｐａｌｅｔｔｅ＿ｆｌａｇ［ｘＣｂ］［ｙＣｂ］が１と等しい場合、以下が適用される。

－ ツリータイプがＳＩＮＧＬＥ＿ＴＲＥＥに等しい場合、以下が適用される。

１． ＸＸＸ項で指定されているパレットブロックの一般的な復号化処理は、輝度位置（ｘＣｂ、ｙＣｂ）、変数ｓｔａｒｔＣｏｍｐを０、変数ｃＩｄｘを１、変数ｎＴｂＷを（ｃｂＷｉｄｔｈ／２）、変数（ｃｂＨｅｉｇｈｔ／２）をｃｂＨｅｉｇｈｔと設定して呼び出される。

２． ＸＸＸ項で指定されているパレットブロックの一般的な復号化処理は、輝度位置（ｘＣｂ、ｙＣｂ）、変数ｓｔａｒｔＣｏｍｐを０、変数ｃＩｄｘを２、変数ｎＴｂＷを（ｃｂＷｉｄｔｈ／２）、変数（ｃｂＨｅｉｇｈｔ／２）をｃｂＨｅｉｇｈｔと設定して呼び出される。

３． ８．Ｘ．Ｘ項で指定されているパレットブロックの一般的なパレット予測子更新プロセスは、輝度位置（ｘＣｂ、ｙＣｂ）、変数ｓｔａｒｔＣｏｍｐを０、変数ｎｕｍＣｏｍｐｓを３に設定して呼び出される。

－ そうでない場合、以下が適用される。

１． ＸＸＸ項で指定されているパレットブロックの一般的な復号化処理は、輝度位置（ｘＣｂ、ｙＣｂ）、変数ｓｔａｒｔＣｏｍｐを１、変数ｃＩｄｘを１、変数ｎＴｂＷを（ｃｂＷｉｄｔｈ／２）、変数（ｃｂＨｅｉｇｈｔ／２）をｃｂＨｅｉｇｈｔと設定して呼び出される。

２． ＸＸＸ項で指定されているパレットブロックの一般的な復号化処理は、輝度位置（ｘＣｂ、ｙＣｂ）、変数ｓｔａｒｔＣｏｍｐを１、変数ｃＩｄｘを２、変数ｎＴｂＷを（ｃｂＷｉｄｔｈ／２）、変数（ｃｂＨｅｉｇｈｔ／２）をｃｂＨｅｉｇｈｔと設定して呼び出される。

３． ＸＸＸ項で指定されているパレットブロックの一般的なパレット予測子更新プロセスは、輝度位置（ｘＣｂ、ｙＣｂ）、変数ｓｔａｒｔＣｏｍｐを１、変数ｎｕｍＣｏｍｐｓを２と設定して呼び出される。

【０２９８】

－ そうでない場合、以下が適用される。

１． ８．２．３項で指定されているクロマイントラ予測モードの導出処理は、輝度位

置（ｘＣｂ、ｙＣｂ）、輝度サンプルの現在の符号化ブロックの幅ｃｂＷｉｄｔｈ、および輝度サンプルの現在の符号化ブロックの高さｃｂＨｅｉｇｈｔを入力として呼び出される。

２．　８．２．４．１項で指定されているイントラブロックの一般的な復号化処理は、クロマ位置（ｘＣｂ／２、ｙＣｂ／２）、ツリータイプはｔｒｅｅＴｙｐｅ、変数ｎＴｂＷは（ｃｂＷｉｄｔｈ／２）に等しく、変数ｎＴｂＨは（ｃｂＨｅｉｇｈｔ／２）に等しく、変数ｐｒｅｄＭｏｄｅＩｎｔｒａがＩｎｔｒａＰｒｅｄＭｏｄｅＣ［ｘＣｂ］［ｙＣｂ］に等しく、変数ｃＩｄｘは１に等しく設定して呼び出され、出力はループ内フィルタリングの前に修正された再構成ピクチャである。

３．　８．２．４．１項で指定されているイントラブロックの一般的な復号化処理は、クロマ位置（ｘＣｂ／２、ｙＣｂ／２）、ツリータイプはｔｒｅｅＴｙｐｅ、変数ｎＴｂＷは（ｃｂＷｉｄｔｈ／２）に等しく、変数ｎＴｂＨは　（ｃｂＨｅｉｇｈｔ／２）に等しく、変数ｐｒｅｄＭｏｄｅＩｎｔｒａがＩｎｔｒａＰｒｅｄＭｏｄｅＣ［ｘＣｂ］［ｙＣｂ］に等しく、変数ｃＩｄｘは２に等しく設定して呼び出され、出力は、ループ内フィルタリングの前に修正された再構成ピクチャである。

【０２９９】

パレットモードの復号化処理

【０３００】

この処理への入力は以下の通りである。

－　現在のピクチャの左上の輝度サンプルに対する現在のブロックの左上のサンプルを規定する位置（ｘＣｂ，ｙＣｂ）

－　変数ｓｔａｒｔＣｏｍｐは、パレットテーブルの最初の色成分を規定し、

－　現在のブロックの色成分を規定する変数ｃＩｄｘ、

－　現在のブロックの幅と高さをそれぞれ指定する２つの変数ｎＴｂＷとｎＴｂＨ。

【０３０１】

このプロセスの出力は、配列ｒｅｃＳａｍｐｌｅｓ［ｘ］［ｙ］であり、ｘ＝０．．ｎＴｂＷ－１、ｙ＝０．．ｎＴｂＨ－１は、ブロックの再構築されたサンプル値を指定する。ｃＩｄｘの値に基づいて、変数ｎＳｕｂＷｉｄｔｈおよびｎＳｕｂＨｅｉｇｈｔは、以下のように導出される。

【０３０２】

－　ｃＩｄｘが０に等しい場合、ｎＳｕｂＷｉｄｔｈは１に設定され、ｎＳｕｂＨｅｉｇｈｔは１に設定される。

－　ｓｔａｒｔＣｏｍｐが１に等しく、ｃＩｄｘ＞１の場合、ｎＳｕｂＷｉｄｔｈは１に設定され、ｎＳｕｂＨｅｉｇｈｔは１に設定される。

－　あるいは、ｎＳｕｂＷｉｄｔｈをＳｕｂＷｉｄｔｈＣに設定し、ｎＳｕｂＨｅｉｇｈｔをＳｕｂＨｅｉｇｈｔＣに設定する。

－　ｃｕ＿ｐａｌｅｔｔｅ＿ｉｂｃ＿ｍｏｄｅ［ｘＣｂ］［ｙＣｂ］が１に等しい場合、以下が適用される。

－　８．６．２項に規定される動きベクトルの導出処理は、現在の画像の左上の輝度サンプルに対する現在の輝度符号化ブロックの左上のサンプルの輝度位置（ｘＣｂ，ｙＣｂ）、輝度サンプルにおける現在の符号化ブロックの幅を指定する変数ｃｂＷｉｄｔｈ、及び輝度サンプルにおける現在の符号化ブロックの高さを指定する変数ｃｂＨｅｉｇｈｔによって呼び出される。出力は、－１／１６分数サンプル精度ｍｖＬにおける輝度動きベクトルである。

－　８．６．３項で指定されている一般的なＩＢＣ予測プロセスは、現在のピクチャの左上の輝度サンプルに対する現在の符号化ブロックの左上のサンプルを指定する輝度位置（ｘＣｂ、ｙＣｂ）、輝度サンプルの現在の符号化ブロックの幅を指定する変数ｃｂＷｉｄｔｈ、輝度サンプルの現在の符号化ブロックの高さを指定する変数ｃｂＨｅｉｇｈｔ、水平および垂直方向の輝度符号化サブブロックの数を指定する変数ｎｕｍＳｂＸおよびｎｕｍＳｂＹ、動きベクトルｍｖ［ｘＳｂＩｄｘ］［ｙＳｂＩｄｘ］、ｘＳｂＩｄｘ＝０．

．ｎｕｍＳｂＸ－１、およびｙＳｂＩｄｘ＝０．．ｎｕｍＳｂＹ－１、現在のブロックの色成分インデックスを指定する変数ｃＩｄｘで、呼び出される。出力は、予測サンプルｐｒｅｄＳａｍｐｌｅｓ［ｘ］［ｙ］の配列である。

【０３０３】

位置（ｘＣｂ、ｙＣｂ）にある再構成されたサンプル配列ｒｅｃＳａｍｐｌｅｓの（ｎＴｂＷ　ｘｎＴｂＨ）ブロックは、ｘ＝０．．ｎＴｂＷ－１、ｙ＝０．．ｎＴｂＨ－１のｒｅｃＳａｍｐｌｅｓ［ｘ］［ｙ］で表され、０からｎＴｂＷ－１までの範囲の各ｘと、０からｎＴｂＨ－１までの範囲の各ｙに対するｒｅｃＳａｍｐｌｅｓ［ｘ］［ｙ］の値は、次のように導出される。

【０３０４】

－変数ｘＬおよびｙＬは、以下のように導出される。

ｘＬ＝ｐａｌｅｔｔｅ＿ｔｒａｎｓｐｏｓｅ＿ｆｌａｇ　？　ｘ＊ｎＳｕｂＨｅｉｇｈｔ：ｘ＊ｎＳｕｂＷｉｄｔｈ　（５－４）

ｙＬ＝ｐａｌｅｔｔｅ＿ｔｒａｎｓｐｏｓｅ＿ｆｌａｇ　？　ｙ＊ｎＳｕｂＷｉｄｔｈ：ｙ＊ｎＳｕｂＨｅｉｇｈｔ　（５－５）

－変数ｂＩｓＥｓｃａｐｅＳａｍｐｌｅは、以下のように導出される。

－ＰａｌｅｔｔｅＩｎｄｅｘＭａｐ［ｘＣｂ＋ｘＬ］［ｙＣｂ＋ｙＬ］がＭａｘＰａｌｅｔｔｅＩｎｄｅｘに等しく、ｐａｌｅｔｔｅ＿ｅｓｃａｐｅ＿ｖａｌ＿ｐｒｅｓｅｎｔ＿ｆｌａｇが１に等しい場合、ｂＩｓＥｓｃａｐｅＳａｍｐｌｅは１に設定される。

－あるいは、ｂＩｓＥｓｃａｐｅＳａｍｐｌｅを０に設定する。

－ｂＩｓＥｓｃａｐｅＳａｍｐｌｅが０に等しい場合、以下が適用される。

－ｃｕ＿ｐａｌｅｔｔｅ＿ｉｂｃ＿ｍｏｄｅ　［ｘＣｂ］　［ｙＣｂ］が１に等しい場合、以下が適用される。

－ＰａｌｅｔｔｅＩｎｄｅｘＭａｐ［ｘＣｂ＋ｘＬ］［ｙＣｂ＋ｙＬ］が０に等しい場合、以下が適用される。

ｒｅｃＳａｍｐｌｅｓ［ｘ］［ｙ］＝ｐｒｅｄＳａｍｐｌｅｓ［ｘ］［ｙ］

－そうでない場合、以下が適用される。

ｒｅｃＳａｍｐｌｅｓ［ｘ］［ｙ］＝ＣｕｒｒｅｎｔＰａｌｅｔｔｅＥｎｔｒｉｅｓ［ｃＩｄｘ］［ＰａｌｅｔｔｅＩｎｄｅｘＭａｐ［ｘＣｂ＋ｘＬ］［ｙＣｂ＋ｙＬ］－１］　（５－６）

－そうでない場合、以下が適用される。

ｒｅｃＳａｍｐｌｅｓ［ｘ］［ｙ］＝ＣｕｒｒｅｎｔＰａｌｅｔｔｅＥｎｔｒｉｅｓ［ｃＩｄｘ］［ＰａｌｅｔｔｅＩｎｄｅｘＭａｐ［ｘＣｂ＋ｘＬ］［ｙＣｂ＋ｙＬ］］　（５－７）

－あるいは、ｃｕ＿ｔｒａｎｓｑｕａｎｔ＿ｂｙｐａｓｓ＿ｆｌａｇが１に等しい場合には、以下が適用される。

ｒｅｃＳａｍｐｌｅｓ［ｘ］［ｙ］＝ＰａｌｅｔｔｅＥｓｃａｐｅＶａｌ［ｃＩｄｘ］［ｘＣｂ＋ｘＬ］［ｙＣｂ＋ｙＬ］　（５－８）

－あるいは、（ｂＩｓＥｓｃａｐｅＳａｍｐｌｅが１に等しく、ｃｕ＿ｔｒａｎｓｑｕａｎｔ＿ｂｙｐａｓｓ＿ｆｌａｇが０に等しい場合）、以下の順序付けられたステップが適用される。

１．量子化パラメータの導出処理は、現在のピクチャの左上サンプルに対する現在のブロックの左上サンプルを規定する位置（ｘＣｂ，ｙＣｂ）を使用して呼び出される。

２．量子化パラメータｑＰは、以下のように導出される。

－　ｃＩｄｘが０に等しい場合

ｑＰ＝Ｍａｘ（０，Ｑｐ’Ｙ）　（５－９）’

－　あるいは、ｃＩｄｘが１に等しい場合、

ｑＰ＝Ｍａｘ（０，Ｑｐ’Ｃｂ）　（５－１０）

－あるいは、（ｃＩｄｘが２に等しい場合）、

ｑＰ＝Ｍａｘ（０，Ｑｐ’Ｃｒ）　（５－１１）

３．変数ｂｉｔＤｅｐｔｈは、以下のように導出される。
ｂｉｔＤｅｐｔｈ＝（ｃＩｄｘ＝＝０ ）　？ＢｉｔＤｅｐｔｈＹ：ＢｉｔＤｅｐｔｈｃ　（５－１２）
４．ｌｉｓｔ　ｌｅｖｅｌＳｃａｌｅ［］は、ｋ＝０．．５の時、ｌｅｖｅｌＳｃａｌｅ［ｋ］＝｛４０，４５，５１，５７，６４，７２｝として規定される。
５．　以下が適用される
ｔｍｐＶａｌ＝（ＰａｌｅｔｔｅＥｓｃａｐｅＶａｌ［ｃＩｄｘ］［ｘＣｂ＋ｘＬ］［ｙＣｂ＋ｙＬ］＊
ｌｅｖｅｌＳｃａｌｅ［ｑＰ％６］）＜＜（ｑＰ／６）＋３２）＞＞６　（５－１３）
ｒｅｃＳａｍｐｌｅｓ［ｘ］［ｙ］＝Ｃｌｉｐ３（０，（１＜＜ｂｉｔＤｅｐｔｈ）－１，ｔｍｐＶａｌ）　（５－１４）

【０３０５】
パレットモードのパレット予測子更新プロセス

【０３０６】
この処理への入力は以下の通りである。
－　現在のピクチャの左上の輝度サンプルに対する現在のブロックの左上のサンプルを規定する位置（ｘＣｂ，ｙＣｂ）
－　変数ｓｔａｒｔＣｏｍｐは、パレットテーブルの最初の色成分を規定し、
－　変数ｎｕｍＣｏｍｐｓは、パレットテーブル内の色成分の数を指定する。

【０３０７】
この処理の出力は、更新されたパレット予測子のサイズを指定する変数ＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅ［ｓｔａｒｔＣｏｍｐ］であり、ｃＩｄｘ＝ｓｔａｒｔＣｏｍｐ、…、ｓｔａｒｔＣｏｍｐ＋ｎｕｍＣｏｍｐｓ－１、ｉ＝０の配列ＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＥｎｔｒｉｅｓ［ｃＩｄｘ］［ｉ］である。ＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅ－１は、次のブロックの更新されたパレット予測子を指定する。

【０３０８】
変数ＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅおよび配列ＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＥｎｔｒｉｅｓは、以下のように導出または修正される。

【０３０９】
```
for（ｉ＝０；ｉ＜ＣｕｒｒｅｎｔＰａｌｅｔｔｅＳｉｚｅ；ｉ＋＋）
 for（ｃＩｄｘ＝ｓｔａｒｔＣｏｍｐ；ｃＩｄｘ＜（ｓｔａｒｔＣｏｍｐ＋ｎｕｍＣｏｍｐｓ）；ｃＩｄｘ＋＋）
  ｎｅｗＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＥｎｔｒｉｅｓ［ｃＩｄｘ］［ｉ］＝ＣｕｒｒｅｎｔＰａｌｅｔｔｅＥｎｔｒｉｅｓ［ｃＩｄｘ］［ｉ］
ｎｅｗＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅ＝ＣｕｒｒｅｎｔＰａｌｅｔｔｅＳｉｚｅ
for（ｉ＝０；ｉ＜ＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅ ＆＆ ｎｅｗＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅ＜ＰａｌｅｔｔｅＭａｘＰｒｅｄｉｃｔｏｒＳｉｚｅ；ｉ＋＋）
 if（！ＰａｌｅｔｔｅＰｒｅｄｉｃｔｏｒＥｎｔｒｙＲｅｕｓｅＦｌａｇｓ［ｉ］）｛
  for（ｃＩｄｘ＝ｓｔａｒｔＣｏｍｐ；ｃＩｄｘ＜（ｓｔａｒｔＣｏｍｐ＋ｎｕｍＣｏｍｐｓ）；ｃＩｄｘ＋＋）　（５－１５）
   ｎｅｗＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＥｎｔｒｉｅｓ［ｃＩｄｘ］［ｎｅｗＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅ］＝
    ＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＥｎｔｒｉｅｓ［ｃＩｄｘ］［ｉ］
  ｎｅｗＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅ＋＋
 ｝
for（ｃＩｄｘ＝ｓｔａｒｔＣｏｍｐ；ｃＩｄｘ＜（ｓｔａｒｔＣｏｍｐ＋ｎｕｍＣｏ
```

ｍｐｓ）；ｃＩｄｘ＋＋）
　ｆｏｒ（ｉ＝０；ｉ＜ｎｅｗＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅ；ｉ＋＋）
　　ＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＥｎｔｒｉｅｓ［ｃＩｄｘ］［ｉ］＝ｎｅｗＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＥｎｔｒｉｅｓ［ｃＩｄｘ］［ｉ］
ＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅ＝ｎｅｗＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅ
【０３１０】
ＰｒｅｄｉｃｔｏｒＰａｌｅｔｔｅＳｉｚｅの値が０からＰａｌｅｔｔｅＭａｘＰｒｅｄｉｃｔｏｒＳｉｚｅまでの範囲内にあることが、ビットストリーム適合性の要件である。
【０３１１】
５．１　実施形態＃２
【０３１２】
５．１．１　変換係数のスケーリング処理
【０３１３】
この処理への入力は以下の通りである。
－　現在のピクチャの左上の輝度サンプルに対する現在の輝度変換ブロックの左上のサンプルを規定する輝度位置（ｘＴｂＹ，ｙＴｂＹ）、
－　変換ブロックの幅を指定する変数ｎＴｂＷ、
－　変換ブロックの高さを指定する変数ｎＴｂＨ、
－　現在のブロックの色成分を規定する変数ｃＩｄｘ、
－　現在の色成分のビット深度を指定する変数ｂｉｔＤｅｐｔｈ。
【０３１４】
この処理の出力は、要素ｄ［ｘ］［ｙ］を有するスケーリングされた変換係数の（ｎＴｂＷ）×（ｎＴｂＨ）配列ｄである。量子化パラメータｑＰは、以下のように導出される。
－ｃＩｄｘが０に等しい場合、以下が適用される。
　－　ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇ［ｘＴｂＹ］［ｙＴｂＹ］が１である、またはｉｎｔｒａ＿ｂｄｐｃｍ＿ｆｌａｇ［ｘＴｂＹ］［ｙＴｂＹ］が１である場合、以下が適用される。
　　ｑＰ＝Ｍａｘ（４＋ＱｐＢｄＯｆｆｓｅｔＹ，Qp'_Y）　（８－９５４）
　－　そうでない場合、
　　ｑＰ＝Qp'_Y　（８－９５４）
－　そうでない場合、ｃＩｄｘが１に等しく、ｔｕ＿ｊｏｉｎｔ＿ｃｂｃｒ＿ｒｅｓｉｄｕａｌ［ｘＴｂＹ］［ｙＴｂＹ］が１である場合、以下が適用される。
　ｑＰ＝Qp'_{CbCr}　（８－９５５）
－　そうでない場合、ｃＩｄｘが１である場合、以下が適用される。
　ｑＰ＝Qp'_{Cb}　（８－９５６）
－　そうでない場合（ｃＩｄｘ＝２）、以下が適用される。
ｑＰ＝Qp'_{Cr}
【０３１５】
一例において、強調表示された条件は、以下のようにさらに変更されてもよい。
ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇ［ｘＴｂＹ］［ｙＴｂＹ］が１である場合、以下が適用される。
【０３１６】
５．２　実施形態＃３
【０３１７】
５．２．１　変換係数のスケーリング処理
【０３１８】
この処理への入力は以下の通りである。
－　現在のピクチャの左上の輝度サンプルに対する現在の輝度変換ブロックの左上のサンプルを規定する輝度位置（ｘＴｂＹ，ｙＴｂＹ）、

－　変換ブロックの幅を指定する変数ｎＴｂＷ、
－　変換ブロックの高さを指定する変数ｎＴｂＨ、
－　現在のブロックの色成分を規定する変数ｃＩｄｘ、
－　現在の色成分のビット深度を指定する変数ｂｉｔＤｅｐｔｈ。
【０３１９】
この処理の出力は、要素ｄ［ｘ］［ｙ］を有するスケーリングされた変換係数の（ｎＴｂＷ）×（ｎＴｂＨ）配列ｄである。量子化パラメータｑＰは、以下のように導出される。
－ｃＩｄｘが０に等しい場合、以下が適用される。
　－　ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇ［ｘＴｂＹ］［ｙＴｂＹ］が１である、またはｉｎｔｒａ＿ｂｄｐｃｍ＿ｆｌａｇ［ｘＴｂＹ］［ｙＴｂＹ］が１である場合、以下が適用される。
　　ｑＰ＝Ｍａｘ（４＋（６＊（ｉｎｐｕｔ＿ｂｉｔ＿ｄｅｐｔｈ－８）），Ｑｐ’$_{Y}$）　（８－９５４）
　－　そうでない場合、
　　ｑＰ＝Ｑｐ’$_{Y}$　（８－９５４）
－　そうでない場合、ｃＩｄｘが１に等しく、ｔｕ＿ｊｏｉｎｔ＿ｃｂｃｒ＿ｒｅｓｉｄｕａｌ［ｘＴｂＹ］［ｙＴｂＹ］が１である場合、以下が適用される。
　　ｑＰ＝Ｑｐ’$_{CbCr}$　（８－９５５）
－　そうでない場合、ｃＩｄｘが１である場合、以下が適用される。
　　ｑＰ＝Ｑｐ’$_{Cb}$　（８－９５６）
－　そうでない場合（ｃＩｄｘ＝２）、以下が適用される。
ｑＰ＝Ｑｐ’$_{Cr}$
【０３２０】
一例において、強調表示された条件は、以下のようにさらに変更されてもよい。
ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇ［ｘＴｂＹ］［ｙＴｂＹ］が１である場合、以下が適用される。
【０３２１】
５．３　実施形態＃４
【０３２２】
５．３．１　変換係数のスケーリング処理この処理への入力は以下の通りである。
【０３２３】
－　現在のピクチャの左上の輝度サンプルに対する現在の輝度変換ブロックの左上のサンプルを規定する輝度位置（ｘＴｂＹ，ｙＴｂＹ）、
－　変換ブロックの幅を指定する変数ｎＴｂＷ、
－　変換ブロックの高さを指定する変数ｎＴｂＨ、
－　現在のブロックの色成分を規定する変数ｃＩｄｘ、
－　現在の色成分のビット深度を指定する変数ｂｉｔＤｅｐｔｈ。
【０３２４】
この処理の出力は、要素ｄ［ｘ］［ｙ］を有するスケーリングされた変換係数の（ｎＴｂＷ）×（ｎＴｂＨ）配列ｄである。量子化パラメータｑＰは、以下のように導出される。
－ｃＩｄｘが０に等しい場合、以下が適用される。
　－　ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇ［ｘＴｂＹ］［ｙＴｂＹ］が１である、またはｉｎｔｒａ＿ｂｄｐｃｍ＿ｆｌａｇ［ｘＴｂＹ］［ｙＴｂＹ］が１である場合、以下が適用される。
　　ｑＰ＝Ｍａｘ（４，Ｑｐ’$_{Y}$）　（８－９５４）
　－　そうでない場合、
　　ｑＰ＝Ｑｐ’$_{Y}$　（８－９５４）
－　そうでない場合、ｃＩｄｘが１に等しく、ｔｕ＿ｊｏｉｎｔ＿ｃｂｃｒ＿ｒｅｓｉｄｕａｌ［ｘＴｂＹ］［ｙＴｂＹ］が１である場合、以下が適用される。
　　ｑＰ＝Ｑｐ’$_{CbCr}$　（８－９５５）

－　そうでない場合、ｃＩｄｘが１である場合、以下が適用される。

ｑＰ＝Ｑｐ’$_{Cb}$　（８－９５６）

－　そうでない場合（ｃＩｄｘ＝２）、以下が適用される。

ｑＰ＝Ｑｐ’$_{Cr}$

【０３２５】

一例において、強調表示された条件は、以下のようにさらに変更されてもよい。

ｔｒａｎｓｆｏｒｍ＿ｓｋｉｐ＿ｆｌａｇ［ｘＴｂＹ］［ｙＴｂＹ］が１である場合、以下が適用される。

【０３２６】

図１７Ａは、映像処理装置１７００のブロック図である。装置１７００は、本明細書に記載の方法の１つ以上を実装するために使用してもよい。装置１７００は、スマートフォン、タブレット、コンピュータ、モノのインターネット（ＩｏＴ）受信機等に実施されてもよい。装置１７００は、１つ以上の処理装置１７０２と、１つ以上のメモリ１７０４と、映像処理ハードウェア１７０６と、を含んでもよい。１つまたは複数の処理装置１７０２は、本明細書に記載される１つ以上の方法を実装するように構成されてもよい。メモリ（複数可）１７０４は、本明細書で説明される方法および技術を実装するために使用されるデータおよびコードを記憶するために使用してもよい。映像処理ハードウェア１７０６は、本明細書に記載される技術をハードウェア回路にて実装するために使用してもよい。映像処理ハードウェア１７０６は、専用ハードウェア、またはグラフィカル処理装置ユニット（ＧＰＵ）若しくは専用信号処理ブロックの形式で、処理装置１７０２内に部分的にまたは完全に含まれてもよい。

【０３２７】

図１７Ｂは、開示された技術を実装することができる例示的な映像処理システムを示すブロック図の別の例である。図１７Ｂは、本明細書で開示される様々な技術が実装され得る例示的な映像処理システム１７１０を示すブロック図である。様々な実装形態は、システム１７１０のコンポーネントの一部または全部を含んでもよい。システム１７１０は、映像コンテンツを受信するための入力ユニット１７１２を含んでもよい。映像コンテンツは、未加工または非圧縮フォーマット、例えば、８または１０ビットのマルチコンポーネント画素値で受信されてもよく、または圧縮または符号化フォーマットで受信されてもよい。入力ユニット１７１２は、ネットワークインターフェース、周辺バスインターフェース、または記憶インターフェースを表してもよい。ネットワークインターフェースの例は、イーサネット（登録商標）、パッシブ光ネットワーク（ＰＯＮ）等の有線インターフェース、およびＷｉ－Ｆｉ（登録商標）またはセルラーインターフェース等の無線インターフェースを含む。

【０３２８】

システム１７１０は、本明細書に記載される様々な符号化または符号化方法を実装することができる符号化コンポーネント１７１４を含んでもよい。符号化コンポーネント１７１４は、入力ユニット１７１２から符号化コンポーネント１７１４の出力への映像の平均ビットレートを低減して、映像の符号化表現を生成してもよい。従って、この符号化技術は、映像圧縮または映像コード変換技術と呼ばれることがある。符号化コンポーネント１７１４の出力は、コンポーネント１７１６によって表されるように、記憶されてもよいし、接続された通信を介して送信されてもよい。入力ユニット１７１２において受信された、記憶されたまたは通信された映像のビットストリーム（または符号化）表現は、コンポーネント１７１８によって使用されて、表示インターフェースユニット１７２０に送信される画素値または表示可能な映像を生成してもよい。ビットストリーム表現からユーザが見ることができる映像を生成する処理は、映像伸張（映像展開）と呼ばれることがある。さらに、特定の映像処理動作を「符号化」動作またはツールと呼ぶが、符号化ツールまたは動作は、エンコーダで使用され、対応する復号化ツール又は動作であり符号化の結果を逆にするものは、デコーダによって行われることが理解されよう。

【０３２９】

周辺バスインターフェースユニットまたは表示インターフェースユニットの例は、ユニバーサルシリアルバス（ＵＳＢ）または高精細マルチメディアインターフェース（ＨＤＭＩ（登録商標））またはディスプレイポート等を含んでもよい。ストレージインターフェースの例は、シリアルアドバンスドテクノロジーアタッチメント（ＳＡＴＡ）、ＰＣＩ、ＩＤＥインターフェース等を含む。本明細書に記載される技術は、携帯電話、ノートパソコン、スマートフォン、またはデジタルデータ処理及び／または映像表示を実施可能な他のデバイス等の様々な電子デバイスに実施されてもよい。

【０３３０】

図１８Ａは、映像処理のための例示的な方法１８１０を示す。方法１８１０は、ステップ１８１２において、クロマブロックであり、クロマＤＭモードで符号化された映像の現在の映像ブロックに対して、前記現在の映像ブロックのために、前記現在の映像ブロックの符号化モードに関する規則に従って、イントラ予測モードを導出することを含む。方法１８１０は、ステップ１８１４において、映像の現在の映像ブロックと符号化表現との間で変換を行うことをさらに含む。いくつかの実装形態において、前記規則は、クロマＤＭモードの導出に使用された対応する輝度ブロックのイントラ予測方向に基づいてイントラ予測モードを導出し、現在のビデオブロックに対応する量子化残差ブロックのブロック差分パルス符号変調表現を用いて符号化することを規定する。

【０３３１】

図１８Ｂは、映像処理のための例示的な方法１８１０を示す。この方法１８１０は、ステップ１８２２において、映像の現在の映像ブロックと映像の符号化表現との間で変換を行うことを含む。いくつかの実装形態において、現在の映像ブロックに対応する量子化残差ブロックのブロック差動パルス符号変調表現を使用して、この符号化表現が、この符号化表現において表されるこの現在の映像ブロックのための非ゼロ係数を含むかどうかを示すために、１つのフラグを決定する。

【０３３２】

いくつかの実装形態において、映像の現在の映像ブロックはクロマブロックである。いくつかの実装形態において、現在の映像ブロックの対応する輝度映像ブロックが非イントラモードで符号化されるため、現在の映像ブロックのための予測ブロックを生成するためのイントラ予測方法が使用される。いくつかの実装形態において、現在の映像ブロックのためのクロマＤＭモードは、クロマＤＭモードの導出に用いられ、イントラモードを用いて符号化されない、対応する輝度ブロックによって無効化される。いくつかの実装形態において、規則に従って変換が行われ、現在の映像ブロックは、対応する輝度ブロックの符号化された情報から予測ブロックのためのモードを導出する導出モードを使用して計算された予測ブロックを使用して、符号化表現で表現され、この規則は、導出されたモードを使用することによって、現在の映像ブロックが無効なブロックベクトルを有すると仮定して、予測ブロックを決定することを指定する。

【０３３３】

図１８Ｃは、映像処理のための例示的な方法１８３０を示す。方法１８３０は、ステップ１８３２において、映像のピクチャの現在の映像ブロックと映像の符号化表現との変換のために、現在の映像ブロックの特徴に基づいて、現在の映像ブロックに対してイントラブロックコピー（ＩＢＣ）モードを使用することを示す構文要素のための１つ以上のコンテクストを構成することを含む。方法１８３０は、ステップ１８３４において、この構成に基づいて、この変換を実行することをさらに含み、ＩＢＣモードを用いることは、ピクチャにおける他の映像ブロックの予め符号化されたサンプルを用いて、現在の映像ブロックのサンプル値を予測することを含む。

【０３３４】

図１８Ｄは、映像処理のための例示的な方法１８４０を示す。方法１８４０は、ステップ１８４２において、映像の現在の映像ブロックと前記映像の符号化表現との変換のために、前記現在の映像ブロックの近傍のブロックに関連付けられた第２の構文要素に基づいて、第１の構文要素に関連付けられたビンを符号化するためのコンテクストモデルを構成

することを含む。方法１８４０は、ステップ１８４４において、この構成に基づいて変換を行うことをさらに含む。

【０３３５】

図１８Ｅは、映像処理のための例示的な方法１８５０を示す。方法１８５０は、ステップ１８５２において、映像の現在の映像ブロックとこの映像の符号化表現との変換のために、規則に従って現在の映像ブロックのデフォルトのイントラモードを決定することを含む。方法１８５０は、ステップ１８５４において、このデフォルトのイントラモードを用いて変換を行うことをさらに含む。

【０３３６】

図１８Ｆは、映像処理のための例示的な方法１８６０を示す。方法１８６０は、ステップ１８６２において、最大確率モード（ＭＰＭ）リストを構築する間、サンプル値を量子化するパルス符号化変調（ＰＣＭ）モードで符号化された近傍のブロックに起因して、映像の現在の映像ブロックを直流（ＤＣ）予測モードで符号化することを決定することを含む。方法１８６０は、ステップ１８６４において、映像の現在の映像ブロックと符号化表現との間で変換を行うことをさらに含む。

【０３３７】

図１８Ｇは、映像処理のための例示的な方法１８７０を示す。方法１８７０は、ステップ１８７２において、現在の映像ブロックの現在の映像ブロックとこの映像の符号化表現との変換のために、この現在の映像ブロックの１つ以上の近傍のブロックの符号化された情報に基づいて、三角形分割モード（ＴＰＭ）および／または動き記憶プロセスを使用するプロセスの適用可能性を決定することを含む。方法１８７０は、ステップ１８７４において、この決定に基づいてこの変換を実行することを含み、ＴＰＭを用いることは、映像ブロックを、少なくとも１つの非長方形パーティションを含むサブパーティションに分割することを含む。

【０３３８】

開示される技術のいくつかの実施形態は、映像処理ツールまたはモードを有効化するように決定または判定することを含む。一例において、映像処理ツールまたはモードが有効化される場合、エンコーダは、１つの映像ブロックを処理する際にこのツールまたはモードを使用するまたは実装するが、このツールまたはモードの使用に基づいて、結果として得られるビットストリームを必ずしも修正しなくてもよい。すなわち、映像のブロックから映像のビットストリーム表現への変換は、決定または判定に基づいて映像処理ツールまたはモードが有効化される場合に、この映像処理ツールまたはモードを使用する。別の例において、映像処理ツールまたはモードが有効化される場合、デコーダは、ビットストリームが映像処理ツールまたはモードに基づいて修正されたことを知って、ビットストリームを処理する。すなわち、決定または判定に基づいて有効化された映像処理ツールまたはモードを使用して、映像のビットストリーム表現から映像のブロックへの変換を行う。

【０３３９】

開示される技術のいくつかの実施形態は、映像処理ツールまたはモードを無効化するように決定または判定することを含む。一例において、映像処理ツールまたはモードが無効にされている場合、エンコーダは、映像のブロックを映像のビットストリーム表現に変換する際に、このツールまたはモードを使用しない。別の例において、映像処理ツールまたはモードが無効にされている場合、デコーダは、決定または判定に基づいて無効化された映像処理ツールまたはモードを使用してビットストリームが修正されていないことを知って、ビットストリームを処理する。

【０３４０】

本明細書では、「映像処理」という用語は、映像符号化、映像復号化、映像圧縮、または映像展開を指すことができる。例えば、映像圧縮アルゴリズムは、映像の画素表現から対応するビットストリーム表現への変換、またはその逆の変換中に適用されてもよい。現在の映像ブロックのビットストリーム表現は、例えば、構文によって規定されるように、ビットストリーム内の同じ場所または異なる場所に拡散されるビットに対応していてもよ

い。例えば、１つのマクロブロックは、変換および符号化された誤り残差値の観点から、且つビットストリームにおけるヘッダおよび他のフィールドにおけるビットを使用して符号化されてもよい。

【０３４１】

以下の項目は、いくつかの実施形態及び技術を説明する。第１組の項目では、前章で開示された技術の特定の特徴及び態様を説明する。

【０３４２】

１．映像符号化方法であって、映像の映像領域と映像領域のビットストリーム表現との変換中に、映像領域に関するパレット符号化モード条件に基づいて、映像領域の非ブロック化処理のパラメータを判定することと、選択されたパラメータを使用して、変換中に非ブロック化処理を使用することと、を含む、方法。

【０３４３】

２．非ブロック化処理のパラメータが、イントラ符号化モードまたはインター符号化モードまたはイントラブロックコピー符号化モードで符号化された別の映像領域の変換に使用されるパラメータとは異なる、項目１に記載の方法。

【０３４４】

３．パラメータが、Ｐ側およびＱ側の映像領域の近傍の映像領域がパレットモード符号化を使用して符号化されるかどうかに基づいて決定される境界強度値を含む、項目１～２のいずれかに記載の方法。

【０３４５】

４．パレットモードで符号化されているＰ側およびＱ側の両方の映像領域のために、境界強度がゼロである、項目３の方法。

【０３４６】

５．パレットモードが、イントラ予測モードと組み合わせたパレットモード（ＰＣＩＰ）または複合パレットモード（ＣＰＭ）と組み合わせたパレットモードに対応する、項目１～４のいずれかに記載の方法。

【０３４７】

セクション４の項目１は、上記の方法の追加の例および実施形態を提供する。

【０３４８】

６．映像処理方法であって、映像ピクチャと映像ピクチャのビットストリーム表現との変換中に、割り当て規則に従って、近傍の映像の符号化モードに基づいて映像ピクチャの映像ブロックに非ブロック化処理を適用するための非ブロック化フィルタパラメータを選択することと、割り当て規則に従って、非ブロック化処理を適用することにより、変換を行うことと、を含み、割り当て規則は、近傍のブロックが複合パレットモードで符号化されている場合、または別の符号化モードで符号化されている場合に、同じパラメータを使用するように指定する、方法。

【０３４９】

７．別の符号化モードがイントラブロックコピーモードに対応する、項目６に記載の方法。

【０３５０】

８．別の符号化モードがパレット符号化モードに対応する、項目６に記載の方法。

【０３５１】

セクション４の項目２は、上記の方法の追加の例および実施形態を提供する。

【０３５２】

９．映像処理方法であって、映像ピクチャと映像ピクチャのビットストリーム表現との変換中に、割り当て規則に従って、近傍の映像の符号化モードに基づいて映像ピクチャの映像ブロックに非ブロック化処理を適用するための非ブロック化フィルタパラメータを選択することと、割り当て規則に従って、非ブロック化処理を適用することにより、変換を行うことと、を含み、割り当て規則は、近傍のブロックがイントラ予測モードと組み合わせたパレット（ＰＣＩＰ）モードで符号化されている場合または別の符号化モードで符号

化されている場合に同じパラメータを使用するように指定する、方法。

【０３５３】

１０．別の符号化モードがイントラブロックコピーモードに対応する、項目９に記載の方法。

【０３５４】

１１．別の符号化モードがパレット符号化モードに対応する、項目９に記載の方法。

【０３５５】

セクション４の項目３は、上記の方法の追加の例および実施形態を提供する。

【０３５６】

１２．映像処理方法であって、コンポーネント値のパレットを対応する映像ブロックの画素を符号化するために使用するパレット符号化モードを使用して映像ピクチャの映像ブロックと映像ブロックのビットストリーム表現との変換を行うことを含み、ビットストリーム表現中のフラグは、対応する映像ブロックにおける非ゼロ係数が存在していることを指示し、このフラグは、パレット符号化モード、イントラ予測モードと組み合わせたパレットモード（ＰＣＩＰ）、または複合パレットモード（ＣＰＭ）、で符号化された映像ブロックを信号通知するために使用される、方法。

【０３５７】

１３．フラグの値は、対応するブロックがエスケープ画素を有するかどうかに依存する、項目１２に記載の方法。

【０３５８】

セクション４の項目４は、上記の方法の追加の例および実施形態を提供する。

【０３５９】

１４．映像処理方法であって、複数の映像ブロックを含む映像ピクチャと映像ピクチャのビットストリーム表現との変換中に、複数の映像ブロックの各々に関連したパレット復号化モード条件に基づいて複数の映像ブロックの各々に対する非ブロック化処理の適用可能性を判定することと、その適用可能性に基づいて、変換中に複数のブロックに非ブロック化処理を選択的に使用することと、を含む、方法。

【０３６０】

１５．適用可能性の判定は、非ブロック化処理が、パレットモードまたはイントラ予測モードと組み合わせたパレットモード（ＰＣＩＰ）、または複合パレットモード（ＣＰＭ）で符号化されたブロックには適用できないことを判定する、項目１４に記載の方法。

【０３６１】

１６．適用可能性の判定は、Ｐ側ブロックまたはＱ側ブロックがそれぞれパレットモードまたはＰＣＩＰモードまたはＣＰＭモードで符号化されている場合に、Ｐ側サンプルまたはＱ側サンプルが適用不可能であると判定することを含む、項目１４に記載の方法。

【０３６２】

セクション４の項目５は、上記の方法の追加の例および実施形態を提供する。

【０３６３】

１７．映像処理方法であって、複数の映像ブロックを含む映像フレームと、複合パレットモード（ＣＰＭ）で符号化された映像フレームのビットストリーム表現との変換を行うことを含み、ビットストリーム表現は、映像領域に適用可能なイントラモード、インターモード、イントラブロックコピーモード、またはパレットモードの指示とは別の、映像領域レベルでのＣＰＭモードの指示を含む、方法。

【０３６４】

１８．映像領域が、変換ユニットまたは予測ユニットまたは符号化ブロックユニットに対応する、項目１７に記載の方法。

【０３６５】

１９．ビットストリーム表現が、映像領域に適用可能なイントラモード、インターモード、イントラブロックコピーモードまたはパレットモードの指示の後に、ＣＰＭモードの指示を含む、項目１７～１８のいずれかに記載の方法。

【０３６６】
セクション４の項目６は、上記の方法の追加の例および実施形態を提供する。
【０３６７】
２０．映像処理方法であって、コンポーネント値のパレットを対応する映像ブロックの画素を符号化するために使用する符号化モードを使用して、映像ピクチャの映像ブロックと映像ブロックのビットストリーム表現との変換を行うことを含み、パレット符号化モードに関する構文要素が、所与の映像ブロックが属する色成分の特性に基づいて、所与の映像ブロックのビットストリーム表現に選択的に含まれ、符号化モードは、パレット符号化モード、イントラ予測と組み合わせた（ＰＣＩＰ）モードまたは複合パレットモード（ＣＰＭ）を含む、方法。
【０３６８】
２１．映像処理方法であって、パレット符号化モードに関する構文要素が、所与の映像ブロックに使用されるパーティション構造または平面符号化に基づいてさらに選択的に含まれる、映像処理方法。
【０３６９】
２２．構文要素が二重ツリーパーティション構造に対してのみ含まれる、項目２０～２１のいずれかに記載の方法
【０３７０】
２３．映像ピクチャが４：０：０フォーマットであり、所与の映像ブロックがクロマブロックであることに起因して、構文要素がスキップされる、項目２０～２１のいずれかに記載の方法。
【０３７１】
セクション４の項目７は、上記の方法の追加の例および実施形態を提供する。
【０３７２】
２４．所与の映像ブロックはクロマ成分に対応し、パレット符号化モードに関する構文要素は、所与の映像に対応する輝度ブロックの１つ以上の選択されたブロックの符号化された情報に基づいて選択的に含まれる、項目２０に記載の方法。
【０３７３】
２５．輝度ブロックの１つ以上の選択されたブロックが、最小の符号化ユニットまたは最小の予測ユニットまたは最小の変換ユニットサイズに対応する、項目２４に記載の方法。
【０３７４】
セクション４の項目８は、上記の方法の追加の例および実施形態を提供する。
【０３７５】
２６．符号化モードがＰＣＩＰモードである、項目２０～２５のいずれかに記載の方法。
【０３７６】
セクション４の項目９は、上記の方法の追加の例および実施形態を提供する。
【０３７７】
２７．複数の映像ブロックを含む映像ピクチャと、複数のブロックの少なくとも一部が予測符号化モードで符号化されている映像ピクチャのビットストリーム表現との変換を行い、映像ブロックは、映像ピクチャの他の映像ブロックを符号化するために使用される他の予測符号化モードとは別のまたはその特別な事例として識別される複合パレットモード（ＣＰＭ）で符号化される、映像処理方法。
【０３７８】
２８．他の予測符号化モードは、イントラモード、イントラブロックコピー、パレットモード、イントラスライスまたはＩピクチャまたはイントラタイルグループのＣＰＭモードを含む、項目２７に記載の方法。
【０３７９】
セクション４の項目１０は、上記の方法の追加の例および実施形態を提供する。

【０３８０】
２９．ＣＰＭはビットストリーム表現における他の予測符号化モードの特殊なケースとして識別され、ＣＰＭを信号通知する構文要素は、ＣＰＭが特殊なケースである他の予測モードのタイプに基づいて選択的に信号通知される、項目２７の方法。
【０３８１】
３０Ａ．他の予測モードのタイプがイントラモードであり、ＣＰＭの構文要素の信号通知がスキップされる、項目２９に記載の方法。
【０３８２】
３０Ｂ．他の予測モードのタイプがスキップモードであり、ＣＰＭの構文要素の信号通知がスキップされる、項目２９に記載の方法。
【０３８３】
セクション４の項目１１は、上記の方法の追加の例および実施形態を提供する。
【０３８４】
３１．映像処理方法であって、現在の映像ブロックと、現在の映像ブロックが複合パレットモード（ＣＰＭ）を用いて符号化された現在の映像ブロックの第１のビットストリーム表現と、の第１の変換を行い、現在の映像ブロックは現在の映像ブロックに関連付けられた動きベクトルを有し、そして、次の映像ブロックと、次の映像ブロックの第２のビットストリーム表現と、の第２の変換を行い、現在の映像ブロックに関連付けられた動きベクトルを用いた動き予測が第２の変換中に使用される、方法。
【０３８５】
３２．動き予測が第２の変換のマージ候補として使用される、項目３１に記載の方法。
【０３８６】
３３．動き予測が、第２の変換中に履歴に基づく動きベクトル予測器として使用される、項目３１または３２に記載の方法。
【０３８７】
セクション４の項目１２は、上記の方法の追加の例および実施形態を提供する。
【０３８８】
３４．映像処理方法であって、現在の映像ブロックと、現在の映像ブロックがイントラ予測と組み合わせた（ＰＣＩＰ）モードで符号化されている現在の映像ブロックの第１のビットストリーム表現と、の第１の変換を行い、現在の映像ブロックは、現在の映像ブロックに関連付けられたイントラ予測方向を有し、そして、次の映像ブロックと次の映像ブロックの第２のビットストリーム表現と、の第２の変換を行い、現在の映像ブロックに関連付けられたイントラ予測方向に基づくイントラモード予測子が第２の変換中に使用される、方法。
【０３８９】
３５．現在の映像ブロックに関連付けられたイントラ予測方向が、第２の変換中に最大確率モード候補として使用される、項目３４に記載の方法。
【０３９０】
セクション４の項目１３は、上記の方法の追加の例および実施形態を提供する。
【０３９１】
３６．映像処理方法であって、映像の現在の映像ブロックと映像のビットストリーム表現との変換中に使用されるパレットの最大エントリ数を判定することと、最大数を使用して変換を行うことと、を含み、ビットストリーム表現内のフィールドが、映像ユニット全体で最大数の変化を信号通知する、方法。
【０３９２】
３７．映像ユニットが、符号化ツリーブロックまたは符号化ユニットまたは予測ユニットまたは変換ユニットに対応する、項目３６に記載の方法。
【０３９３】
３８．最大数が現在の映像ブロックの符号化された情報に基づく、項目３６～３７の方法。

【０３９４】
セクション４の項目１４は、上記の方法の追加の例および実施形態を提供する。
【０３９５】
３９．映像処理方法であって、１つ以上の映像ブロックを含む映像ピクチャと映像ピクチャのビットストリーム表現との変換中に、パレット符号化モードを使用することに起因して、スケーリング行列の使用が無効であると判定することと、スケーリング行列の使用が無効であると判定したことに基づいて変換を行うことと、を含む、方法。
【０３９６】
４０．映像処理方法であって、１つ以上の映像ブロックを含む映像ピクチャと映像ピクチャのビットストリーム表現との変換中に、パレット符号化モードを使用することに起因して、スケーリング行列の使用が許可されていることを判定することと、スケーリング行列の使用が許可されているという判定に基づいて変換を行うことと、を含む、方法。
【０３９７】
セクション４の項目１５は、上記の方法の追加の例および実施形態を提供する。
【０３９８】
４１．変換が行われている映像の特性を判定する際にこの方法が適用される、項目１から４０のいずれかに記載の方法。
【０３９９】
４２．映像の特性が映像のコンテンツタイプを含む、項目４１に記載の方法。
【０４００】
４３．映像の特性が、変換に使用されるブロック寸法を含む、項目４１に記載の方法。
【０４０１】
４４．映像の特性が、変換に使用されるパーティションツリー構造を含む、項目４１に記載の方法。
【０４０２】
セクション４の項目２５は、上記の方法の追加の例および実施形態を提供する。
【０４０３】
４６．映像処理方法であって、現在の映像ブロックの１つ以上の特徴に基づいて、この現在の映像ブロックにおけるイントラブロックコピー（ＩＢＣ）モードの使用を符号化するコンテクストを構成することと、この構成に基づいて、この現在の映像ブロックとこの現在の映像ブロックのビットストリーム表現とを変換することと、を含む方法。
【０４０４】
４７．前記１つ以上の特徴は、前記現在の映像ブロックのブロックサイズを含む、項目４６に記載の方法。
【０４０５】
４８．前記１つ以上の特徴は、前記現在の映像ブロックの形状を含む、項目４６に記載の方法。
【０４０６】
４９．前記形状は、正方形または非正方形である、項目４８に記載の方法。
【０４０７】
セクション４の項目１６は、上記の方法の追加の例および実施形態を提供する。
【０４０８】
５０．前記構成は、現在の映像ブロックの近傍のブロックの符号化モードにさらに基づく、項目４６に記載の方法。
【０４０９】
セクション４の項目１７は、上記の方法の追加の例および実施形態を提供する。
【０４１０】
５１．前記コンテクストは、第１の構文要素を含み、前記構成は、現在の映像ブロックの近傍のブロックに関連付けられた第２の構文要素にさらに基づく、項目４６に記載の方法。

【０４１１】
セクション４の項目１８は、上記の方法の追加の例および実施形態を提供する。
【０４１２】
５２．映像処理方法であって、現在の映像ブロックのビットストリーム表現における符号化モードの指示を選択的に含めることに関して、現在の映像ブロックに対して、恒等変換を使用する符号化モードを決定することと、この決定に基づいて、現在の映像ブロックとビットストリーム表現との変換を行うことと、を含む方法。
【０４１３】
５３．前記符号化モードは、変換スキップモードまたはブロック差動パルス符号変調（ＱＲ－ＢＤＰＣＭ）である、項目５２に記載の方法。
【０４１４】
セクション４の項目２３は、上記の方法の追加の例および実施形態を提供する。
【０４１５】
５４．変換が、ビットストリーム表現から映像領域の画素値を生成することを含む、項目１～５３のいずれかに記載の方法。
【０４１６】
５５．変換が、映像領域の画素値からビットストリーム表現を生成することを含む、項目１～５３のいずれかに記載の方法。
【０４１７】
５６．項目１～５５のいずれか１項以上に記載の方法を実装するように構成された処理装置を備える映像処理装置。
【０４１８】
５７．実行されると、項目１～５５のいずれか１つ以上に記載の方法を処理装置に実装させるプログラムコードを記憶するコンピュータ可読媒体。
【０４１９】
第２組の項目は、前のセクションで開示された手法の特定の機能と側面、たとえば、実装例１６から２５について説明している。
【０４２０】
１．映像処理方法であって、クロマブロックであり、クロマＤＭモードで符号化された映像の現在の映像ブロックに対して、前記現在の映像ブロックのために、前記現在の映像ブロックの符号化モードに関する規則に従って、イントラ予測モードを導出することと、現在の映像ブロックと前記映像の符号化表現との変換を行うことと、を含み、前記規則は、クロマＤＭモードの導出に使用され、現在のビデオブロックに対応する量子化残差ブロックのブロック差分パルス符号変調表現を用いて符号化された、対応する輝度ブロックのイントラ予測方向に基づいて、イントラ予測モードを導出することを規定する、方法。
【０４２１】
２．前記クロマＤＭモードで使用されるイントラ予測モードは、対応する輝度ブロックのイントラ予測方向が垂直である場合、垂直モードである、項目１に記載の方法。
【０４２２】
３．前記クロマＤＭモードで使用されるイントラ予測モードは、対応する輝度ブロックのイントラ予測方向が水平である場合、水平モードである、項目１に記載の方法。
【０４２３】
４．前記規則は、複合ＩＢＣおよび／またはパレットモードの使用に基づいて前記イントラ予測モードを導出することを指定する、項目１に記載の方法。
【０４２４】
５．前記イントラ予測モードは、平面予測モード、水平予測モード、または垂直予測モードである、項目４に記載の方法。
【０４２５】
６．映像処理方法であって、クロマブロックである映像の現在の映像ブロックと前記映像の符号化表現との変換を行うことを含み、現在の映像ブロックの対応する輝度映像ブロ

ックが非イントラモードで符号化されるため、現在の映像ブロックのための予測ブロックを生成するためのイントラ予測方法が使用される、方法。

【０４２６】

７．前記イントラ予測方法は、対応する輝度ブロックをパレットモードおよび／またはパレットモードの変形で符号化する場合、垂直イントラ予測に対応し、パレットモードは、代表的なサンプル値のパレットを使用して現在の映像ブロックを表現または再構成することを可能にする、項目６に記載の方法。

【０４２７】

８．前記予測ブロックは、対応する輝度ブロックをパレットモードおよび／またはパレットモードの変形で符号化する場合、水平予測で埋められ、パレットモードは、代表的なサンプル値のパレットを使用して現在の映像ブロックを表現または再構成することを可能にする、項目６に記載の方法。

【０４２８】

９．前記予測ブロックは、対応する輝度ブロックをパレットモードおよび／またはパレットモードの変形で符号化する場合、平面予測で埋められ、パレットモードは、代表的なサンプル値のパレットを使用して現在の映像ブロックを表現または再構成することを可能にする、項目６に記載の方法。

【０４２９】

１０．前記予測ブロックは、対応する輝度ブロックをパレットモードおよび／またはパレットモードの変形で符号化する場合、ＤＣ予測で埋められ、パレットモードは、代表的なサンプル値のパレットを使用して現在の映像ブロックを表現または再構成することを可能にする、項目６に記載の方法。

【０４３０】

１１．前記予測ブロックは、対応する輝度ブロックをパレットモードおよび／またはパレットモードの変形で符号化する場合、イントラ予測モードで埋められ、パレットモードは、代表的なサンプル値のパレットを使用して現在の映像ブロックを表現または再構成することを可能にする、項目６に記載の方法。

【０４３１】

１２．前記予測ブロックは、近傍の輝度ブロックのイントラ予測モードおよび／または近傍のクロマブロックのイントラ予測モードに基づいて生成される、項目６に記載の方法。

【０４３２】

１３．対応する輝度ブロックは、ＩＢＣモードおよび／またはＩＢＣモードの変形で符号化され、ＩＢＣモードは、別の映像ブロックの予め符号化されたサンプルを使用して、現在の映像ブロックのサンプル値を予測することを可能にする、項目６に記載の方法。

【０４３３】

１４．前記対応する輝度ブロックは、パルス符号変調（ＰＣＭ）モードおよび／または前記ＰＣＭモードの変形によって符号化され、前記ＰＣＭモードは、サンプリングされたアナログ信号をデジタル的に表すことを可能にする、項目６に記載の方法。

【０４３４】

１５．前記対応する輝度ブロックは、ＱＲ－ＢＤＰＣＭモードで符号化され、残差ブロックにおける残差サンプルが量子化され、量子化残差サンプルに対して差動パルス符号変調が適用される、項目６に記載の方法。

【０４３５】

１６．前記イントラ予測方法は、最大確率モード（ＭＰＭ）リストを構築する間に、ＰＣＭブロックに割り当てられるイントラ予測モードに対応する、項目６に記載の方法。

【０４３６】

１７．前記イントラ予測方法は、輝度ＰＣＭブロックに割り当てられるイントラ予測モードに対応する、項目６に記載の方法。

【０４３７】

１８．前記イントラ予測方法は、クロマＰＣＭブロックに割り当てられるイントラ予測モードに対応する、項目６に記載の方法。

【０４３８】

１９．前記イントラ予測方法は、対応する輝度ブロックのＭＰＭリストにおける第１のイントラ予測モードに対応する、項目６に記載の方法。

【０４３９】

２０．映像処理方法であって、映像のピクチャの現在の映像ブロックと前記映像の符号化表現とを変換するために、現在の映像ブロックの特性に基づいて、前記現在の映像ブロックに対してイントラブロックコピー（ＩＢＣ）モードを使用することを示す構文要素のための１つ以上のコンテクストを構成することと、この構成に基づいてこの変換を行うことと、を含み、前記ＩＢＣモードを使用することは、前記ピクチャにおける別の映像ブロックの予め符号化されたサンプルを使用して前記現在の映像ブロックのサンプル値を予測することを含む、方法。

【０４４０】

２１．前記特徴は、前記現在の映像ブロックのブロックサイズを含む、項目２０に記載の方法。

【０４４１】

２２．前記構文要素は、異なるブロックサイズを有する映像ブロックに対して異なるコンテクストを使用することで信号通知される、項目２０に記載の方法。

【０４４２】

２３．前記構文要素は、ブロックサイズが所定の値より小さいまたは大きくない映像ブロックのために単一のコンテクストを使用することによって信号通知される、項目２０に記載の方法。

【０４４３】

２４．前記構文要素は、ブロックサイズが所定の値より大きいまたは小さくない映像ブロックのために単一のコンテクストを使用することによって信号通知される、項目２０に記載の方法。

【０４４４】

２５．前記所定の値は、ｉ）前記映像のコンテンツタイプ、ｉｉ）パラメータセット、ヘッダ、符号化ユニット、変換ユニット、または予測ユニットにおいて信号通知されるメッセージ、ｉｉｉ）映像のカラーフォーマットの指示、ｉｖ）符号化ツリー構造、ｖ）スライス、タイルグループ、またはピクチャのタイプ、ｖｉ）色成分、ｖｉｉ）時間層識別（ＩＤ）、ｖｉｉｉ）標準に関する情報、またはｘｉ）デコーダに信号通知される情報、のうちの少なくとも１つに基づいて決定される、項目２３または２４に記載の方法。

【０４４５】

２６．前記特徴は、前記現在の映像ブロックのブロック形状を含む、項目２０に記載の方法。

【０４４６】

２７．前記構文要素は、異なるブロックサイズを有する映像ブロックに対して異なるコンテクストを使用することで信号通知される、項目２６に記載の方法。

【０４４７】

２８．現在の映像ブロックを含むスライスのタイプがＩスライスであるかどうかに基づいて、１つ以上のコンテクストが異なる、項目２０に記載の方法。

【０４４８】

２９．前記１つ以上のコンテクストを構成するステップは、現在の映像ブロックの近傍のブロックに関連付けられた別の構文要素にさらに基づく、項目２８に記載の方法。

【０４４９】

３０．１つのコンテクストのみが前記構文要素を符号化するように構成される、項目２０に記載の方法。

【０４５０】

３１．現在の映像ブロックが特定のタイプのスライスまたはピクチャに含まれる、項目２０に記載の方法。

【０４５１】

３２．前記構文要素は、現在の映像ブロックの１つ以上の近傍のブロックがＩＢＣモードで符号化されているかどうかに依存する、項目２０に記載の方法。

【０４５２】

３３．前記１つ以上のコンテクストは、前記近傍のブロックのうちの少なくとも１つが前記ＩＢＣモードで符号化される場合に構成される、項目３２に記載の方法。

【０４５３】

３４．映像の現在の映像ブロックと前記映像の符号化表現との変換のために、前記現在の映像ブロックの近傍のブロックに関連付けられた第２の構文要素に基づいて、第１の構文要素に関連付けられたビンを符号化するためのコンテクストモデルを構成することと、前記構成に基づいて前記変換を行うこととを含む、映像処理方法。

【０４５４】

３５．第１の構文要素は、インター予測とイントラ予測とを組み合わせたイントラインター予測モード、または映像ブロックをサブパーティションに分割し、かつ少なくとも１つのサブパーティションが非長方形パーティションである三角形予測モード（ＴＰＭ）、の使用を示すためのものである、項目３４に記載の方法。

【０４５５】

３６．前記第２の構文要素は、近傍のブロックのＩＢＣモードの使用に関する、項目３４に記載の方法。

【０４５６】

３７．映像処理方法であって、映像の現在の映像ブロックとこの映像の符号化表現との変換を行うことを含み、ここで、現在の映像ブロックに対応する量子化残差ブロックのブロック差動パルス符号変調表現を使用して、この符号化表現が、この符号化表現において表されるこの現在の映像ブロックのための非ゼロ係数を含むかどうかを示すために、１つのフラグを決定する、方法。

【０４５７】

３８．前記フラグは、現在の映像ブロックに対して１に推測される、項目３７に記載の方法。

【０４５８】

３９．前記フラグは、輝度成分および／またはクロマ成分のためのものである、項目３７または項目３８に記載の方法。

【０４５９】

４０．映像処理方法であって、クロマブロックである映像の現在の映像ブロックとこの映像の符号化表現との変換を行うことを含み、現在の映像ブロックのためのクロマＤＭモードは、クロマＤＭモードを導出するために使用され、イントラモードで符号化されない対応する輝度ブロックに起因して無効化される、方法。

【０４６０】

４１．対応する輝度ブロックがイントラモードで符号化されていない場合、クロマＤＭモードが適用されない、項目４０に記載の方法。

【０４６１】

４２．前記クロマＤＭモードは、対応する輝度ブロックがＩＢＣモードおよび／またはＩＢＣモードの変形で符号化される場合には、適用されず、ＩＢＣモードは、別の映像ブロックの予め符号化されたサンプルを使用して、現在の映像ブロックのサンプル値を予測することを可能にする、項目４０に記載の方法。

【０４６２】

４３．前記クロマＤＭモードは、対応する輝度ブロックがパレットモードおよび／またはパレットモードの変形で符号化される場合には、適用されず、パレットモードは、代表的なサンプル値のパレットを使用して現在の映像ブロックを表現または再構成することを

可能にする、項目４０に記載の方法。
【０４６３】
４４．前記クロマＤＭモードは、対応する輝度ブロックがパルス符号変調（ＰＣＭ）モードおよび／または前記ＰＣＭモードの変形によって符号化される場合には、適用されず、ＰＣＭモードは、サンプリングされたアナログ信号をデジタル的に表すことを可能にする、項目４０に記載の方法。
【０４６４】
４５．現在の映像ブロックに対してクロマＤＭモードが許可されていない場合、クロマＤＭモードの指示が映像の符号化表現に含まれていない、項目４０～４４のいずれかに記載の方法。
【０４６５】
４６．適合ビットストリームが、対応する構文要素の復号化された情報が特定の値でないことを満たす場合、クロマＤＭモードの指示が、映像の符号化表現に含まれる、項目４０～４４のいずれかに記載の方法。
【０４６６】
４７．映像処理方法であって、規則に従って、クロマブロックである映像の現在の映像ブロックとこの映像の符号化表現とを変換することを含み、現在の映像ブロックは、対応する輝度ブロックの符号化情報から予測ブロックのためのモードを導出する導出モードを使用して計算された予測ブロックを使用して、符号化表現で表現され、この規則は、導出モードを使用することによって、現在の映像ブロックが無効なブロックベクトルを有すると仮定して、予測ブロックを決定することを指定する、方法。
【０４６７】
４８．前記予測ブロックは、デフォルト値で埋められる、項目４７に記載の方法。
【０４６８】
４９．前記デフォルト値は、入力および／または内部ビット深度に依存する、項目４８に記載の方法。
【０４６９】
５０．対応する輝度ブロックの予測モードは、非イントラモードである、項目４７に記載の方法。
【０４７０】
５１．対応する輝度ブロックの予測モードは、ＩＢＣモードおよび／またはＩＢＣモードの変形であり、ＩＢＣモードは、別の映像ブロックの予め符号化されたサンプルを使用して、現在の映像ブロックのサンプル値を予測することを可能にする、項目５０に記載の方法。
【０４７１】
５２．前記対応する輝度ブロックの前記予測モードは、パレットモードおよび／または前記パレットモードの変形であり、パレットモードは、代表的なサンプル値のパレットを用いて現在の映像ブロックを表現または再構成することを可能にする、項目５０に記載の方法。
【０４７２】
５３．映像の現在の映像ブロックとこの映像の符号化表現との変換のために、規則に従って現在の映像ブロックのデフォルトのイントラモードを決定することと、前記デフォルトのイントラモードを使用して前記変換を実行することと、を含む、映像処理方法。
【０４７３】
５４．追加情報がない場合、現在の映像ブロックを符号化表現に表現するためにデフォルトモードが使用されると想定される、項目５３に記載の方法。
【０４７４】
５５．前記デフォルトのイントラモードは、直流（ＤＣ）モード、平面モード、垂直モード、または水平モードである、項目５３に記載の方法。
【０４７５】

５６．前記デフォルトのイントラモードを、イントラモード符号化プロセスにおける最も確率の高いモード（ＭＰＭ）として用いる、項目５３に記載の方法。

【０４７６】

５７．ＤＭモードで符号化されたクロマブロックである現在の映像ブロックに対して、対応する輝度ブロックのデフォルトモードを用いて、現在の映像ブロックのＤＭモードを導出する、項目５３に記載の方法。

【０４７７】

５８．前記現在の映像ブロックは、非イントラモードで符号化される、項目５３に記載の方法。

【０４７８】

５９．前記現在の映像ブロックは、ＩＢＣモード、インターモード、ＱＲ－ＤＰＣＭモード、ＰＣＭモードまたはパレットモードで符号化される、項目５３に記載の方法。

【０４７９】

６０．映像処理方法であって、最大確率モード（ＭＰＭ）リストを構築する間、サンプル値を量子化するパルス符号化変調（ＰＣＭ）モードで符号化された近傍のブロックに起因して、映像の現在の映像ブロックを直流（ＤＣ）予測モードで符号化することを決定することと、現在の映像ブロックと映像の符号化表現との変換を行うことと、を含む、方法。

【０４８０】

６１．映像処理方法であって、現在の映像ブロックの現在の映像ブロックとこの映像の符号化表現との変換のために、この現在の映像ブロックの１つ以上の近傍のブロックの符号化された情報に基づいて、三角形分割モード（ＴＰＭ）および／または動き記憶プロセスを使用するプロセスの適用可能性を決定することと、前記決定に基づいて、前記変換を行うことと、を含み、前記ＴＰＭを使用することは、映像ブロックを、少なくとも１つの非長方形パーティションを含むサブパーティションに分割することを含む、方法。

【０４８１】

６２．前記決定することは、前記近傍のブロックがＩＢＣモード、パレットモード、ＴＳモード、量子化残差ブロック差動パルス符号変調（ＱＲ－ＢＤＰＣＭ）モードのうち少なくとも１つのモードで符号化される場合、ＴＰＭを使用して前記プロセスを適用しないことを決定する、項目６１に記載の方法。

【０４８２】

６３．前記決定することは、ＩＢＣモード、パレットモード、ＴＳモード、量子化残差ブロック差動パルス符号変調（ＱＲ－ＢＤＰＣＭ）モード、のうちの少なくとも１つを用いて符号化された近傍のブロックに起因して、修正ＴＰＭ関連処理および／または動き記憶処理を適用することを決定する、項目６１に記載の方法。

【０４８３】

６４．前記変換を行うことは、前記映像から前記符号化表現を生成するステップを含む、項目１～６３のいずれかに記載の方法。

【０４８４】

６５．前記変換を行うことは、前記符号化表現から前記映像を生成することを含む、項目１～６３のいずれかに記載の方法。

【０４８５】

６６．項目１～６５のいずれか１つ以上に記載の方法を実装するように構成された処理装置を備える映像処理装置。

【０４８６】

６７．実行されると、項目１～６５のいずれか１つ以上に記載の方法を処理装置に実装させるプログラムコードを記憶するコンピュータ可読媒体。

【０４８７】

本明細書に記載された開示された、およびその他の解決策、実施例、実施形態、モジュール、および機能動作の実装形態は、本明細書に開示された構造およびその構造的等価物

を含め、デジタル電子回路、またはコンピュータソフトウェア、ファームウェア、若しくはハードウェアで実施されてもよく、またはそれらの１つ以上の組み合わせで実施してもよい。開示された、およびその他の実施形態は、１つ以上のコンピュータプログラム製品、すなわち、データ処理装置によって実装されるため、またはデータ処理装置の動作を制御するために、コンピュータ可読媒体上に符号化されたコンピュータプログラム命令の１つ以上のモジュールとして実施することができる。このコンピュータ可読媒体は、機械可読記憶デバイス、機械可読記憶基板、メモリデバイス、機械可読伝播信号をもたらす物質の組成物、またはこれらの１つ以上の組み合わせであってもよい。「データ処理装置」という用語は、例えば、プログラマブル処理装置、コンピュータ、または複数の処理装置、若しくはコンピュータを含む、データを処理するためのすべての装置、デバイス、および機械を含む。この装置は、ハードウェアの他に、当該コンピュータプログラムの実行環境を作るコード、例えば、処理装置ファームウェア、プロトコルスタック、データベース管理システム、オペレーティングシステム、またはこれらの１つ以上の組み合わせを構成するコードを含むことができる。伝播信号は、人工的に生成した信号、例えば、機械で生成した電気、光、または電磁信号であり、適切な受信装置に送信するための情報を符号化するために生成される。

【０４８８】

コンピュータプログラム（プログラム、ソフトウェア、ソフトウェアアプリケーション、スクリプト、またはコードとも呼ばれる）は、コンパイルされた言語または解釈された言語を含む任意の形式のプログラミング言語で記述することができ、また、それは、スタンドアロンプログラムとして、またはコンピューティング環境で使用するのに適したモジュール、成分、サブルーチン、または他のユニットとして含む任意の形式で展開することができる。コンピュータプログラムは、必ずしもファイルシステムにおけるファイルに対応するとは限らない。プログラムは、他のプログラムまたはデータを保持するファイルの一部（例えば、マークアップ言語文書に格納された１つ以上のスクリプト）に記録されていてもよいし、当該プログラム専用の単一のファイルに記憶されていてもよいし、複数の調整ファイル（例えば、１つ以上のモジュール、サブプログラム、またはコードの一部を格納するファイル）に記憶されていてもよい。１つのコンピュータプログラムを、１つのサイトに位置する１つのコンピュータ、または複数のサイトに分散され通信ネットワークによって相互接続される複数のコンピュータで実行させるように展開することも可能である。

【０４８９】

本明細書に記載された処理およびロジックフローは、入力データ上で動作し、出力を生成することによって機能を実行するための１つ以上のコンピュータプログラムを実行する１つ以上のプログラマブル処理装置によって行うことができる。処理およびロジックフローはまた、特定用途のロジック回路、例えば、ＦＰＧＡ（フィールドプログラマブルゲートアレイ）またはＡＳＩＣ（特定用途向け集積回路）によって行うことができ、装置はまた、特別目的のロジック回路として実装することができる。

【０４９０】

コンピュータプログラムの実行に適した処理装置は、例えば、汎用および専用マイクロ処理装置の両方、並びに任意の種類のデジタルコンピュータの任意の１つ以上の処理装置を含む。一般的に、処理装置は、リードオンリーメモリまたはランダムアクセスメモリまたはその両方から命令およびデータを受信する。コンピュータの本質的な要素は、命令を実行するための処理装置と、命令およびデータを記憶するための１つ以上のメモリデバイスとである。一般的に、コンピュータは、データを記憶するための１つ以上の大容量記憶デバイス、例えば、磁気、光磁気ディスク、または光ディスクを含んでもよく、またはこれらの大容量記憶デバイスからデータを受信するか、またはこれらにデータを転送するように動作可能に結合されてもよい。しかしながら、コンピュータは、このようなデバイスを有する必要はない。コンピュータプログラム命令およびデータを記憶するのに適したコンピュータ可読媒体は、あらゆる形式の不揮発性メモリ、媒体、およびメモリデバイスを

含み、例えば、ＥＰＲＯＭ、ＥＥＰＲＯＭ、フラッシュ記憶装置、磁気ディスク、例えば内部ハードディスクまたはリムーバブルディスク、光磁気ディスク、およびＣＤ－ＲＯＭおよびＤＶＤ－ＲＯＭディスク等の半導体記憶装置を含む。処理装置およびメモリは、特定用途のロジック回路によって補完されてもよく、または特定用途のロジック回路に組み込まれてもよい。

【０４９１】

本特許明細書は多くの詳細を含むが、これらは、任意の主題の範囲または特許請求の範囲を限定するものと解釈されるべきではなく、むしろ、特定の技術の特定の実施形態に特有であり得る特徴の説明と解釈されるべきである。本特許文献において別個の実施形態のコンテクストで説明されている特定の特徴は、１つの例において組み合わせて実装してもよい。逆に、１つの例のコンテクストで説明された様々な特徴は、複数の実施形態において別個にまたは任意の適切なサブコンビネーションで実装してもよい。さらに、特徴は、特定の組み合わせで作用するものとして上記に記載され、最初にそのように主張されていてもよいが、主張された組み合わせからの１つ以上の特徴は、場合によっては、組み合わせから抜粋されることができ、主張された組み合わせは、サブコンビネーションまたはサブコンビネーションのバリエーションに向けられてもよい。

【０４９２】

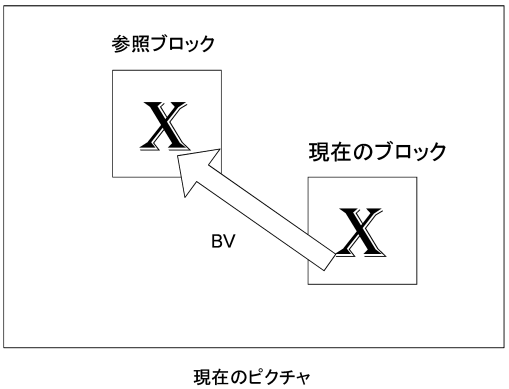
同様に、動作は図面において特定の順番で示されているが、これは、所望の結果を達成するために、このような動作が示された特定の順番でまたは連続した順番で行われること、または示された全ての動作が行われることを必要とするものと理解されるべきではない。また、本特許明細書に記載されている例における様々なシステムの構成要素の分離は、全ての実施形態においてこのような分離を必要とするものと理解されるべきではない。

【０４９３】

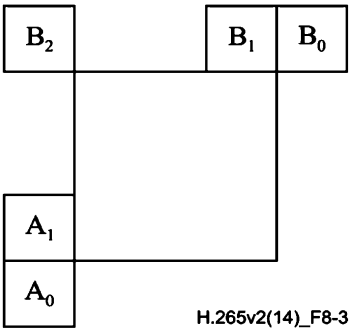
いくつかの実装形態および例のみが記載されており、この特許文献に記載され図示されているコンテンツに基づいて、他の実施形態、拡張および変形が可能である。

【図面】

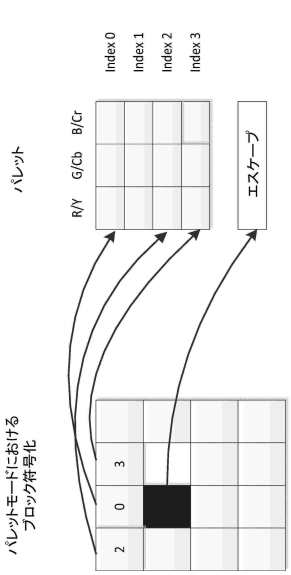
【図１】


参照ブロック
X
現在のブロック
X
BV
現在のピクチャ

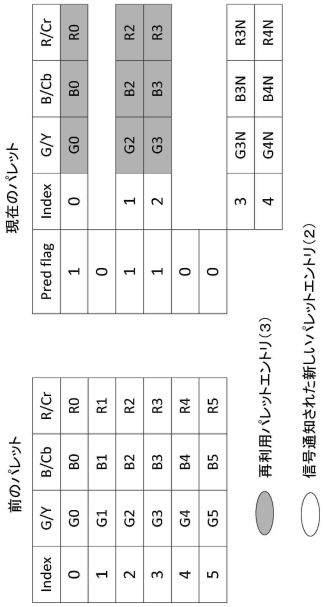
【図２】


B2
B1
B0
A1
A0
H.265v2(14)_F8-3

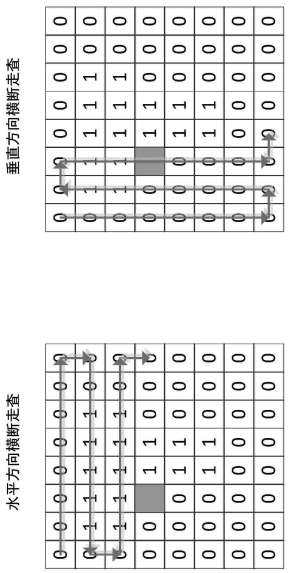
【図３】


パレットモードにおける
ブロック符号化
2
0
3
パレット
R/Y
G/Cb
B/Cr
Index 0
Index 1
Index 2
Index 3
エスケープ

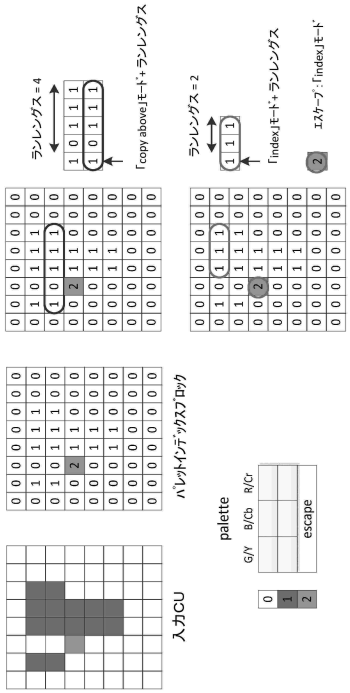
【図４】


前のパレット
Index G/Y B/Cb R/Cr
0 G0 B0 R0
1 G1 B1 R1
2 G2 B2 R2
3 G3 B3 R3
4 G4 B4 R4
5 G5 B5 R5
現在のパレット
Pred flag Index G/Y B/Cb R/Cr
1 0 G0 B0 R0
0
1 1 G2 B2 R2
1 2 G3 B3 R3
0
0
3 G3N B3N R3N
4 G4N B4N R4N
再利用パレットエントリ(3)
信号通知された新しいパレットエントリ(2)

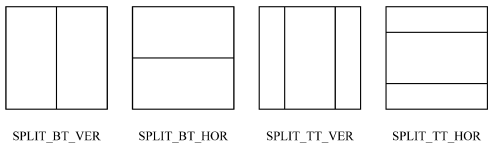
【図５】


水平方向横断走査
垂直方向横断走査

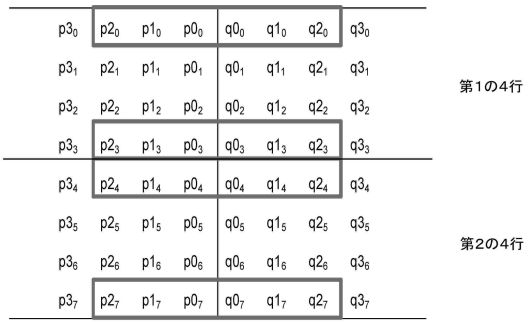
【図６】


入力CU
パレットインデックスブロック
ランレングス = 4
「copy above」モード + ランレングス
ランレングス = 2
「index」モード + ランレングス
エスケープ: 「index」モード
palette
G/Y
B/Cb
R/Cr
escape

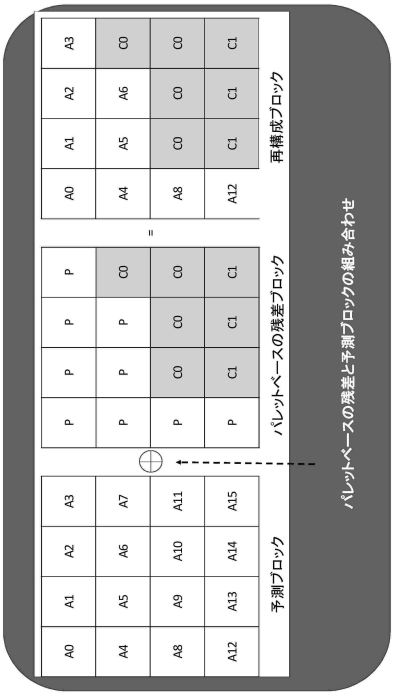
【図７】


SPLIT_BT_VER
SPLIT_BT_HOR
SPLIT_TT_VER
SPLIT_TT_HOR

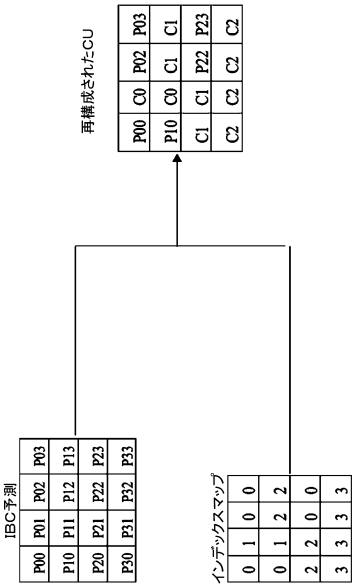
【図８】


第1の4行
第2の4行

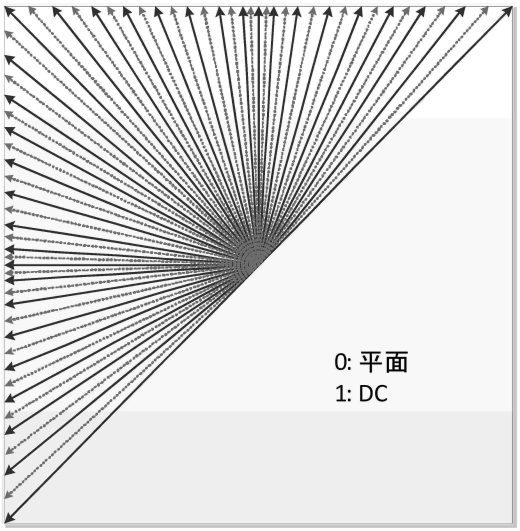
【図９】



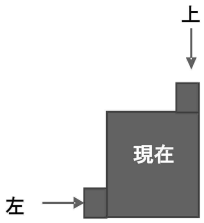
【図１０】



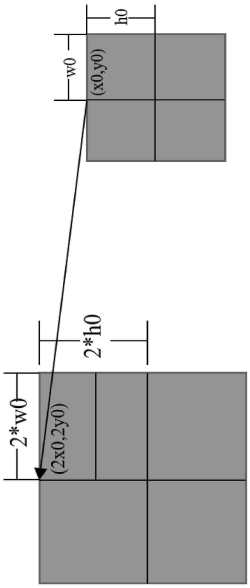
【図１１】



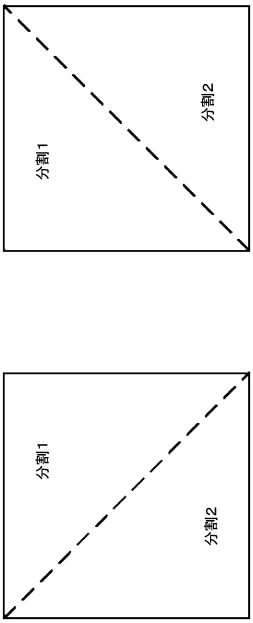
【図１２】



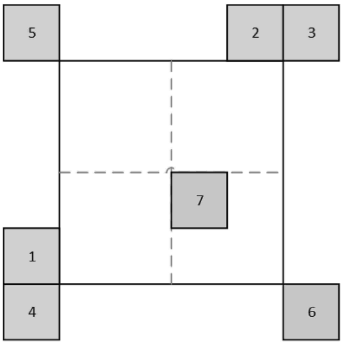
【図１３】



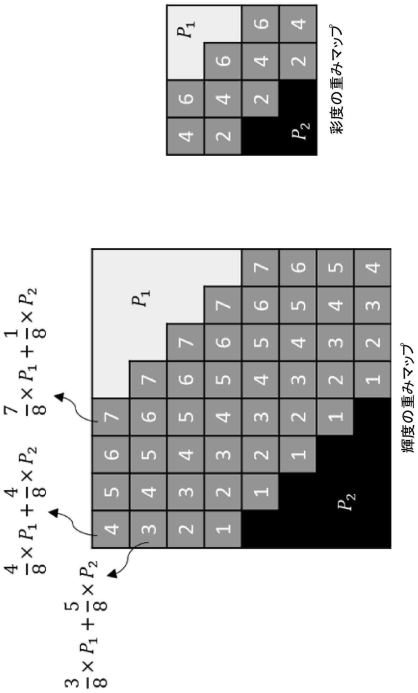
【図１４】



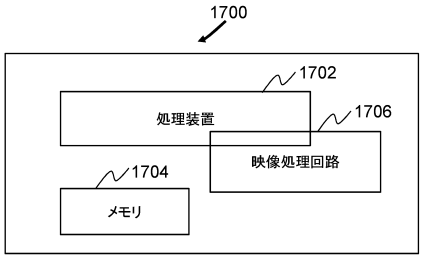
【図１５】



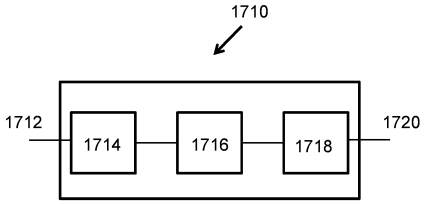
【図１６】



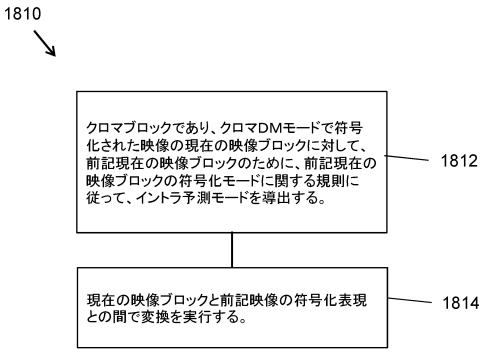
【図１７Ａ】



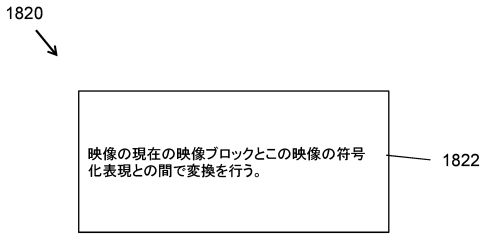
【図１７Ｂ】



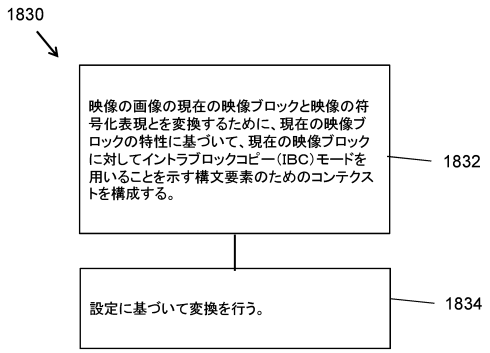
【図１８Ａ】



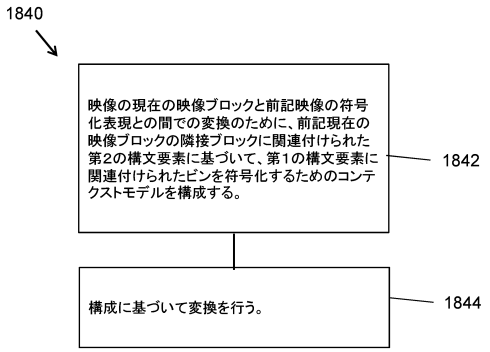
【図１８Ｂ】



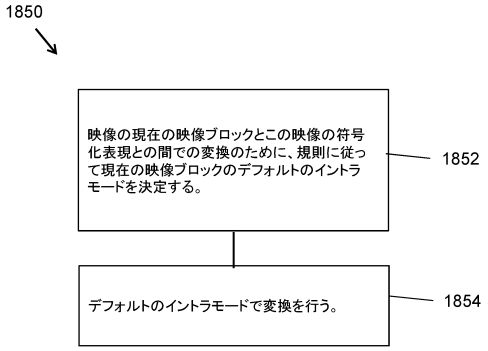
【図１８Ｃ】



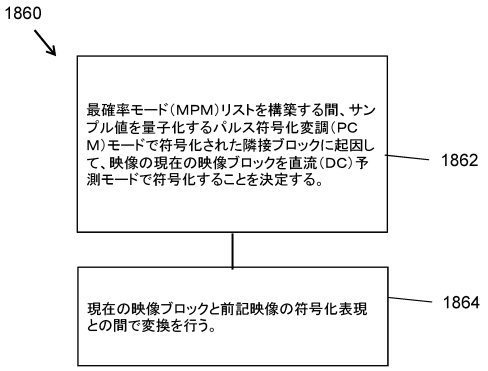
【図１８Ｄ】



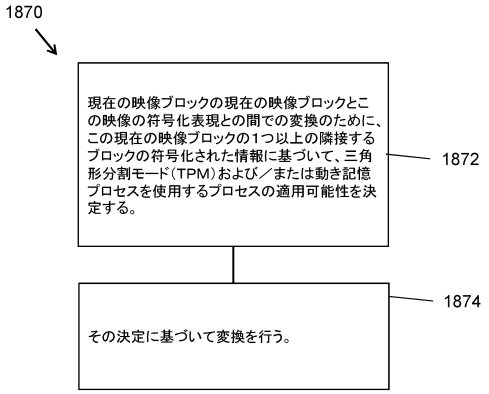
【図１８Ｅ】



【図１８Ｆ】



【図１８Ｇ】


1870
現在の映像ブロックの現在の映像ブロックとこの映像の符号化表現との間での変換のために、この現在の映像ブロックの1つ以上の隣接するブロックの符号化された情報に基づいて、三角形分割モード（TPM）および／または動き記憶プロセスを使用するプロセスの適用可能性を決定する。
1872
その決定に基づいて変換を行う。
1874

フロントページの続き

(33)優先権主張国・地域又は機関
中国(CN)
弁理士法人栄光事務所
(72)発明者　ジュー ウェイジャ
アメリカ合衆国 ９００６６ カリフォルニア州， ロサンゼルス， ウェスト ジェファーソン ブールバード １２６５５， シックスス フロア， スイート ナンバー１３７
(72)発明者　ジャン リー
アメリカ合衆国 ９００６６ カリフォルニア州， ロサンゼルス， ウェスト ジェファーソン ブールバード １２６５５， シックスス フロア， スイート ナンバー１３７
(72)発明者　シュー ジジョン
アメリカ合衆国 ９００６６ カリフォルニア州， ロサンゼルス， ウェスト ジェファーソン ブールバード １２６５５， シックスス フロア， スイート ナンバー１３７
審査官　田中 純一
(56)参考文献　特表２０２２－５１４９２１（ＪＰ，Ａ）
米国特許出願公開第２０２１／０３４４９６１（ＵＳ，Ａ１）
Man-Shu Chiang, et.al.，CE8-related: Intra prediction mode for BDPCM，Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 15th Meeting: Gothenburg, SE, 3-12 July 2019，JVET-O0296-v1，インターネット＜URL:https://jvet-experts.org/doc_end_user/documents/15_Gothenburg/wg11/JVET-O0296-v1.zip＞＜JVET-O0296-v1.docx＞，2019年06月25日，pp.1-5
Narae Choi, et.al.，CE3-related: Chroma DM modification，Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 12th Meeting: Macao, CN, 3-12 Oct. 2018，JVET-L0053-v3，インターネット＜URL:http://phenix.it-sudparis.eu/jvet/doc_end_user/documents/12_Macao/wg11/JVET-L0053-v3.zip＞＜JVET-L0053-v3.docx＞，2018年10月07日，pp.1-5
Benjamin Bross, et.al.，Versatile Video Coding (Draft 5)，Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 14th Meeting: Geneva, CH, 19-27 Mar. 2019，JVET-N1001-v9，インターネット＜URL:http://phenix.it-sudparis.eu/jvet/doc_end_user/documents/14_Geneva/wg11/JVET-N1001-v9.zip＞＜JVET-N1001-v9.docx＞，2019年06月25日，pp.i-iii,pp.53-58,113-120,131-168
Man-Shu Chiang, et.al.，CE3-related: Chroma DM derivation for IBC and PCM，Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 15th Meeting: Gothenburg, SE, 3-12 July 2019，JVET-O0275-v1，インターネット＜URL:https://jvet-experts.org/doc_end_user/documents/15_Gothenburg/wg11/JVET-O0275-v1.zip＞＜JVET-O0275-v1.docx＞，2019年06月25日，pp.1-3
Joel Sole, et.al.，Non-CE6: Delta QP signalling for palette，Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 19th Meeting: Strasbourg, FR, 17-24 Oct. 2014，JCTVC-S0043-r2，インターネット＜URL:http://phenix.it-sudparis.eu/jct/doc_end_user/documents/19_Strasbourg/wg11/JCTVC-S0043-v5.zip＞＜JCTVC-S0043-r2.doc＞，2014年10月22日，pp.1-3
(58)調査した分野　(Int.Cl.，ＤＢ名)
Ｈ０４Ｎ　７／１２
Ｈ０４Ｎ　１９／００　－　１９／９８