

(19)日本国特許庁(JP)

(12)公開特許公報(A)

(11)公開番号
特開2024-41558
(P2024-41558A)

(43)公開日 令和6年3月27日(2024.3.27)

(51)国際特許分類
H 0 4 N 19/55 (2014.01)

F I
H 0 4 N 19/55

テーマコード (参考)
5 C 1 5 9

審査請求 未請求 請求項の数 8 O L (全32頁)

(21)出願番号 (22)出願日 (出願人による申告) 令和4年度、国立研究開発法人 情報通信研究機構、「Beyond 5G研究開発促進事 業（一般型）／日米豪国際連携を通じた超カバレッジB eyond 5G無線通信・映像符号化標準化技術の研究 開発」、産業技術力強化法第17条の適用を受ける特許 出願	(71)出願人 000005049 シャープ株式会社 大阪府堺市堺区匠町1番地 (74)代理人 100161207 弁理士 西澤 和純 (74)代理人 100129115 弁理士 三木 雅夫 (74)代理人 100133569 弁理士 野村 進 (74)代理人 100131473 弁理士 覚田 功二 (72)発明者 八杉 将伸 大阪府堺市堺区匠町1番地 シャープ株 式会社内 (72)発明者 猪飼 知宏
--	---

最終頁に続く

(54)【発明の名称】 動画復号装置および動画符号化装置

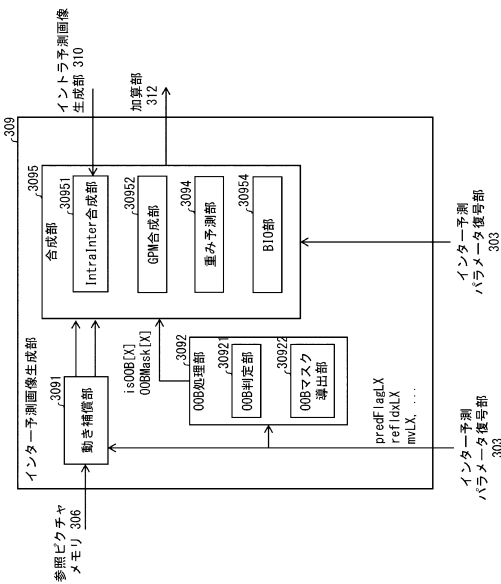
(57)【要約】

【課題】符号化効率を向上させることができる動画符号化・復号装置を提供する。

【解決手段】

動画復号装置は、参照ブロックの座標をピクチャの座標と比較することで、参照ブロックがOOB処理の対象であるか否かを判定するOOB判定部と、参照ブロックに含まれる画素の座標をピクチャの座標と比較することで、各画素の利用可否を表すマスクデータ導出するOOBマスク導出部を備え、前記OOB判定部は、上記参照ブロックが含まれる参照ピクチャに適用される処理の有無を用いて上記参照ブロックがOOB処理対象であるか否かを判定することを特徴とする。

【選択図】 図4



【特許請求の範囲】**【請求項 1】**

参照ブロックの座標をピクチャの座標と比較することで、参照ブロックがOOB処理の対象であるか否かを判定するOOB判定部と、参照ブロックに含まれる画素の座標とピクチャ境界もしくは領域境界の座標と比較することで、各画素の利用可否を表すマスクデータを導出するOOBマスク導出部を備える動画像復号装置であって、

前記OOB判定部は、上記参照ブロックが含まれる参照ピクチャに適用される処理の有無を用いて上記参照ブロックがOOB処理対象であるか否かを判定することを特徴とする動画像復号装置。

【請求項 2】

前記OOB判定部は、L0参照領域の有効領域とL1参照領域の有効領域を比べ、参照領域の有効領域が大きい参照ピクチャに対しOOB判定を無効とすることを特徴とする請求項1に記載の動画像復号装置。

【請求項 3】

前記OOB判定部は、L0参照領域とL1参照領域の有効領域の面積の比較により、上記有効領域の大きさを判定することを特徴とする請求項1に記載の動画像復号装置。

【請求項 4】

前記OOB判定部は、L0参照領域とL1参照領域の有効領域の幅もしくは高さの比較により、上記有効領域の大きさを判定することを特徴とする請求項1に記載の動画像復号装置。

【請求項 5】

上記OOB判定部は、ブロック単位の重み付き双予測(BCW:Bi-prediction with CU-level Weight)が適用されるブロックでは、OOB判定を無効とすることを特徴とする請求項1に記載の動画像復号装置。

【請求項 6】

上記OOBマスク導出部は、NxM画素(N,Mは正の整数)を単位としてマスクとして導出することを特徴とする請求項1に記載の動画像復号装置。

【請求項 7】

上記OOB判定部は、対象ブロックが所定のサイズ以下の場合はOOB判定を無効とすることを特徴とする請求項1に記載の動画像復号装置。

【請求項 8】

参照ブロックの座標をピクチャの座標と比較することで、参照ブロックがOOB処理の対象であるか否かを判定するOOB判定部と、参照ブロックに含まれる画素の座標をピクチャの座標と比較することで、各画素の利用可否を表すマスクデータを導出するOOBマスク導出部を備える動画像符号化装置であって、

前記OOB判定部は、上記参照ブロックが含まれる参照ピクチャに適用される処理の有無を用いて上記参照ブロックがOOB処理対象であるか否かを判定することを特徴とする動画像符号化装置。

【発明の詳細な説明】**【技術分野】****【0001】**

本発明の実施形態は、動画像復号装置および動画像符号化装置に関する。

【背景技術】**【0002】**

動画像を効率的に伝送または記録するために、動画像を符号化することによって符号化データを生成する動画像符号化装置、および、当該符号化データを復号することによって復号画像を生成する動画像復号装置が用いられている。

【0003】

具体的な動画像符号化方式としては、例えば、H.264/AVCやHEVC(High-Efficiency Video Coding)、Versatile Video Coding(VVC)方式などが挙げられる。

10

20

30

40

50

【0004】

このような動画像符号化方式においては、動画像を構成する画像（ピクチャ）は、画像を分割することにより得られるスライス、スライスを分割することにより得られる符号化ツリーユニット（CTU：Coding Tree Unit）、符号化ツリーユニットを分割することにより得られる符号化単位（符号化ユニット（Coding Unit：CU）と呼ばれることもある）、及び、符号化単位を分割することにより得られる変換ユニット（TU：Transform Unit）からなる階層構造により管理され、CU毎に符号化／復号される。

【0005】

また、このような動画像符号化方式においては、通常、入力画像を符号化／復号することによって得られる局所復号画像に基づいて予測画像が生成され、当該予測画像を入力画像（原画像）から減算して得られる予測誤差（「差分画像」または「残差画像」と呼ぶこともある）が符号化される。予測画像の生成方法としては、画面間予測（インター予測）、および、画面内予測（イントラ予測）が挙げられる。VVCでは、図8(a)のように、動きベクトルの座標をラップアラウンド(wrap-around)処理して、水平方向の座標系をピクチャの左端と右端を連続させて動き補償することができる。従って、360度パノラマ画像や360度画像のようにピクチャの左右端が空間的に連続する映像にラップアラウンド処理を適用すると、符号化効率を向上させることができる。また、VVCでは、図8(b)のようにピクチャ単位で解像度を变化させ動き補償する参照ピクチャリサンプリング(RPR: Reference Picture Resampling)処理が可能である。インターネット上での動画像配信のような伝送レートが変動するサービスにRPR処理を適用すると、画質を向上させることができる。

【0006】

また、非特許文献1には、参照画像のピクチャ外の領域を含む双予測の一方の参照ブロックの一部あるいは全ての領域を、他方の参照ブロックの一部あるいは全ての領域で代替するOOB(Out-Of-Boundary)処理技術が開示されている。

【先行技術文献】

【非特許文献】

【0007】

【非特許文献1】“AHG12: Enhanced bi-directional motion compensation”, Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1 /SC 29 25th Meeting, by teleconference, JVET-Y0125

【発明の概要】

【発明が解決しようとする課題】

【0008】

非特許文献1に記載の方法では、参照ブロックがピクチャ外の領域を含む場合、ピクチャ外の領域を示すマスクを導出しつつ、双予測画像を導出する。従って計算量が多いという課題がある。

【課題を解決するための手段】

【0009】

上記の課題を解決するために、本発明の一態様に係る動画像復号装置は、

【0010】

参照ブロックの座標をピクチャの座標と比較することで、参照ブロックがOOB処理の対象であるかどうかを判定するOOB判定部と、参照ブロックに含まれる画素の座標とピクチャ境界もしくは領域境界の座標と比較することで、各画素の利用可否を表すマスクデータを導出するOOBマスク導出部を備え、前記OOB判定部は、上記参照ブロックが含まれる参照ピクチャに適用される処理の有無を用いて上記参照ブロックがOOB処理対象であるか否かを判定することを特徴とする。

【0011】

本発明の一態様に係る動画像復号装置において、前記OOB判定部は、L0参照領域の有効領域とL1参照領域の有効領域を比べ、参照領域の有効領域が大きい参照ピクチャに対

10

20

30

40

50

しOOB判定を無効とすることを特徴とする。

【発明の効果】

【0012】

本発明の態様によれば、動画像符号化・復号処理においてOOB処理にかかる計算量を削減することができる。

【図面の簡単な説明】

【0013】

【図1】本実施形態に係る画像伝送システムの構成を示す概略図である。

【図2】符号化ストリームのデータの階層構造を示す図である。

【図3】動画像復号装置の構成を示す概略図である。

10

【図4】インター予測画像生成部の構成を示す概略図である。

【図5】インター予測パラメータ導出部の構成を示す概略図である。

【図6】動画像復号装置の概略的動作を説明するフローチャートである。

【図7】動画像符号化装置の構成を示すブロック図である。

【図8】従来技術のラップアラウンド処理と参照ピクチャリサンプリング処理の例を示す図である。

【図9】OOB処理の概略的動作を説明するフローチャートである。

【図10】OOB判定の例を示す概略図である。

【発明を実施するための形態】

【0014】

20

(第1の実施形態)

【0015】

以下、図面を参照しながら本発明の実施形態について説明する。

【0016】

図1は、本実施形態に係る画像伝送システム1の構成を示す概略図である。

【0017】

画像伝送システム1は、対象画像を符号化した符号化ストリームを送信し、送信された符号化ストリームを復号し画像を表示するシステムである。画像伝送システム1は、動画像符号化装置（画像符号化装置）11、ネットワーク21、動画像復号装置（画像復号装置）31、及び動画像表示装置（画像表示装置）41を含んで構成される。

30

【0018】

動画像符号化装置11には画像Tが入力される。

【0019】

ネットワーク21は、動画像符号化装置11が生成した符号化ストリームTeを動画像復号装置31に伝送する。ネットワーク21は、インターネット（Internet）、広域ネットワーク（WAN:Wide Area Network）、小規模ネットワーク（LAN:Local Area Network）またはこれらの組み合わせである。ネットワーク21は、必ずしも双方向の通信網に限らず、地上デジタル放送、衛星放送等の放送波を伝送する一方向の通信網であっても良い。また、ネットワーク21は、DVD（Digital Versatile Disc:登録商標）、BD（Blue-ray Disc:登録商標）等の符号化ストリームTeを記録した記憶媒体で代替されても良い。

40

【0020】

動画像復号装置31は、ネットワーク21が伝送した符号化ストリームTeのそれぞれを復号し、復号した1または複数の復号画像Tdを生成する。

【0021】

動画像表示装置41は、動画像復号装置31が生成した1または複数の復号画像Tdの全部または一部を表示する。動画像表示装置41は、例えば、液晶ディスプレイ、有機EL（Electro-luminescence）ディスプレイ等の表示デバイスを備える。ディスプレイの形態としては、据え置き、モバイル、HMD等が挙げられる。また、動画像復号装置31が高い処理能力を有する場合には、画質の高い画像を表示し、より低い処理能力しか有しな

50

い場合には、高い処理能力、表示能力を必要としない画像を表示する。

【0022】

<演算子>

【0023】

本明細書で用いる演算子を以下に記載する。

【0024】

>>は右ビットシフト、<<は左ビットシフト、&はビットワイズAND、|はビットワイズOR、|=はOR代入演算子であり、||は論理和を示す。

【0025】

x?y:zは、xが真（0以外）の場合にy、xが偽（0）の場合にzをとる3項演算子である。

10

【0026】

Clip3(a,b,c)は、cをa以上b以下の値にクリップする関数であり、c<aの場合にはaを返し、c>bの場合にはbを返し、その他の場合にはcを返す関数である（ただし、a<=b）。

【0027】

ClipH(o, W, x)は、x<0の場合x、x>W-1の場合x-o、それ以外の場合xを返す関数である。

【0028】

sign(a)はa>0の場合1、a==0の場合1、a<0の場合-1を返す関数である。

20

【0029】

abs(a)はaの絶対値を返す関数である。

【0030】

Int(a)はaの整数値を返す関数である。

【0031】

floor(a)はa以下の最大の整数を返す関数である。

【0032】

ceil(a)はa以上の最小の整数を返す関数である。

【0033】

a/dはdによるaの除算（小数点以下切り捨て）を表す。

30

【0034】

<符号化ストリームTeの構造>

【0035】

本実施形態に係る動画像符号化装置11および動画像復号装置31の詳細な説明に先立って、動画像符号化装置11によって生成され、動画像復号装置31によって復号される符号化ストリームTeのデータ構造について説明する。

【0036】

図2は、符号化ストリームTeにおけるデータの階層構造を示す図である。符号化ストリームTeは、例示的に、シーケンス、およびシーケンスを構成する複数のピクチャを含む。図2の(a)~(f)は、それぞれ、シーケンスSEQを既定する符号化ビデオシーケンス、ピクチャPICTを規定する符号化ピクチャ、スライスSを規定する符号化スライス、スライスデータを規定する符号化スライスデータ、符号化スライスデータに含まれる符号化ツリーユニット、符号化ツリーユニットに含まれる符号化ユニットを示す図である。

40

【0037】

（符号化ビデオシーケンス）

【0038】

符号化ビデオシーケンスでは、処理対象のシーケンスSEQを復号するために動画像復号装置31が参照するデータの集合が規定されている。シーケンスSEQは、図2に示すように、ビデオパラメータセット（Video Parameter Set）、シーケンスパラメータセットSPS（Sequence Parameter Set）、ピクチャパラメータセットPPS（Picture

50

Parameter Set)、Adaptation Parameter Set(APS)、ピクチャPICT、及び、付加拡張情報SEI (Supplemental Enhancement Information) を含んでいる。

【0039】

ビデオパラメータセットVPSは、複数のレイヤから構成されている動画像において、複数の動画像に共通する符号化パラメータの集合および動画像に含まれる複数のレイヤおよび個々のレイヤに関連する符号化パラメータの集合が規定されている。

【0040】

シーケンスパラメータセットSPSでは、対象シーケンスを復号するために動画像復号装置31が参照する符号化パラメータの集合が規定されている。例えば、ピクチャの幅や高さが規定される。なお、SPSは複数存在してもよい。その場合、PPSから複数のSPSの何れかを選択する。

【0041】

ピクチャパラメータセットPPSでは、対象シーケンス内の各ピクチャを復号するために動画像復号装置31が参照する符号化パラメータの集合が規定されている。例えば、ピクチャの復号に用いられる量子化幅の基準値 (pic_init_qp_minus26) や重み付き予測の適用を示すフラグ (weighted_pred_flag) が含まれる。なお、PPSは複数存在してもよい。その場合、対象シーケンス内の各ピクチャから複数のPPSの何れかを選択する。

【0042】

(符号化ピクチャ)

【0043】

符号化ピクチャでは、処理対象のピクチャPICTを復号するために動画像復号装置31が参照するデータの集合が規定されている。ピクチャPICTは、図2に示すように、スライス0～スライスNS-1を含む (NSはピクチャPICTに含まれるスライスの総数)。

【0044】

なお、以下、スライス0～スライスNS-1のそれぞれを区別する必要がある場合、符号の添え字を省略して記述することがある。また、以下に説明する符号化ストリームTeに含まれるデータであって、添え字を付している他のデータについても同様である。

【0045】

(符号化スライス)

【0046】

符号化スライスでは、処理対象のスライスSを復号するために動画像復号装置31が参照するデータの集合が規定されている。スライスは、図2に示すように、スライスヘッダ、および、スライスデータを含んでいる。

【0047】

スライスヘッダには、対象スライスの復号方法を決定するために動画像復号装置31が参照する符号化パラメータ群が含まれる。スライスタイプを指定するスライスタイプ指定情報 (slice_type) は、スライスヘッダに含まれる符号化パラメータの一例である。

【0048】

スライスタイプ指定情報により指定可能なスライスタイプとしては、(1) 符号化の際にイントラ予測のみを用いるIスライス、(2) 符号化の際に単方向予測、または、イントラ予測を用いるPスライス、(3) 符号化の際に単方向予測、双方向予測、または、イントラ予測を用いるBスライスなどが挙げられる。なお、インター予測は、単予測、双予測に限定されず、より多くの参照ピクチャを用いて予測画像を生成してもよい。以下、P、Bスライスと呼ぶ場合には、インター予測を用いることができるブロックを含むスライスを指す。

【0049】

なお、スライスヘッダは、ピクチャパラメータセットPPSへの参照 (pic_parameter_set_id) を含んでもよい。

【0050】

(符号化スライスデータ)

10

20

30

40

50

【0051】

符号化スライスデータでは、処理対象のスライスデータを復号するために動画像復号装置31が参照するデータの集合が規定されている。スライスデータは、図2(d)に示すように、CTUを含んでいる。CTUは、スライスを構成する固定サイズ（例えば64x64）のブロックであり、最大符号化単位（LCU:Largest Coding Unit）と呼ぶこともある。

【0052】

（符号化ツリーユニット）

【0053】

図2には、処理対象のCTUを復号するために動画像復号装置31が参照するデータの集合が規定されている。CTUは、再帰的な4分木分割（QT（Quad Tree）分割）、2分木分割（BT（Binary Tree）分割）あるいは3分木分割（TT（Ternary Tree）分割）により、符号化処理の基本的な単位である符号化ユニットCUに分割される。BT分割とTT分割を合わせてマルチツリー分割（MT（Multi Tree）分割）と呼ぶ。再帰的な4分木分割により得られる木構造のノードのことを符号化ノード（Coding Node）と称する。4分木、2分木、及び3分木の間ノードは、符号化ノードであり、CTU自身も最上位の符号化ノードとして規定される。また、最下位の符号化ノードは符号化ユニットとして規定される。

【0054】

（符号化ユニット）

【0055】

図2は、処理対象の符号化ユニットを復号するために動画像復号装置31が参照するデータの集合が規定されている。具体的には、CUは、CUヘッダCUH、予測パラメータ、変換パラメータ、量子化変換係数等から構成される。CUヘッダでは予測モード等が規定される。

【0056】

予測処理は、CU単位で行われる場合と、CUをさらに分割したサブCU単位で行われる場合がある。CUとサブCUのサイズが等しい場合には、CU中のサブCUは1つである。CUがサブCUのサイズよりも大きい場合、CUはサブCUに分割される。たとえばCUが8x8、サブCUが4x4の場合、CUは水平2分割、垂直2分割からなる、4つのサブCUに分割される。

【0057】

予測の種類（予測モード）は、イントラ予測（MODE_INTRA）と、インター予測（MODE_INTER）、イントラブロックコピー（MODE_IBC）がある。イントラ予測は、同一ピクチャ内の予測であり、インター予測は、互いに異なるピクチャ間（例えば、表示時刻間、レイヤ画像間）で行われる予測処理を指す。

【0058】

変換・量子化処理はCU単位で行われるが、量子化変換係数は4x4等のサブブロック単位でエントロピー符号化してもよい。

【0059】

（予測パラメータ）

【0060】

予測画像は、ブロックに付随する予測パラメータによって導出される。予測パラメータには、イントラ予測とインター予測の予測パラメータがある。

【0061】

（インター予測の予測パラメータ）

【0062】

インター予測の予測パラメータについて説明する。インター予測パラメータは、予測リスト利用フラグpredFlagL0とpredFlagL1、参照ピクチャインデックスrefIdxL0とrefIdxL1、動きベクトルmvL0とmvL1から構成される。predFlagL0、predFlagL1は、参照ピクチャリスト（L0リスト、L1リスト）が用いられるか否かを示すフラグであ

10

20

30

40

50

り、値が1の場合に対応する参照ピクチャリストが用いられる。なお、本明細書中「XXであるか否かを示すフラグ」と記す場合、フラグが0以外（たとえば1）をXXである場合、0をXXではない場合とし、論理否定、論理積などでは1を真、0を偽と扱う（以下同様）。但し、実際の装置や方法では真値、偽値として他の値を用いることもできる。

【0063】

インター予測パラメータを導出するためのシンタックス要素には、例えば、マージフラグmerge_flag(general_merge_flag)、マージインデックスmerge_idx、merge_subblock_flag、regular_merge_flag、ciip_flag、merge_gpm_partition_idx、merge_gpm_idx0、merge_gpm_idx1、inter_pred_idc、参照ピクチャインデックスrefIdxLX、mvp_LX_idx、差分ベクトルmvdLX、動きベクトル精度モードamvr_modeがある。merge_subblock_flagはサブブロック単位のインター予測を用いるかを示すフラグである。regular_merge_flagは通常マージモードあるいはMMVDが使われるかを示すフラグである。ciip_flagはCIIP (combined inter-picture merge and intra-picture prediction) モードを用いるか否かを示すフラグである。merge_gpm_partition_idxはGPMモードの分割形状を示すインデックスである。merge_gpm_idx0、merge_gpm_idx1はGPMモードのマージインデックスを示すインデックスである。inter_pred_idcはAMVPモードで用いる参照ピクチャを選択するためのインター予測識別子である。mvp_LX_idxは動きベクトルを導出するための予測ベクトルインデックスである。

10

【0064】

(参照ピクチャリスト)

20

【0065】

参照ピクチャリストは、参照ピクチャメモリ306に記憶された参照ピクチャからなるリストである。個々のCUでは、参照ピクチャリストRefPicListX (X=0または1) 中のどのピクチャを実際に参照するかをrefIdxLXで指定する。なお、LXは、L0予測とL1予測を区別しない場合に用いられる記述方法であり、以降では、LXをL0、L1に置き換えることでL0リストに対するパラメータとL1リストに対するパラメータを区別する。

【0066】

(マージ予測とAMVP予測)

【0067】

予測パラメータの復号（符号化）方法には、マージ予測（merge）モード（マージモード）とAMVP (Advanced Motion Vector Prediction、適応動きベクトル予測) モードがあり、general_merge_flagは、これらを識別するためのフラグである。マージモードは、動きベクトル差分の一部もしくは全てを省略する予測モードであり、予測リスト利用フラグpredFlagLX、参照ピクチャインデックスrefIdxLX、動きベクトルmvLXを符号化データに含めずに、既に処理した近傍ブロックの予測パラメータ等から導出するモードである。AMVPモードは、inter_pred_idc、refIdxLX、mvLXを符号化データに含めるモードである。なお、mvLXは、予測ベクトルmvpLXを識別するmvp_LX_idxと差分ベクトルmvdLXとして符号化される。また動きベクトル差分を省略・簡素化する予測モードの総称を一般マージモード (general merge mode) と呼び、general_merge_flagにより一般マージモードとAMVP予測を選択してもよい。

30

40

【0068】

general_merge_flagが1の場合、別途regular_merge_flagを伝送してもよい。regular_merge_flagが1である場合に、通常マージモードもしくはMMVDを選択し、それ以外の場合にCIIPモードもしくはGPMモードを選択してもよい。CIIPモードはインター予測画像とイントラ予測画像の重み付け和により予測画像を生成する。GPMモードは、対象CUを線分で分割される2つの非矩形の予測単位として、予測画像を生成する。

【0069】

inter_pred_idcは、参照ピクチャの種類および数を示す値であり、PRED_L0、PRED_L1、PRED_BIの何れかの値をとる。PRED_L0、PRED_L1は、各々L0リスト、L1

50

リストで管理された 1 枚の参照ピクチャを用いる単予測を示す。PRED_BIはL0リストとL1リストで管理された 2 枚の参照ピクチャを用いる双予測を示す。

【0070】

merge_idxは、処理が完了したブロックから導出される予測パラメータ候補（マージ候補）のうち、いずれの予測パラメータを対象ブロックの予測パラメータとして用いるかを示すインデックスである。

【0071】

（動きベクトル）

【0072】

mvLXは、異なる 2 つのピクチャ上のブロック間のシフト量を示す。mvLXに関する予測ベクトル、差分ベクトルを、それぞれmvpLX、mvdLXと呼ぶ。

【0073】

（インター予測識別子inter_pred_idcと予測リスト利用フラグpredFlagLX）

【0074】

inter_pred_idcと、predFlagL0、predFlagL1の関係は以下のとおりであり、相互に変換可能である。

【0075】

inter_pred_idc = (predFlagL1 < 1) + predFlagL0

【0076】

predFlagL0 = inter_pred_idc & 1

【0077】

predFlagL1 = inter_pred_idc > 1

【0078】

なお、インター予測パラメータは、予測リスト利用フラグを用いても良いし、インター予測識別子を用いてもよい。また、予測リスト利用フラグを用いた判定は、インター予測識別子を用いた判定に置き替えてもよい。逆に、インター予測識別子を用いた判定は、予測リスト利用フラグを用いた判定に置き替えてもよい。

【0079】

（双予測biPredの判定）

【0080】

双予測であるかのフラグbiPredは、2 つの予測リスト利用フラグがともに 1 であるかによって導出できる。例えば以下の式で導出できる。

【0081】

biPred = (predFlagL0==1 && predFlagL1==1)

【0082】

あるいは、biPredは、インター予測識別子が 2 つの予測リスト（参照ピクチャ）を使うことを示す値であるか否かによっても導出できる。例えば以下の式で導出できる。

【0083】

biPred = (inter_pred_idc==PRED_BI) ? 1 : 0

【0084】

（動画復号装置の構成）

【0085】

本実施形態に係る動画復号装置 31（図 3）の構成について説明する。

【0086】

動画復号装置 31は、エントロピー復号部 301、パラメータ復号部（予測画像復号装置） 302、ループフィルタ 305、参照ピクチャメモリ 306、予測パラメータメモリ 307、予測画像生成部（予測画像生成装置） 308、逆量子化・逆変換部 311、及び加算部 312、予測パラメータ導出部 320を含んで構成される。なお、後述の動画符号化装置 11に合わせ、動画復号装置 31にループフィルタ 305が含まれない構成もある。

【0087】

10

20

30

40

50

パラメータ復号部302は、さらに、ヘッダ復号部3020、CT情報復号部3021、及びCU復号部3022（予測モード復号部）を備えており、CU復号部3022はさらにTU復号部3024を備えている。これらを総称して復号モジュールと呼んでもよい。ヘッダ復号部3020は、符号化データからVPS、SPS、PPS、APSなどのパラメータセット情報、スライスヘッダ（スライス情報）を復号する。CT情報復号部3021は、符号化データからCTを復号する。CU復号部3022は符号化データからCUを復号する。TU復号部3024は符号化データからCUを復号する。

【0088】

TU復号部3024は、TUに予測誤差が含まれている場合、符号化データからQP更新情報と量子化変換係数を復号する。QP更新情報は、量子化パラメータQPの予測値である量子化パラメータ予測値 qP_{pred} からの差分値である。

10

【0089】

予測画像生成部308は、インター予測画像生成部309（図4）及びイントラ予測画像生成部310を含んで構成される。

【0090】

予測パラメータ導出部320は、インター予測パラメータ導出部303（図5）及びイントラ予測パラメータ導出部を含んで構成される。

【0091】

また、以降では処理の単位としてCTU、CUを使用した例を記載するが、この例に限らず、サブCU単位で処理をしてもよい。あるいはCTU、CUをブロック、サブCUをサブブロックと読み替え、ブロックあるいはサブブロック単位の処理としてもよい。

20

【0092】

エントロピー復号部301は、外部から入力された符号化ストリーム Te に対してエントロピー復号を行って、個々の符号（シンタックス要素）を復号する。エントロピー符号化には、シンタックス要素の種類や周囲の状況に応じて適応的に選択したコンテキスト（確率モデル）を用いてシンタックス要素を可変長符号化する方式と、あらかじめ定められた表、あるいは計算式を用いてシンタックス要素を可変長符号化する方式がある。

【0093】

エントロピー復号部301は、復号した符号をパラメータ復号部302に出力する。復号した符号とは、例えば、予測モード $predMode$ 、 $general_merge_flag$ 、 $merge_idx$ 、 $inter_pred_idc$ 、 $refIdxLX$ 、 mvp_LX_idx 、 $mvdLX$ 、 $amvr_mode$ 等である。どの符号を復号するかは、パラメータ復号部302の指示に基づいて行われる。

30

【0094】

（基本フロー）

【0095】

図6は、動画画像復号装置31の概略的動作を説明するフローチャートである。

【0096】

（S1100：パラメータセット情報復号）ヘッダ復号部3020は、符号化データからVPS、SPS、PPSなどのパラメータセット情報を復号する。

【0097】

40

（S1200：スライス情報復号）ヘッダ復号部3020は、符号化データからスライスヘッダ（スライス情報）を復号する。

【0098】

以下、動画画像復号装置31は、対象ピクチャに含まれる各CTUについて、S1300からS5000の処理を繰り返すことにより各CTUの復号画像を導出する。

【0099】

（S1300：CTU情報復号）CT情報復号部3021は、符号化データからCTUを復号する。

【0100】

（S1400：CT情報復号）CT情報復号部3021は、符号化データからCTを復号する。

50

【0101】

(S1500:CU復号)CU復号部3022はS1510、S1520を実施して、符号化データからCUを復号する。

【0102】

(S1510:CU情報復号)CU復号部3022は、符号化データからCU情報、予測情報、TU分割フラグ、CU残差フラグ等を復号する。

【0103】

(S1520:TU情報復号)TU復号部3024は、TUに予測誤差が含まれている場合に、符号化データから量子化予測誤差等を復号する。

【0104】

(S2000:予測画像生成)予測画像生成部308は、対象CUに含まれる各ブロックについて、予測情報に基づいて予測画像を生成する。

【0105】

(S3000:逆量子化・逆変換)逆量子化・逆変換部311は、対象CUに含まれる各TUについて、逆量子化・逆変換処理を実行する。

【0106】

(S4000:復号画像生成)加算部312は、予測画像生成部308より供給される予測画像と、逆量子化・逆変換部311より供給される予測誤差とを加算することによって、対象CUの復号画像を生成する。

【0107】

(S5000:ループフィルタ)ループフィルタ305は、復号画像にデブロッキングフィルタ、SAO、ALFなどのループフィルタをかけ、復号画像を生成する。

【0108】

ループフィルタ305は、符号化ループ内に設けたフィルタで、ブロック歪やリングング歪を除去し、画質を改善するフィルタである。ループフィルタ305は、加算部312が生成したCUの復号画像に対し、デブロッキングフィルタ、サンプル適応オフセット(SAO)、適応ループフィルタ(ALF)等のフィルタを施す。

【0109】

参照ピクチャメモリ306は、加算部312が生成したCUの復号画像を、対象ピクチャ及び対象CU毎に予め定めた位置に記憶する。

【0110】

予測パラメータメモリ307は、復号対象のCTUあるいはCU毎に予め定めた位置に予測パラメータを記憶する。具体的には、予測パラメータメモリ307は、パラメータ復号部302が復号したパラメータ及びエントロピー復号部301が分離した予測モードpredMode等を記憶する。

【0111】

予測画像生成部308には、predMode、予測パラメータ等が入力される。また、予測画像生成部308は、参照ピクチャメモリ306から参照ピクチャを読み出す。予測画像生成部308は、predModeが示す予測モードで、予測パラメータと読み出した参照ピクチャ(参照ブロック)を用いてブロックもしくはサブブロックの予測画像を生成する。ここで、参照ブロックとは、参照ピクチャ上の画素の集合(通常矩形であるのでブロックと呼ぶ)であり、予測画像を生成するために参照する領域である。

【0112】

(インター予測パラメータ導出部の構成)

【0113】

図5に示すように、インター予測パラメータ導出部303は、パラメータ復号部302から入力されたシンタックス要素に基づいて、予測パラメータメモリ307に記憶された予測パラメータを参照してインター予測パラメータを導出する。また、インター予測パラメータをインター予測画像生成部309、予測パラメータメモリ307に出力する。インター予測パラメータ導出部303及びその内部の要素であるAMVP予測パラメータ導出部3032、

10

20

30

40

50

マージ予測パラメータ導出部3036、MMVD予測部30376、MV加算部3038は、動画画像符号化装置、動画画像復号装置で共通する手段であるので、これらを総称して動きベクトル導出部（動きベクトル導出装置）と称してもよい。

【0114】

general_merge_flagが1、すなわち、マージ予測モードを示す場合、merge_idxを導出し、マージ予測パラメータ導出部3036に出力する。

【0115】

general_merge_flagが0、すなわち、AMVP予測モードを示す場合、AMVP予測パラメータ導出部3032はinter_pred_idc、refIdxLXかmvp_LX_idxからmvpLXを導出する。

10

【0116】

（MV加算部）

【0117】

MV加算部3038では導出されたmvpLXとmvdLXを加算し、mvLXを導出する。

【0118】

（マージ予測）

【0119】

マージ予測パラメータ導出部3036は、予測パラメータ（predFlagLX、mvLX、refIdxLX）を含んで構成されるマージ候補を導出し、マージ候補リストを構成する。さらにマージ予測パラメータ導出部3036は、マージ候補リストに含まれるマージ候補のうち、merge_idxが示すマージ候補Nの動き情報（mvLXN[0]、mvLXN[1]）、predFlagLXN、refIdxLXNを対象ブロックのインター予測パラメータとして選択する。マージ予測パラメータ導出部3036は選択したマージ候補のインター予測パラメータを予測パラメータメモリ307に記憶するとともに、インター予測画像生成部309に出力する。

20

【0120】

（インター予測画像生成部309）

【0121】

predModeがインター予測を示す場合、インター予測画像生成部309は、インター予測パラメータ導出部303から入力されたインター予測パラメータと参照ピクチャを用いてインター予測によりブロックもしくはサブブロックの予測画像を生成する。

30

【0122】

図4は、本実施形態に係る予測画像生成部308に含まれるインター予測画像生成部309の構成を示す概略図である。インター予測画像生成部309は、動き補償部（予測画像生成装置）3091、OOB処理部3092、合成部3095を含んで構成される。合成部3095は、重み予測部3094を含んで構成される。

【0123】

（動き補償）

【0124】

動き補償部3091（補間画像生成部3091）は、インター予測パラメータ導出部303から入力された、インター予測パラメータ（predFlagLX、refIdxLX、mvLX）に基づいて、参照ピクチャメモリ306から参照ブロックを読み出すことによって補間画像（動き補償画像）を生成する。参照ブロックは、refIdxLXで指定された参照ピクチャRefPicLX上で、対象ブロックの位置からmvLXシフトした位置のブロックである。ここで、mvLXが整数精度でない場合には、動き補償フィルタと呼ばれる小数位置の画素を生成するためのフィルタを施して、補間画像を生成する。

40

【0125】

動き補償部3091は、bW*bHサイズのブロックの左上座標(xPb,yPb)、予測ブロック内座標(xL,yL)、動きベクトル(mvLX[0],mvLX[1])に対応する整数位置(xInt,yInt)および位相(xFrac,yFrac)を以下の式で導出する(MC-P1)。

【0126】

50

```

    xInt = xPb+(mvLX[0]>>(log2(MVPREC)))+xL
【0127】
    xFrac = mvLX[0]&(MVPREC-1)
【0128】
    yInt = yPb+(mvLX[1]>>(log2(MVPREC)))+yL
【0129】
    yFrac = mvLX[1]&(MVPREC-1)
【0130】

```

ここで、MVPRECはmvLXの精度（1/MVPREC画素精度）、 $\log_2 MVPREC = (\log_2(MVPREC))$ 、 $x=0 \cdots bW-1$ 、 $y=0 \cdots bH-1$ を示す。例えばMVPREC=16であってもよい。

10

```
【0131】
```

なお、RPRを行うために、後述の(MC-P2)のように導出してよい。さらに、(MC-P1)、(MC-P2)で導出された(xInt,yInt)に対して、動き補償部3091は、ラップアラウンドのために位置を修正してもよい。

```
【0132】
```

サブピクチャ境界をピクチャ境界とするフラグが有効である場合（sps_subpic_treated_as_pic_flag == 1）かつ、参照画像のサブピクチャの数が1より大きい場合（参照画像refPicLXのためのsps_num_subpics_minus1 > 0）

```
【0133】
```

```

xInt = Clip3( SubpicLeftBoundaryPos, SubpicRightBoundaryPos, refWrap
aroundEnabledFlag ? ClipH( ( PpsRefWraparoundOffset ) * MinCbSizeY,
picW, xInt ) : xInt )

```

20

```
【0134】
```

```

yInt = Clip3( SubpicTopBoundaryPos, SubpicBotBoundaryPos, yInt )

```

```
【0135】
```

ここで、SubpicLeftBoundaryPos、SubpicRightBoundaryPos、SubpicTopBoundaryPos、SubpicBottomBoundaryPosは、各々サブピクチャの左、右、上、下の境界位置。

```
【0136】
```

上記以外の場合（サブピクチャ境界をピクチャ境界とするフラグが無効である場合（sps_subpic_treated_as_pic_flag == 0）もしくは、参照画像のサブピクチャの数が1の場合（参照画像refPicLXのためのsps_num_subpics_minus1 == 0）

30

```
【0137】
```

```

xInt = Clip3( 0, picW - 1, refWraparoundEnabledFlag ? ClipH( ( PpsRef
WraparoundOffset ) * MinCbSizeY, picW, xInt ) : xInt )

```

```
【0138】
```

```

yInt = Clip3( 0, picH - 1, yInt )

```

```
【0139】
```

ここで、refWraparoundEnabledFlag = pps_ref_wraparound_enabled_flag && !refPicIsScaled、PpsRefWraparoundOffset = $\text{pps_pic_width_in_luma_samples} / \text{MinCbSizeY} - \text{pps_pic_width_minus_wraparound_offset}$ 。MinCbSizeYは所定の定数もしくは変数（たとえば4）、pps_pic_width_minus_wraparound_offsetはラップアラウンドの位置を示す符号化データから復号されるオフセット。

40

```
【0140】
```

動き補償部3091は、参照ピクチャrefImgに補間フィルタを用いて水平補間処理を行うことで、一時的画像temp[][]を導出する。以下のΣは $k=0 \cdots NTAP-1$ のkに関する和、mcFilter[Frac][k]は位相Fracにおけるk番目の補間フィルタ係数であり、shift1は値のレンジを調整する正規化パラメータ、offset1=1 << (shift1-1)である。

```
【0141】
```

```

temp[x][y] = ( Σ mcFilter[xFrac][k]*refImg[xInt+k-NTAP/2+1][yInt]+of

```

50

fset1)>> shift1

【0142】

続いて、動き補償部3091は、一時的画像temp[][]を垂直補間処理により、補間画像Pred[][]を導出する。以下の Σ は $k=0..NTAP-1$ の k に関する和、shift2は値のレンジを調整する正規化パラメータ、offset2=1<<(shift2-1)である。

【0143】

$$Pred[x][y] = (\Sigma mcFilter[yFrac][k]*temp[x][y+k-NTAP/2+1]+offset2)>> shift2 \quad (式MC-1)$$

【0144】

(ピクチャ範囲外(OOB)処理)

10

【0145】

OOB処理部3092はOOB判定部30921およびOOBマスク導出部30922を含む。OOB判定部30921は、インター予測パラメータ導出部303から入力された、インター予測パラメータ(predFlagLX、refIdxLX、mvLX等)に基づいて、双予測モードの各参照ブロックが参照ピクチャの有効範囲外(OOB: Out-Of-Boundary)の領域を含むかどうかを判定する。

【0146】

なお、上記判定は、ピクチャもしくはサブピクチャの有効範囲外の領域を含むか否かを判定してもよい。また、スライス外の領域を含むか否かをを用いて判定してもよい。

【0147】

20

図9を参照して説明する。

【0148】

OOB判定部30921が参照ブロックの一部の画素がピクチャ外(有効範囲外)である、すなわち、参照ブロックがOOBであると判定した場合(isOOB==true、S1002)、OOBマスク導出部30922は、参照ブロックのどの領域が有効範囲外であることを示すマスクデータ(マスク値、OOBマスク)を導出する(S1004)。インター予測画像生成部309は、OOBである参照ブロックに対し、OOBマスクを適用して予測画像を生成する(S1006)。インター予測画像生成部309は、OOBではない参照ブロックに対し、通常の予測画像を生成する(S1008)。以下で詳細を説明する。

【0149】

30

(OOB判定)

【0150】

OOB判定部30921は、双予測において、2つ(第1および第2)の参照ブロックに対し、各参照ブロックの一部がピクチャ外(OOB)であるか否かを判定する。参照画像refLXに対応するLX参照ブロック($X=0$ または 1)の領域は、ブロック左上座標($xRefLX, yRefLX$)とブロックサイズ(幅および高さ: bW, bH)で表される。ピクチャの幅と高さをそれぞれpicW, picHとする。なお、以下では、mvLXの精度MVPRECに応じてmvLXを整数化した値を座標系に用いる。これらの値を用いて、OOB判定部30921は、次式によりLX参照ブロックがピクチャ外(有効範囲外)を含むか否かを表す真理値isOOB[X]を導出する。また有効範囲内の領域を有効領域と記す。

40

【0151】

図10に一例を示す。図10は整数単位の図であり、動きベクトルmvLX'、参照画素位置($xRefLX', yRefLX'$)は動きベクトルmvLX、参照画素位置($xRefLX, yRefLX$)を整数表現したパラメータである。図の例は $X=0$ であるが、 $X=1$ の場合も同様である。

【0152】

$$xRefLX = (xPb \ll \log_2 MVPREC) + mvLX[0] \quad (DecideOOB1)$$

【0153】

$$yRefLX = (yPb \ll \log_2 MVPREC) + mvLX[1]$$

【0154】

$$picMinX = 0 - MVPREC/2$$

50

【0155】

$\text{picMinY} = 0 - \text{MVPREC}/2$

【0156】

$\text{picMaxX} = ((\text{picW}-1) \ll (\log_2 \text{MVPREC}) + \text{MVPREC}/2$

【0157】

$\text{picMaxY} = ((\text{picH}-1) \ll (\log_2 \text{MVPREC}) + \text{MVPREC}/2$

【0158】

$\text{isOOB}[X] = (\text{xRefLX} \ll \text{picMinX}) \mid \mid (\text{yRefLX} \ll \text{picMinY}) \mid \mid$
 $(\text{xRefLX} + (\text{bW} \ll (\log_2 \text{MVPREC}) - 1) \gg \text{picMaxX}) \mid \mid$
 $(\text{yRefLX} + (\text{bH} \ll (\log_2 \text{MVPREC}) - 1) \gg \text{picMaxY})$

10

【0159】

$\log_2 \text{MVPREC}$ ビット左シフト (" $\ll \log_2 \text{MVPREC}$ ")を MVPREC との乗算 (" $* \text{MVPREC}$ ")で処理してもよいし、後述の" $* \text{MVPREC}$ "を" $\ll \log_2 \text{MVPREC}$ "で処理してもよい。

【0160】

あるいは、(DecideOOB1)における $\text{isOOB}[X]$ の導出式は下式であってもよい。

【0161】

$\text{isOOB}[X] = (\text{xRefLX} \ll \text{picMinX}) \mid \mid (\text{yRefLX} \ll \text{picMinY}) \mid \mid$
 $(\text{xRefLX} + (\text{bW} \ll (\log_2 \text{MVPREC}) - 1) \gg \text{picMaxX}) \mid \mid$
 $(\text{yRefLX} + (\text{bH} \ll (\log_2 \text{MVPREC}) - 1) \gg \text{picMaxY})$

20

【0162】

あるいは、OOB判定部30921は、丸めのための項を省き、次のように簡略に判定してもよい。

【0163】

$\text{picMaxX} = (\text{picW}-1) \ll (\log_2 \text{MVPREC})$ (DecideOOB2)

【0164】

$\text{picMaxY} = (\text{picH}-1) \ll (\log_2 \text{MVPREC})$

【0165】

$\text{isOOB}[X] = (\text{xRefLX} \ll 0) \mid \mid (\text{yRefLX} \ll 0) \mid \mid$
 $(\text{xRefLX} + (\text{bW} \ll (\log_2 \text{MVPREC}) - 1) \gg \text{picMaxX}) \mid \mid$
 $(\text{yRefLX} + (\text{bH} \ll (\log_2 \text{MVPREC}) - 1) \gg \text{picMaxY})$

30

【0166】

また、OOB判定部30921は、小数単位の境界座標ではなく整数単位の座標で判定を行ってもよい。

【0167】

$\text{xRefLX} = \text{xPb} + (\text{mvLX}[0] \gg \log_2 \text{MVPREC})$ (DecideOOB3)

【0168】

$\text{yRefLX} = \text{yPb} + (\text{mvLX}[1] \gg \log_2 \text{MVPREC})$

【0169】

$\text{isOOB}[X] = (\text{xRefLX} \ll 0) \mid \mid (\text{yRefLX} \ll 0) \mid \mid$
 $(\text{xRefLX} + \text{bW} - 1 \gg \text{picW}-1) \mid \mid$
 $(\text{yRefLX} + \text{bH} - 1 \gg \text{picH}-1)$

40

【0170】

あるいは、(DecideOOB3)における $\text{isOOB}[X]$ の導出式は下式であってもよい。

【0171】

$\text{isOOB}[X] = (\text{xRefLX} \ll 0) \mid \mid (\text{yRefLX} \ll 0) \mid \mid$
 $(\text{xRefLX} + \text{bW} - 1 \gg \text{picW}-1) \mid \mid$
 $(\text{yRefLX} + \text{bH} - 1 \gg \text{picH}-1)$

【0172】

上記によれば、動きベクトル $\text{mvLX}[]$ のMV精度による右シフトによって得られる整数

50

座標を用いるため、小数精度で判定するよりも処理量を省略できるという効果を奏する。

【0173】

OOB判定部30921は、ブロックサイズに応じてOOB処理を適用するか否かを切り替えてもよい。例えば、OOB判定部30921がisOOB[X]の導出に次式を用いて、幅および高さがともにBLKTHW1、BLKTHH1画素以上のブロックにのみisOOB=trueに設定して、OOB処理を適用してもよい。

【0174】

```
isOOB[X] = (bW >= BLKTHW1 && bH >= BLKTHH1) &&
            ((xRefLX < picMinX) || (yRefLX < picMinY) ||
             (xRefLX + bW*MVPREC - 1 >= picMaxX) ||
             (yRefLX + bH*MVPREC - 1 >= picMaxY))
```

【0175】

ここでOOB処理を適用するとは、isOOB[X]=trueとすることである。動き補償部3091（補間画像生成部3091）でいずれかの参照画像を重み付けせず優先して用いて、予測画像Pred[][]を導出することを意味する。

【0176】

例えば、BLKTHW1=4,8,16,32、BLKTHH1=4,8,16,32等であってもよい。OOBを適用するブロックサイズの閾値は8x8,4x4,4x8,8x4,16x16などでもよい。

【0177】

あるいは大きなブロックにおいてOOB処理を禁止してもよい。

【0178】

```
isOOB[X] = (bW <= BLKTHW2 && bH <= BLKTHH2) &&
            ((xRefLX < picMinX) || (yRefLX < picMinY) ||
             (xRefLX + bW*MVPREC - 1 >= picMaxX) ||
             (yRefLX + bH*MVPREC - 1 >= picMaxY))
```

【0179】

例えば、BLKTHW2=16,32,64、BLKTHH2=16,32,64等であってもよい。

【0180】

または特定の範囲の大きさのブロックにのみOOB処理が適用されるようにしてもよい。

【0181】

```
isOOB[X] = (bW > BLKTHW3 && bW <= BLKTHW4 && bH > BLKTHH3 &&
            bH <= BLKTHH4) &&
            ((xRefLX < picMinX) || (yRefLX < picMinY) ||
             (xRefLX + bW*MVPREC - 1 >= picMaxX) ||
             (yRefLX + bH*MVPREC - 1 >= picMaxY))
```

【0182】

例えば、(BLKTHW3, BLKTHW4)={(4,16),(8,32),(16,32),(32,64)}、(BLKTHH3, BLKTHH4)={(4,16),(8,32),(16,32),(32,64)}、等であってもよい。

【0183】

または面積の大小により判定してOOB処理が適用されるようにしてもよい。

【0184】

```
isOOB[X] = (bW*bH >= BLKTHW5*BLKTHH5) &&
            ((xRefLX < picMinX) || (yRefLX < picMinY) ||
             (xRefLX + bW*MVPREC - 1 >= picMaxX) ||
             (yRefLX + bH*MVPREC - 1 >= picMaxY))
```

【0185】

例えば、BLKTHW5=4,8,16,32、BLKTHH5=4,8,16,32等であってもよい。あるいはBLKTHW5*BLKTHH5の代わりにBLKTHS5を用いてもよい。例えば、BLKTHS5=32,64,128等であってもよい。

【0186】

10

20

30

40

50

上記の例では (DecideOOB1) の OOB 判定式をもとにブロックサイズを制限する条件を追加したが、(DecideOOB2) あるいは (DecideOOB3) の OOB 判定式をもとにブロックサイズを制限する条件を追加してもよい。

【0187】

上記の例では、isOOB[X]の導出式の条件の一部として (xRefLX < picMinX) || (yRefLX < picMinY) を用いたが、(xRefLX <= picMinX) || (yRefLX <= picMinY) を用いてもよい。

【0188】

以上のように、OOB 処理が適用されるブロックサイズを制限し、計算量を低減することができる。

【0189】

OOB 判定部 30921 は、isOOB[X] と isOOB[1-X] (X=0,1) の両方が 1 となることを避ける構成であってもよい。つまり参照ブロックの双方において、インター予測画像生成部 309 が OOB マスクを用いた予測画像生成をすることを禁止する。これにより、参照ブロックの一方は必ず isOOB[X]=false すなわち OOB 処理の適用外となり、後段で OOB マスクの作成およびマスクの適用を省略できる。したがって、計算量の削減が可能となる。

【0190】

<OOB 判定省略構成の例 1>

【0191】

OOB 判定部 30921 は、L0 参照領域と L1 参照領域の有効範囲の大きさに応じた判定により isOOB[X] (X=0,1) を導出してもよい。LX 参照ブロックのうちピクチャに含まれる有効面積 InsideAreaX がより大きい方を X=K (K=0,1) とする。有効面積が大きい参照ピクチャの OOB 判定を無効 (OOB 判定有無フラグ isOOB[K]=false) とし、有効面積が小さい他方 (isOOB[!K] あるいは isOOB[1-K]) の OOB 判定を有効 (isOOB[1-K]=true) とする。OOB 判定部 30921 は、たとえば次式により InsideAreaX を求める。なお、(false, true) の値として (0,1) を用いてもよいし、別の値 (1,0) を用いてもよい。LX 参照領域の有効範囲は、LX 参照ブロックとピクチャの共通部分で導出してもよい。

【0192】

xRefLX = (xPb < <log2MVPREC) + mvLX[0]

【0193】

yRefLX = (yPb < <log2MVPREC) + mvLX[1]

【0194】

InsideWidthX = (min(picMaxX, xRefLX + bW*MVPREC-1) - max(picMinX, xRefLX))

【0195】

InsideHeightX = (min(picMaxY, yRefLX + bH*MVPREC-1) - max(picMinY, yRefLX))

【0196】

InsideAreaX = (InsideWidthX > 0 && InsideHeightX > 0) ? InsideWidthX * InsideHeightX : 0

【0197】

InsideWidthX または InsideHeightX の値が負になる場合は、参照ブロックはピクチャに共通部分はない (参照ブロックはピクチャに含まれない) ため、InsideAreaX=0 としてもよい。

【0198】

OOB 判定部 30921 は、L0 参照ブロックと L1 参照ブロックに対してそれぞれ InsideArea0 と InsideArea1 を導出し、下記のように isOOB[X] を導出する。

【0199】

```
if (InsideArea0 > InsideArea1) {
    isOOB[0] = false, isOOB[1] = true
```

10

20

30

40

50

```

} else {
    isOOB[0] = true, isOOB[1] = false
}

```

【0200】

また下記のように一方のisOOB[X]の判定値を先に導出し、他方isOOB[Y] (Y=1-X)の判定値をisOOB[X]の判定値から導出してもよい。

【0201】

```
isOOB[0] = (InsideArea0 > InsideArea1) ? 0 : 1
```

【0202】

```
isOOB[1] = 1 - isOOB[0]
```

【0203】

上記条件式は、"InsideArea0 > InsideArea1"であってもよい。

【0204】

上記構成によれば、有効領域が大きい方の参照ピクチャにおけるOOB判定を無効（フラグisOOB[K]=false）にし、OOB判定を省略（OOB処理を適用しない）できるため、予測画像生成における計算量の削減が可能となる。特に、有効領域が大きい場合には、OOB判定を無効としても画質の低下がほぼない。

【0205】

<OOB判定省略構成の例2>

【0206】

OOB判定部30921は、2つの参照画像に関する有効領域InsideAreaのサイズが等しい場合には両方ともOOB判定を無効としてもよい。具体的には、OOB判定を省略することを示す値（false）となるようにisOOB[X]を導出してもよい。

【0207】

```

if (InsideArea0 > InsideArea1) {
    isOOB[0] = false, isOOB[1] = true
} else if (InsideArea0 < InsideArea1) {
    isOOB[0] = true, isOOB[1] = false
} else {
    isOOB[0] = false, isOOB[1] = false
}

```

【0208】

上記構成によれば、予測画像生成における計算量の削減が可能となる。

【0209】

<OOB判定省略構成の例3>

【0210】

OOB判定部30921は、LX参照領域の有効範囲（例えば、LX参照ブロックとピクチャの共通部分）の縦横の大きさを表すInsideWidthX, InsideHeightXの大小関係を用いて、OOB判定有無フラグisOOB[X]を導出してもよい。例えば、あるLX参照ブロックの有効領域の幅と高さの両方が他方のLY参照ブロック(Y=1-X)の幅と高さより大きい場合に、LX参照ブロックのOOB判定を無効（isOOB=false）としてもよい。

【0211】

```

if (InsideWidth0 > InsideWidth1 && InsideHeight0 > InsideHeight1) {
    isOOB[0] = false, isOOB[1] = true
} else {
    isOOB[0] = true, isOOB[1] = false
}

```

【0212】

または以下でもよい。

【0213】

10

20

30

40

50

```

if (InsideWidth0 >= InsideWidth1 && InsideHeight0 >= InsideHeight1) {
    isOOB[0] = false, isOOB[1] = true
} else {
    isOOB[0] = true, isOOB[1] = false
}

```

【0214】

あるいは以下でもよい。

【0215】

```

if (InsideWidth0 >= InsideWidth1 && InsideHeight0 >= InsideHeight1) {
    isOOB[0] = false, isOOB[1] = true
} else {
    isOOB[0] = true, isOOB[1] = false
}

```

10

【0216】

または、あるLX参照ブロックの有効領域の幅と高さの両方が他方のLY参照ブロック($Y=1-X$)の幅と高さより大きい場合、LX参照ブロックのOOB判定を無効 ($isOOB=false$) とし、2つの有効領域の幅もしくは高さが等しい場合に、OOB判定を両方とも無効としてもよい。

【0217】

```

if (InsideWidth0 >= InsideWidth1 && InsideHeight0 >= InsideHeight1) {
    isOOB[0] = false, isOOB[1] = true
} else if (InsideWidth0 < InsideWidth1 && InsideHeight0 < InsideHeight1) {
    isOOB[0] = true, isOOB[1] = false
} else {
    isOOB[0] = false, isOOB[1] = false
}

```

20

【0218】

または、上記判定を幅もしくは高さのみを用いてOOB判定を行ってもよい。

【0219】

```

if (InsideWidth0 >= InsideWidth1) {
    isOOB[0] = false, isOOB[1] = true
} else {
    isOOB[0] = true, isOOB[1] = false
}

```

30

【0220】

または、以下でもよい

【0221】

```

if (InsideHeight0 >= InsideHeight1) {
    isOOB[0] = false, isOOB[1] = true
} else {
    isOOB[0] = true, isOOB[1] = false
}

```

40

【0222】

上記構成によれば、有効領域が大きい参照ブロック（もしくは両方の参照ブロック）のOOB判定を省略できるため、予測画像生成における計算量の削減が可能となる。さらに、面積を導出する必要がなくなり乗算の計算量を削減できる。

【0223】

<まとめ>

【0224】

以上まとめると、L0参照領域の有効領域と、L1参照領域の有効領域を比べ、参照領域が大きい方の参照ピクチャのOOB判定を無効としてもよい。また、参照領域が大きいか

50

否かの判定を、L0とL1の有効領域の面積の比較により行ってもよい。また、参照領域が大きいかな否かの判定を、L0とL1の有効領域の幅もしくは高さの比較により行ってもよい。

【0225】

(OOBマスク導出処理)

【0226】

OOBマスク導出部30922は、参照ブロックに対してOOBマスク(OOBMask[X])を導出する。OOBMask[X]はbW×bH画素の二値の画像(マスク)であり、OOBMask[X][px][py](px=0..bW-1,py=0..bH-1)の値は、0(false)であれば画素(px,py)がピクチャ内、1(true)であればピクチャ外であることを示す。なお、falseとtrueは0,1に限定されず、別の値、例えば2進数でBitDepth以上の長さ0か1の列からなるビットマスク値"0000000000b","1111111111b"などであってもよい。また16進数の"000","3FF"や、"000","FFF","0000","FFFF"を用いてもよい。OOBマスク導出部30922は、OOB判定部30921がOOB処理対象ではない(isOOB[X]==false)と判定したブロックに対しては、OOBMask[X]の全ての値をfalseとする。また、OOB判定部30921がOOB処理対象である(isOOB[X]==true)と判定したブロックに対しては、次式を用いてマスクを導出する。なお、不等号(">","<")または"<="は等号を含まなく(">","<")でも構わない。

【0227】

OOBMask[X][px][py] = ((xRefLX + px < picMinX) || (yRefLX + py < picMinY) || (xRefLX + px > picMaxX) || (yRefLX + py > picMaxY)) ? 1 : 0

【0228】

ただしpx=0..bW-1,py=0..bH-1である。上記は(DecideOOB1)におけるisOOB[X]の判定式をもとにOOBMaskを導出した。これ以外にも、(DecideOOB2)、(DecideOOB3)におけるisOOB[X]の判定式をもとにOOBMaskを導出してよい。さらに、(DecideOOB1)、(DecideOOB2)、(DecideOOB3)に有効範囲の制約を加えたisOOB[X]の判定式をもとにOOBMaskを導出してよい。

【0229】

OOBマスク導出部30922は、OOBMask[X]をNxM画素単位のマスクとして導出してよい。すなわち、1つのマスク値をNxM画素のOOB判定に対応させる。このときOOBMask[X]の大きさは(bW/N)*(bH/M)である。各マスク値OOBMask[X][x][y](x=0..bW/N-1,y=0..bH/M-1)は、たとえば以下のように求められる。

【0230】

OOBMask[X][px][py] = ((xRefLX + px*N + N/2 <= picMinX) || (yRefLX + py*M + M/2 <= picMinY) || (xRefLX + px*N + N/2 > picMaxX) || (yRefLX + py*M + M/2 > picMaxY)) ? 1 : 0

【0231】

ここでpx=0..bW/N-1,py=0..bH/M-1である。N/2およびM/2の項は省略してもよい。上記は(DecideOOB1)におけるisOOB[X]の判定式をもとにOOBMaskを導出した。これ以外にも、(DecideOOB2)、(DecideOOB3)におけるisOOB[X]の判定式をもとにOOBMaskを導出してよい。さらに、(DecideOOB1)、(DecideOOB2)、(DecideOOB3)に有効範囲の制約を加えたisOOB[X]の判定式をもとにOOBMaskを導出してよい。

【0232】

これによりNxM画素単位でマスクが同一の値になることから、マスクに必要なメモリを削減するとともに計算量を削減できる。

【0233】

10

20

30

40

50

(OOB予測画像生成)

【0234】

動き補償部3091（補間画像生成部3091）は、双予測の場合には、上記の(式MC-1)で導出したPred[][]を、L0リスト、L1リスト毎に補間画像PredL0[][]とPredL1[][]として設定する。そして、PredL0[][]とPredL1[][]から補間画像Pred[][]を生成する。双予測の参照ブロックがOOB処理の対象である場合、予測画像の生成に当該参照ブロックに対応するOOBMask[X]が用いられる。このときPred[][]は次のように導出される。

【0235】

```
if (OOBMask[0][px][py] && !OOBMask[1][px][py]) { (式WP-1)
    Pred[px][py] = (PredL1[px][py] + offset1) >> shift1
} else if (!OOBMask[0][px][py] && OOBMask[1][px][py]) {
    Pred[px][py] = (PredL0[px][py] + offset1) >> shift1
} else {
    Pred[px][py] = (PredL0[px][py] + PredL1[px][py] + offset2) >> shift2
}
```

10

【0236】

ここでpx=0..bW-1, py=0..bH-1である。また、shift1=Max(2, 14-bitDepth)、shift2=Max(3, 15-bitDepth)、offset1=1<<(shift1-1)、offset2=1<<(shift2-1)である。なお、shift1は補間画像ビット深度（例えば14bit）を元のビット深度へ戻すためのシフト（例えば14-bitDepth）である。shift2は元のビットに戻すと同時に平均を兼ねた右シフト量、offset1、offset2はshift1、shift2だけ右シフトする際の丸め値である。

20

【0237】

なお、CU単位の明示的重み付き双予測（BCW）あるいはイントラインター結合予測CIIP（combined inter-picture merge and intra-picture prediction）を用いるか否かに応じて上記処理を切り替えてもよい。bcwIndexはBCWの重みを示すインデクスである。ciip_flagはCIIPのオンオフを示すフラグである。bcwIdx==0もしくはCIIPフラグ(ciip_flag)が1の場合には上記（式WP-1）の処理を行う。それ以外の場合には、以下の処理を行う。

30

【0238】

```
w1 = bcwWLut[bcwIdx]
```

【0239】

```
w0 = (8 - w1)
```

【0240】

ここでbcwWLut[k] = {4, 5, 3, 10, -2}

【0241】

```
if (OOBMask[0][px][py] && !OOBMask[1][px][py]) {
    Pred[px][py] = (PredL1[px][py] + offset1) >> shift1
} else if (!OOBMask[0][px][py] && OOBMask[1][px][py]) {
    Pred[px][py] = (PredL0[px][py] + offset1) >> shift1
} else {
```

40

【0242】

```
Pred[px][py] = ( w0 * PredL0[px][py] + w1 * PredL1[px][py] + offset3
)>>(shift1+3))
}
```

【0243】

ここでoffset3=1<<(shift1+2)である。

【0244】

ブロック単位でLX参照ピクチャを使うか否か示すフラグpredFlagLXと、OOBMask

50

による処理を一つの分岐にまとめて以下のように処理すると処理量を軽減できる効果を奏する。

```

【0245】
if ((predFlagL0 == 1 && predFlagL1 == 0) || (!OOBMask[0][px][py] && OOBMask[1][px][py])) { (式WP-2)
    Pred[px][py] = (PredL0[px][py]*we0 + offset1)>> shift1
} else if ((predFlagL0 == 0 && predFlagL1 == 1) || OOBMask[0][px][py] && !OOBMask[1][px][py]) {
    Pred[px][py] = (PredL1[px][py]*we1 + offset1)>> shift1
} else if ((bcwIdx == 0 || ciip_flag == 1) {
    Pred[px][py] = (w0 * PredL0[px][py] + w1 * PredL1[px][py] + offset2)
}>> shift2
} else { // bcwIdx != 0 and ciip_flag == 0
    Pred[px][py] = (we0 * PredL0[px][py] + we1 * PredL1[px][py] + offset3)>> (shift1+3)
}

```

10

【0246】

OOB判定部30921は、2枚の参照ピクチャ上の参照ブロックの重み付き和により予測画像を生成する重み付き予測では、ブロックにOOB処理を適用しなくてもよい。すなわち、重み付き予測が適用されているブロックはOOB判定有無フラグisOOB[X]をfalseとする。これにより、双方向の重み付き予測が適用される参照ブロックの一方が用いられないことによる歪の発生あるいは符号化効率の低下を防ぐことができる。このような重み付き予測には、BCW(Bi-prediction with CU-level Weight)予測とWP(Weighted prediction)予測がある。

20

【0247】

BCW予測は、ブロック単位で重み係数(gbw0、gbw1)を選択し、双予測におけるL0補間画像およびL1補間画像に重みを乗算して予測画像を生成する。パラメータ復号部302は、符号化ユニットCUで重み係数を示すパラメータbcw_idxを符号化データから復号する。パラメータ符号化部111はbcw_idxを符号化する。マージ予測の場合には、参照ブロック(隣接ブロック)のbcwIdxを対象ブロックのbcwIdxに設定する。それ以外の場合には、対象ブロックのbcwIdxにbcw_idxを設定する。bcwIdx!=0であれば当該CUにBCW予測を適用する。OOB判定部30921は、BCW予測を適用する場合(bcwIdx!=0)、isOOB[X]をfalseとする。

30

【0248】

WP予測は、スライス単位で参照ピクチャ毎の重み係数とオフセットを切り替え、CU単位で参照ピクチャインデックスをもとに重み係数とオフセットを選択し、L0補間画像およびL1補間画像に重みを乗算して予測画像を生成する。パラメータ復号部302は、PPSの符号化データから、BピクチャにWP予測を適用するか否かを示すpps_weighted_bipred_flagを復号する。pps_weighted_bipred_flag==1の場合、当該PPSのBピクチャにWP予測を適用する。OOB判定部30921は、明示的なWP予測を適用する場合(pps_weighted_bipred_flag==1)、isOOB[X]をfalseとする。なお、参照ピクチャを2枚使わない場合(predFlagLX(X=0,1)のいずれかが0)は、OOB処理を適用してもよい。

40

【0249】

<省略構成>

【0250】

上述の省略構成を利用する場合は、OOBMask[0]、OOBMask[1]ともに1となることはないので、動き補償部3091(補間画像生成部3091)は、以下の分岐処理により補間画像を導出することができる。

【0251】

50

```

if (OOBMask[0][px][py]) {
    Pred[px][py] = (PredL1[px][py] + offset1) >> shift1
} else if (OOBMask[1][px][py]) {
    Pred[px][py] = (PredL0[px][py] + offset1) >> shift1
} else {
    Pred[px][py] = (PredL0[px][py] + PredL1[px][py] + offset2) >> shift2
}

```

【0252】

また、予めPredL0、PredL1、PredBIを算出しておいて、以下のマスク処理で最終的なPredを導出することが適当である。

【0253】

まず、PredL0、PredL1、PredBIを算出する。

【0254】

```
PredL0[px][py] = (PredL0[px][py] + offset1) >> shift1
```

【0255】

```
PredL1[px][py] = (PredL1[px][py] + offset1) >> shift1
```

【0256】

```
PredBI[px][py] = (PredL0[px][py] + PredL1[px][py] + offset2) >> shift2
```

【0257】

次に、OOBMask[0][px][py]、OOBMask[1][px][py]の値として、2進数でBitDepth長の"111111111b"、"000000000b"のビットマスクを用いる場合、以下の処理で最終的なPredを導出する。

【0258】

```

Pred[px][py]=
    (PredBI[px][py] & (!OOBMask[0][px][py] & !OOBMask[1][px][py])) +
    (PredL0[px][py] & OOBMask[0][px][py]) +
    (PredL1[px][py] & OOBMask[1][px][py])

```

【0259】

なお、true=1, false=0の値である場合には、0, 1からビットマスク値"000000000b", "111111111b"を生成して以下の処理でもよい。ここで"+"は"|"でもよい（以下のマスク演算処理でも同様）

【0260】

```

Pred[px][py]= (式excl-1)
    (PredBI[px][py] & (0-(!OOBMask[0][px][py] & !OOBMask[1][px][py]) ?
1 : 0)))+
    (PredL0[px][py] & (0-(OOBMask[0][px][py] ? 1 : 0))) +
    (PredL1[px][py] & (0-(OOBMask[1][px][py] ? 1 : 0)))

```

【0261】

上記の式は、画素値のビット深度以上のビット幅を持つマスクを作成するために、二値(0または1)のマスクを0(全てのビットが0,画素値が無効)または-1(全てのビットが1,画素値が有効)に変換する計算を含んでいる。予測画像の導出はこれに限らず、PredBI, PredL0, PredL1に対応する3項のうちOOBマスクにより無効となる項が0となる計算を用いればよい。たとえば画素値のビット深度を最大16ビットとするとき、次式を用いてもよい。

【0262】

```

Pred[px][py]=
    (PredBI[px][py] & ((!OOBMask[0][px][py] & !OOBMask[1][px][py])? 0xFF
FFF : 0))+
    (PredL0[px][py] & (OOBMask[0][px][py] ? 0xFFFF : 0)) +
    (PredL1[px][py] & (OOBMask[1][px][py] ? 0xFFFF : 0))

```

10

20

30

40

50

【0263】

3項の結合に加算を用いず、下記のようにビットORを用いてもよい。

【0264】

```
Pred[px][py]=
  (PredBl[px][py] & ((!OOBMask[0][px][py] & !OOBMask[1][px][py])? 0xFF
  FFF : 0)) |
  (PredL0[px][py] & (OOBMask[0][px][py] ? 0xFFFF : 0)) |
  (PredL1[px][py] & (OOBMask[1][px][py] ? 0xFFFF : 0))
```

【0265】

あるいは乗算を用いてもよい。

10

【0266】

```
Pred[px][py]=
  (PredBl[px][py] * ((!OOBMask[0][px][py] & !OOBMask[1][px][py])? 1 :
  0)) +
  (PredL0[px][py] * (OOBMask[0][px][py] ? 1 : 0)) +
  (PredL1[px][py] * (OOBMask[1][px][py] ? 1 : 0))
```

【0267】

上記によれば排他処理を簡略化できる。排他処理により、(式excl-1)の演算よりも処理を簡略化できる。

20

【0268】

動き補償部3091(補間画像生成部3091)は、isOOBの値を参照して予め処理を分けてもよい。たとえば以下のようにしてもよい。このようにすれば、isOOB[X]がfalseの場合に画素毎にOOBMask[X][px][py]を参照する必要がなく、計算量を削減できるという効果を奏する。

【0269】

```
if (isOOB[0] && !isOOB[1]) {
  if (OOBMask[0][px][py]) {
    Pred[px][py] = (PredL1[px][py] + offset1)>> shift2
  } else {
    Pred[px][py] = (PredL0[px][py] + PredL1[px][py] + offset2)>> shift
  }
} else if (!isOOB[0] && isOOB[1]) {
  if (OOBMask[1][px][py]) {
    Pred[px][py] = (PredL0[px][py] + offset1)>> shift2
  } else {
    Pred[px][py] = (PredL0[px][py] + PredL1[px][py] + offset2)>> shift
  }
} else if (isOOB[0] && isOOB[1]) {
  if (OOBMask[0][px][py] && !OOBMask[1][px][py]) {
    Pred[px][py] = (PredL1[px][py] + offset1)>> shift2
  } else if (!OOBMask[0][px][py] && OOBMask[1][px][py]) {
    Pred[px][py] = (PredL0[px][py] + offset1)>> shift2
  } else {
    Pred[px][py] = (PredL0[px][py] + PredL1[px][py] + offset2)>> shift
  }
} else {
  Pred[px][py] = (PredL0[px][py] + PredL1[px][py] + offset2)>> shift
}
```

30

40

【0270】

(重み予測)

50

【0271】

重み予測部3094は、補間画像PredLXに重み係数を乗算することによりブロックの予測画像を生成する。

【0272】

インター予測画像生成部309は生成したブロックの予測画像を加算部312に出力する。

【0273】

(動画像符号化装置の構成)

【0274】

次に、本実施形態に係る動画像符号化装置11の構成について説明する。図7は、本実施形態に係る動画像符号化装置11の構成を示すブロック図である。動画像符号化装置11は、予測画像生成部101、減算部102、変換・量子化部103、逆量子化・逆変換部105、加算部106、ループフィルタ107、予測パラメータメモリ(予測パラメータ記憶部、フレームメモリ)108、参照ピクチャメモリ(参照画像記憶部、フレームメモリ)109、符号化パラメータ決定部110、パラメータ符号化部111、予測パラメータ導出部120、エントロピー符号化部104を含んで構成される。

10

【0275】

予測画像生成部101はCU毎に予測画像を生成する。予測画像生成部101は既に説明したインター予測画像生成部309とイントラ予測画像生成部を含んでおり、説明を省略する。

【0276】

減算部102は、予測画像生成部101から入力されたブロックの予測画像の画素値を、画像Tの画素値から減算して予測誤差を生成する。減算部102は予測誤差を変換・量子化部103に出力する。

20

【0277】

変換・量子化部103は、減算部102から入力された予測誤差に対し、周波数変換によって変換係数を算出し、量子化によって量子化変換係数を導出する。変換・量子化部103は、量子化変換係数をパラメータ符号化部111及び逆量子化・逆変換部105に出力する。

【0278】

変換・量子化部103は、分離変換部(第1の変換部)と、非分離変換部(第2の変換部)と、スケーリング部と、を備えている。

30

【0279】

分離変換部は、予測誤差に分離変換を適用する。スケーリング部は、変換係数に対して、量子化マトリックスによるスケーリングを行う。

【0280】

逆量子化・逆変換部105は、動画像復号装置31における逆量子化・逆変換部311と同じであり、説明を省略する。算出した予測誤差は加算部106に出力される。

【0281】

パラメータ符号化部111は、ヘッダ符号化部1110、CT情報符号化部1111、CU符号化部1112(予測モード符号化部)を備えている。CU符号化部1112はさらにTU符号化部1114を備えている。以下、各モジュールの概略動作を説明する。

40

【0282】

ヘッダ符号化部1110はヘッダ情報、分割情報、予測情報、量子化変換係数等のパラメータの符号化処理を行う。

【0283】

CT情報符号化部1111は、QT、MT(BT、TT)分割情報等を符号化する。

【0284】

CU符号化部1112はCU情報、予測情報、分割情報等を符号化する。

【0285】

TU符号化部1114は、TUに予測誤差が含まれている場合に、QP更新情報と量子化予

50

測誤差を符号化する。

【0286】

CT情報符号化部1111、CU符号化部1112は、インター予測パラメータ（predMode、general_merge_flag、merge_idx、inter_pred_idc、refIdxLX、mvp_LX_idx、mvdLX）、イントラ予測パラメータ、量子化変換係数等のシンタックス要素をパラメータ符号化部111に供給する。

【0287】

エントロピー符号化部104には、パラメータ符号化部111から量子化変換係数と符号化パラメータ（分割情報、予測パラメータ）が入力される。エントロピー符号化部104はこれらをエントロピー符号化して符号化ストリームTeを生成し、出力する。

10

【0288】

予測パラメータ導出部120は、インター予測パラメータ符号化部112、イントラ予測パラメータ符号化部を含む手段であり、符号化パラメータ決定部110から入力されたパラメータからイントラ予測パラメータ及びインター予測パラメータを導出する。導出されたイントラ予測パラメータ及びインター予測パラメータは、パラメータ符号化部111に出力される。

【0289】

（インター予測パラメータ符号化部の構成）

【0290】

インター予測パラメータ符号化部112は、パラメータ符号化制御部1121、インター予測パラメータ導出部303を含んで構成される。インター予測パラメータ導出部303は動画像復号装置と共通の構成である。パラメータ符号化制御部1121は、マージインデックス導出部11211とベクトル候補インデックス導出部11212を含む。

20

【0291】

マージインデックス導出部11211は、マージ候補等を導出し、インター予測パラメータ導出部303に出力する。ベクトル候補インデックス導出部11212は予測ベクトル候補等を導出し、インター予測パラメータ導出部303とパラメータ符号化部111に出力する。

【0292】

（イントラ予測パラメータ符号化部の構成）

【0293】

イントラ予測パラメータ符号化部は、パラメータ符号化制御部とイントラ予測パラメータ導出部を備える。イントラ予測パラメータ導出部は動画像復号装置と共通の構成である。

30

【0294】

ただし、動画像復号装置と異なり、インター予測パラメータ導出部303、イントラ予測パラメータ導出部への入力には符号化パラメータ決定部110、予測パラメータメモリ108であり、パラメータ符号化部111に出力する。

【0295】

加算部106は、予測画像生成部101から入力された予測ブロックの画素値と逆量子化・逆変換部105から入力された予測誤差を画素毎に加算して復号画像を生成する。加算部106は生成した復号画像を参照ピクチャメモリ109に記憶する。

40

【0296】

ループフィルタ107は加算部106が生成した復号画像に対し、デブロッキングフィルタ、SAO、ALFを施す。なお、ループフィルタ107は、必ずしも上記3種類のフィルタを含まなくてもよく、例えばデブロッキングフィルタのみの構成であってもよい。

【0297】

予測パラメータメモリ108は、符号化パラメータ決定部110が生成した予測パラメータを、対象ピクチャ及びCU毎に予め定めた位置に記憶する。

【0298】

参照ピクチャメモリ109は、ループフィルタ107が生成した復号画像を対象ピクチャ

50

及びCU毎に予め定めた位置に記憶する。

【0299】

符号化パラメータ決定部110は、符号化パラメータの複数のセットのうち、1つのセットを選択する。符号化パラメータとは、上述したQT、BTあるいはTT分割情報、予測パラメータ、あるいはこれらに関連して生成される符号化の対象となるパラメータである。予測画像生成部101は、これらの符号化パラメータを用いて予測画像を生成する。

【0300】

符号化パラメータ決定部110は、複数のセットの各々について情報量の大きさと符号化誤差を示すRDコスト値を算出する。RDコスト値は、例えば、符号量と二乗誤差に係数 λ を乗じた値との和である。符号量は、量子化誤差と符号化パラメータをエントロピー符号化して得られる符号化ストリームTeの情報量である。二乗誤差は、減算部102において算出された予測誤差の二乗和である。係数 λ は、予め設定されたゼロよりも大きい実数である。符号化パラメータ決定部110は、算出したコスト値が最小となる符号化パラメータのセットを選択する。符号化パラメータ決定部110は決定した符号化パラメータをパラメータ符号化部111と予測パラメータ導出部120に出力する。

【0301】

なお、上述した実施形態における動画像符号化装置11、動画像復号装置31の一部、例えば、エントロピー復号部301、パラメータ復号部302、ループフィルタ305、予測画像生成部308、逆量子化・逆変換部311、加算部312、予測パラメータ導出部320、予測画像生成部101、減算部102、変換・量子化部103、エントロピー符号化部104、逆量子化・逆変換部105、ループフィルタ107、符号化パラメータ決定部110、パラメータ符号化部111、予測パラメータ導出部120をコンピュータで実現するようにしても良い。その場合、この制御機能を実現するためのプログラムをコンピュータ読み取り可能な記録媒体に記録して、この記録媒体に記録されたプログラムをコンピュータシステムに読み込ませ、実行することによって実現しても良い。なお、ここでいう「コンピュータシステム」とは、動画像符号化装置11、動画像復号装置31のいずれかに内蔵されたコンピュータシステムであって、OSや周辺機器等のハードウェアを含むものとする。また、「コンピュータ読み取り可能な記録媒体」とは、フレキシブルディスク、光磁気ディスク、ROM、CD-ROM等の可搬媒体、コンピュータシステムに内蔵されるハードディスク等の記憶装置のことをいう。さらに「コンピュータ読み取り可能な記録媒体」とは、インターネット等のネットワークや電話回線等の通信回線を介してプログラムを送信する場合の通信線のように、短時間、動的にプログラムを保持するもの、その場合のサーバやクライアントとなるコンピュータシステム内部の揮発性メモリのように、一定時間プログラムを保持しているものも含んでも良い。また上記プログラムは、前述した機能の一部を実現するためのものであっても良く、さらに前述した機能をコンピュータシステムにすでに記録されているプログラムとの組み合わせで実現できるものであっても良い。

【0302】

また、上述した実施形態における動画像符号化装置11、動画像復号装置31の一部、または全部を、LSI (Large Scale Integration) 等の集積回路として実現しても良い。動画像符号化装置11、動画像復号装置31の各機能ブロックは個別にプロセッサ化しても良いし、一部、または全部を集積してプロセッサ化しても良い。また、集積回路化の手法はLSIに限らず専用回路、または汎用プロセッサで実現しても良い。また、半導体技術の進歩によりLSIに代替する集積回路化の技術が出現した場合、当該技術による集積回路を用いても良い。

【0303】

以上、図面を参照してこの発明の一実施形態について詳しく説明してきたが、具体的な構成は上述のものに限られることはなく、この発明の要旨を逸脱しない範囲内において様々な設計変更等を行うことが可能である。

【0304】

本発明の実施形態は上述した実施形態に限定されるものではなく、請求項に示した範囲

10

20

30

40

50

で種々の変更が可能である。すなわち、請求項に示した範囲で適宜変更した技術的手段を組み合わせ得られる実施形態についても本発明の技術的範囲に含まれる。

【産業上の利用可能性】

【0305】

本発明の実施形態は、画像データが符号化された符号化データを復号する動画像復号装置、および、画像データが符号化された符号化データを生成する動画像符号化装置に好適に適用することができる。また、動画像符号化装置によって生成され、動画像復号装置によって参照される符号化データのデータ構造に好適に適用することができる。

【符号の説明】

【0306】

10

31 動画像復号装置

301 エントロピー復号部

302 パラメータ復号部

3022 CU復号部

3024 TU復号部

303 インター予測パラメータ導出部

305、107 ループフィルタ

306、109 参照ピクチャメモリ

307、108 予測パラメータメモリ

308、101 予測画像生成部

20

309 インター予測画像生成部

3092 OOB処理部

30921 OOB判定部

30922 OOBマスク導出部

311、105 逆量子化・逆変換部

312、106 加算部

320 予測パラメータ導出部

11 動画像符号化装置

102 減算部

103 変換・量子化部

30

104 エントロピー符号化部

110 符号化パラメータ決定部

111 パラメータ符号化部

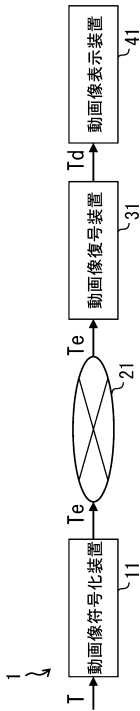
112 インター予測パラメータ符号化部

120 予測パラメータ導出部

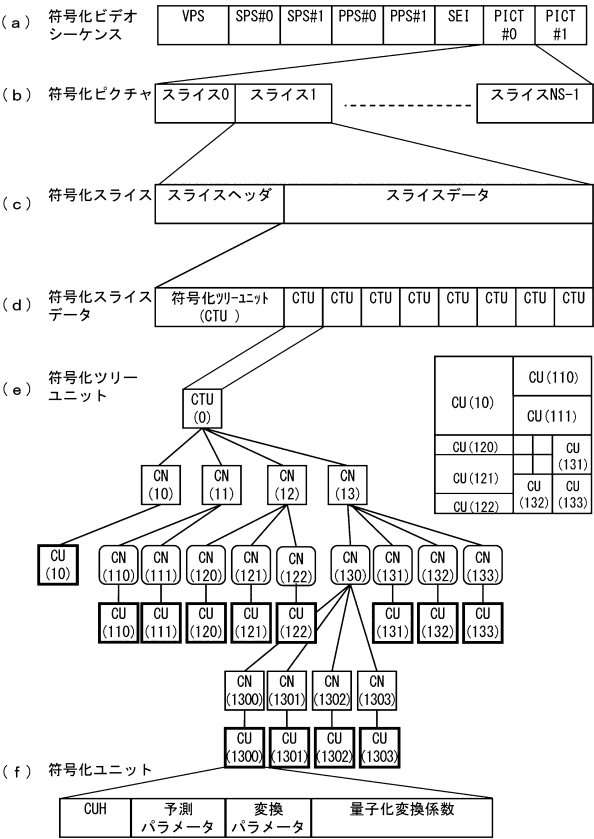
40

50

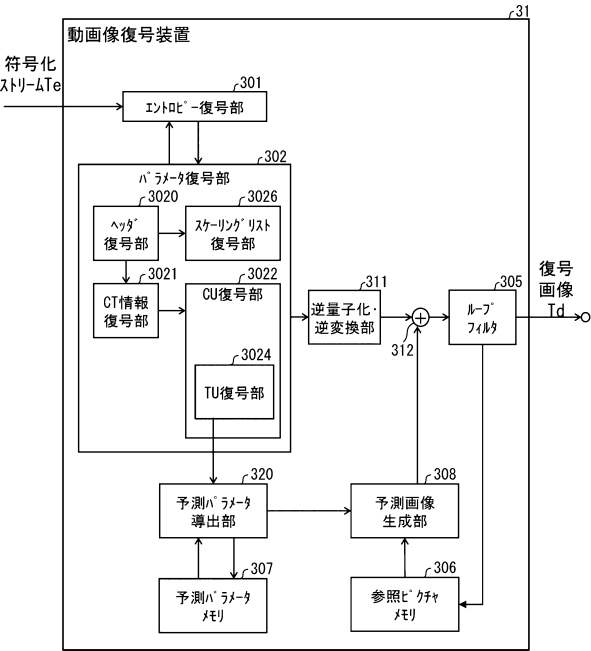
【図面】
【図 1】



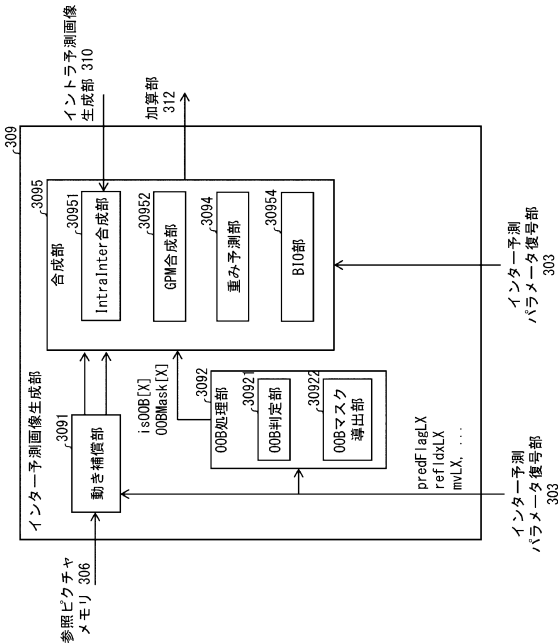
【図 2】



【図 3】



【図 4】



10

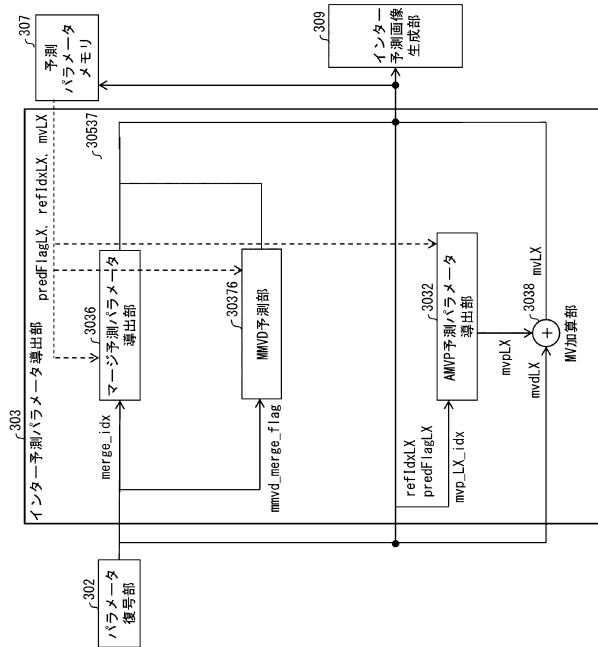
20

30

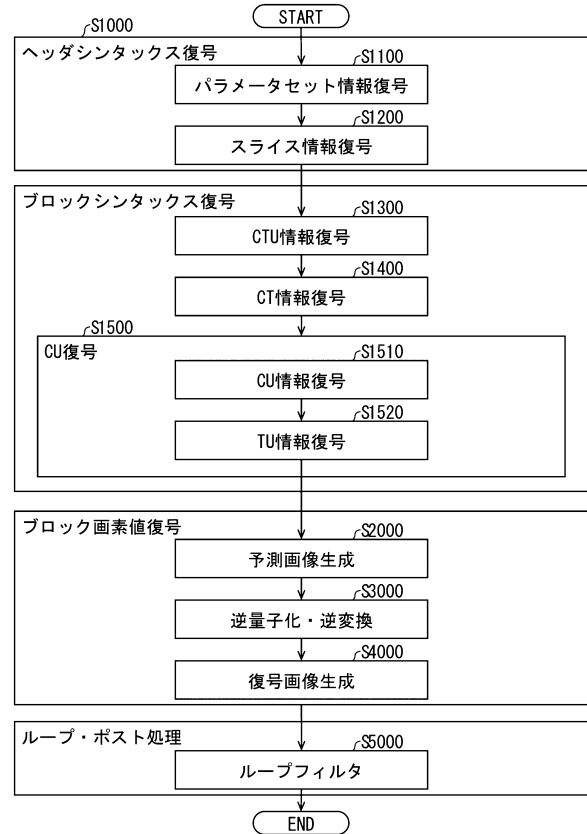
40

50

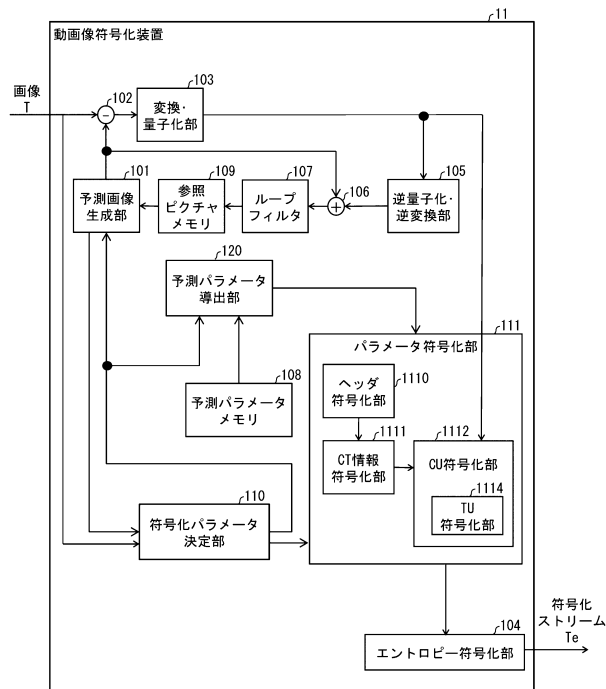
【図 5】



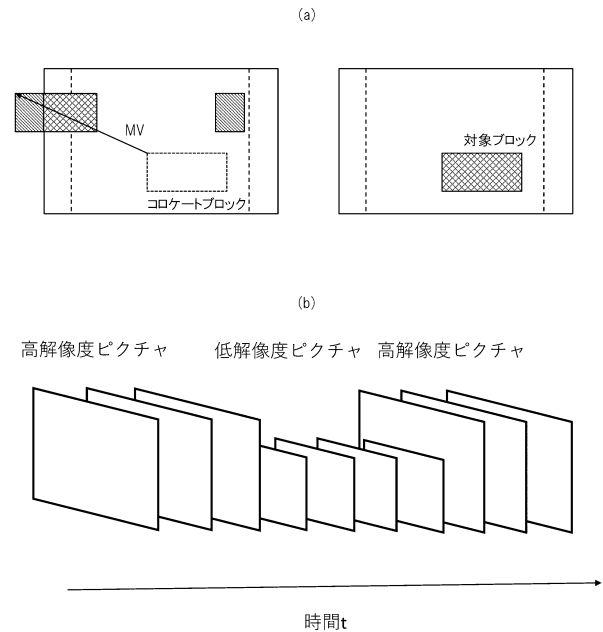
【図 6】



【図 7】



【図 8】



10

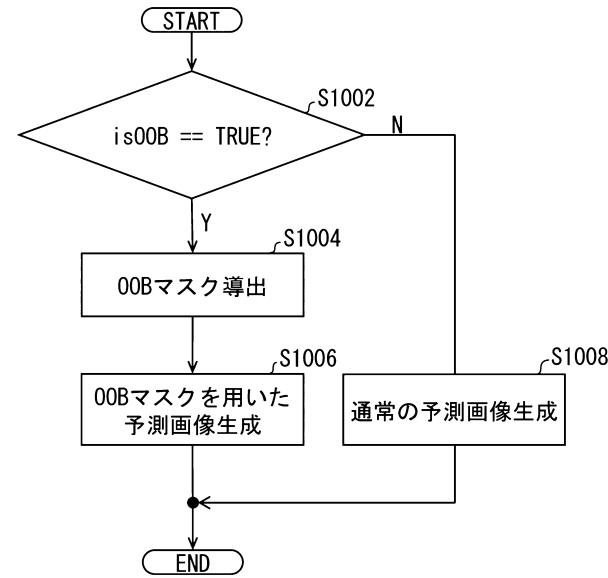
20

30

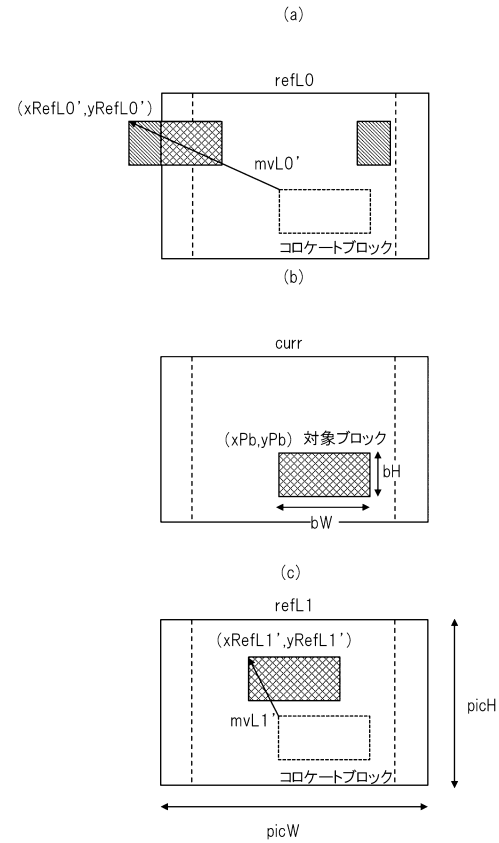
40

50

【図 9】



【図 10】



10

20

30

40

50

フロントページの続き

大阪府堺市堺区匠町 1 番地 シャープ株式会社内

(72)発明者 青野 友子

大阪府堺市堺区匠町 1 番地 シャープ株式会社内

F ターム (参考) 5C159 MA04 MA05 MC11 ME01 NN04 PP04 UA02 UA05 UA16