

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7414975号
(P7414975)

(45)発行日 令和6年1月16日(2024.1.16)

(24)登録日 令和6年1月5日(2024.1.5)

(51)国際特許分類		F I	
H O 4 N	19/119 (2014.01)	H O 4 N	19/119
H O 4 N	19/14 (2014.01)	H O 4 N	19/14
H O 4 N	19/176 (2014.01)	H O 4 N	19/176
H O 4 N	19/70 (2014.01)	H O 4 N	19/70
請求項の数 13 (全76頁)			
(21)出願番号	特願2022-520497(P2022-520497)	(73)特許権者	503433420
(86)(22)出願日	令和2年9月28日(2020.9.28)		華為技術有限公司
(65)公表番号	特表2022-551091(P2022-551091 A)		HUAWEI TECHNOLOGIES CO., LTD.
(43)公表日	令和4年12月7日(2022.12.7)		中華人民共和國 5 1 8 1 2 9 広東省深
(86)国際出願番号	PCT/CN2020/118389		▲チェン▼市龍崗区坂田 華為總部▲ベ
(87)国際公開番号	WO2021/063308		ン▼公樓
(87)国際公開日	令和3年4月8日(2021.4.8)		Huawei Administrat
審査請求日	令和4年5月11日(2022.5.11)		ion Building, Banti
(31)優先権主張番号	PCT/EP2019/076805		an, Longgang Distri
(32)優先日	令和1年10月3日(2019.10.3)		ct, Shenzhen, Guang
(33)優先権主張国・地域又は機関	欧州特許庁(EP)	(74)代理人	dong 5 1 8 1 2 9, P. R. C
			hina
			100107766
			弁理士 伊東 忠重
			最終頁に続く

(54)【発明の名称】 幾何学的パーティション・モードのためのコーディング・プロセス

(57)【特許請求の範囲】

【請求項1】

復号デバイスによって実装されるコーディングの方法であって、
現在のコーディング・ブロックに対する分裂モード・インデックス値を取得することと、
前記分裂モード・インデックス値、及び前記分裂モード・インデックス値に基づいて角
度インデックス値 $angleIdx$ を指定するテーブルに従って、前記現在のコーディ
ング・ブロックに対する前記角度インデックス値 $angleIdx$ を取得することと、
前記角度インデックス値 $angleIdx$ に従って、インデックス値 $partIdx$ を
セットすることであって、前記インデックス値 $partIdx$ は、

【数56】

$$partIdx = (angleIdx \geq threshold1 \ \&\& \ angleIdx \leq threshold2) ? 1 : 0,$$

を満たし、 $threshold1$ 及び $threshold2$ は整数値であり、 $threshold1$ は $threshold2$ より小さい、ことと、

前記インデックス値 $partIdx$ に従って、前記現在のコーディング・ブロックを復
号することと、を含む、方法。

【請求項2】

$threshold1$ が13であり、 $threshold2$ が27である、請求項1に
記載の方法。

【請求項 3】

前記コーディング・ブロックを復号することは、前記インデックス値 `partIdx` に従って、現在のブロックのためのモーション情報を記憶することを含む、請求項 1 ～ 2 のいずれか一項に記載の方法。

【請求項 4】

前記分裂モード・インデックス値は、どの幾何学的パーティション・モードが前記現在のコーディング・ブロックに使用されるかを示すために使用される、請求項 1 ～ 3 のいずれか一項に記載の方法。

【請求項 5】

前記角度インデックス値 `angleIdx` は、前記現在のコーディング・ブロックの幾何学的パーティションに使用される、請求項 1 ～ 4 のいずれか一項に記載の方法。

10

【請求項 6】

ビデオ復号器であって、

現在のコーディング・ブロックに対する分裂モード・インデックス値を取得するように構成されている解析ユニットと、

前記分裂モード・インデックス値、及び前記分裂モード・インデックス値に基づいて角度インデックス値 `angleIdx` を指定するテーブルに従って、前記現在のコーディング・ブロックに対する前記角度インデックス値 `angleIdx` を取得するように構成されている角度インデックス値取得ユニットと、

前記角度インデックス値 `angleIdx` に従って、インデックス値 `partIdx` をセットするように構成されているセッティング・ユニットであって、前記インデックス値 `partIdx` は、

20

【数 57】

$$\text{partIdx} = (\text{angleIdx} \geq \text{threshold1} \ \&\& \ \text{angleIdx} \leq \text{threshold2}) ? 1 : 0,$$

を満たし、`threshold1` 及び `threshold2` は、整数値であり、`threshold1` は、`threshold2` より小さい、セッティング・ユニットと、

前記インデックス値 `partIdx` に従って、前記現在のコーディング・ブロックを復号するように構成されている処理ユニットと、含む、ビデオ復号器。

30

【請求項 7】

`threshold1` が 13 であり、`threshold2` が 27 である、請求項 6 に記載の復号器。

【請求項 8】

前記処理ユニットは、前記インデックス値 `partIdx` に従って、現在のブロックのためのモーション情報を記憶するように構成されている、請求項 6 ～ 7 のいずれか一項に記載の復号器。

【請求項 9】

前記分裂モード・インデックス値は、どの幾何学的パーティション・モードが前記現在のコーディング・ブロックに使用されるかを示すために使用される、請求項 6 ～ 8 のいずれか一項に記載の復号器。

40

【請求項 10】

前記角度インデックス値 `angleIdx` は、前記現在のコーディング・ブロックの幾何学的パーティションに使用される、請求項 6 ～ 9 のいずれか一項に記載の復号器。

【請求項 11】

請求項 1 ～ 5 のいずれか一項に記載の方法を実施するための処理回路を含む、復号器。

【請求項 12】

プログラム・コードを含み、コンピュータ可読媒体に記憶されたコンピュータ・プログラムであって、前記プログラム・コードは、コンピュータにより実行されるときに、前記コンピュータに請求項 1 ～ 5 のいずれか一項に記載の方法を行わせる、コンピュータ・プ

50

ログラム。

【請求項 1 3】

復号器であって、

1つ以上のプロセッサと、

前記1つ以上のプロセッサに結合され、前記1つ以上のプロセッサによる実行のための命令を記憶する非一時的なコンピュータ可読記憶媒体と、を含み、前記命令は、前記1つ以上のプロセッサによって実行されるときに、請求項1～5のいずれか一項に記載の方法を実施するように前記復号器を設定する、復号器。

【発明の詳細な説明】

【技術分野】

10

【0001】

関連出願の相互参照

本特許出願は、2019年10月3日に出願された国際特許出願第PCT/EP2019/076805号に対する優先権を主張する。前述の特許出願の開示は、その全体が参照により本明細書に組み込まれる。

【0002】

本出願（開示）の実施形態は、一般に、ピクチャ処理の分野に関係し、より詳細には、幾何学的パーティションに関係する。

【背景技術】

【0003】

20

ビデオ・コーディング（ビデオ符号化及び復号）は、広範囲のデジタル・ビデオ・アプリケーション、例えば、放送デジタルTV、インターネット及びモバイル・ネットワークを介したビデオ伝送、ビデオチャット、ビデオ会議などのリアルタイムの会話アプリケーション、DVD及びブルーレイ・ディスク、ビデオ・コンテンツの収集及び編集システム、並びにセキュリティ・アプリケーションのカムコーダに使用される。

【0004】

比較的短いビデオでさえ描写するために必要とされるビデオ・データの量は、相当なものであり、これは、データがストリーム化されるか、又は他の方法で限定された帯域幅容量を有する通信ネットワークを介して通信されるときに、困難になることがある。したがって、ビデオ・データは、一般に、現代の電気通信ネットワークを介して通信される前に圧縮される。また、メモリ・リソースが制限されることがあるため、ビデオが記憶デバイスに記憶される場合に、ビデオのサイズも問題となる可能性がある。ビデオ圧縮デバイスは、しばしば、伝送又は記憶の前にビデオ・データをコーディングするためにソースにおいてソフトウェア及び／又はハードウェアを使用し、それによってデジタル・ビデオ画像を表すのに必要なデータ量を減少させる。次いで、圧縮されたデータは、ビデオ・データを復号するビデオ解凍デバイスによって宛先で受信される。ネットワークリソースが限られており、より高いビデオ品質の要求が絶えず増加しているため、ピクチャ品質にほとんど犠牲を払わずに圧縮比を改善する改善された圧縮及び解凍技術が望ましい。

30

【発明の概要】

【0005】

40

本出願の実施形態は、独立請求項にしたがって符号化及び復号するための装置及び方法を提供する。

【0006】

前述の目的及び他の目的は、独立請求項の主題事項によって達成される。さらなる実装は、従属請求項、明細書及び図面から明らかである。

【0007】

本発明の第1の態様は、復号デバイスによって実装されるコーディングの方法であって、現在のコーディング・ブロックに対する分裂モード・インデックス値を取得することと、分裂モード・インデックス値、及び分裂モード・インデックス値に基づいて角度インデックス値 $angleIdx$ を指定するテーブルに従って、現在のコーディング・ブロック

50

に対する角度インデックス値 `angleIdx` を取得することと、角度インデックス値 `angleIdx` に従って、インデックス値 `partIdx` をセットすることと、インデックス値 `partIdx` に従って、現在のコーディング・ブロックを復号することと、を含む、方法を提供する。

【0008】

本発明の実施形態によれば、コーディング・ブロックは、インデックス値に従って復号される。復号プロセスは、サンプル重み導出プロセス、モーション情報記憶プロセス、モーション・ベクトル導出プロセスなどであってもよい。したがって、バッファ利用と復号効率改善されている。

【0009】

一実装形式では、分裂モード・インデックス値は、現在のコーディング・ブロックにどの幾何学的パーティション・モードが使用されるかを示すために使用される。例えば、`geo_partition_idx` 又は `merge_gpm_partition_idx` である。

【0010】

一例では、`merge_gpm_partition_idx[x0][y0]` 又は `geo_partition_idx` は、幾何学的パーティショニング・マージ・モードのパーティショニング形状を指定する。配列インデックス `x0`、`y0` は、ピクチャの左上のルマ・サンプルに対する、考慮されるコーディング・ブロックの左上のルマ・サンプルの位置 (`x0`, `y0`) を指定する。

【0011】

`merge_gpm_partition_idx[x0][y0]` 又は `geo_partition_idx[x0][y0]` が存在しないときに、それは、0 に等しいと推論される。

【0012】

一例では、分裂モード・インデックス値は、ビデオ・ビットストリームにおいてコーディングされたインデックス値を解析することによって取得されてもよいし、分裂モード・インデックス値は、ビデオ・ビットストリームから解析される構文値に従って判定されてもよい。

【0013】

ビットストリームは、無線ネットワーク又は有線ネットワークに従って取得されてもよい。ビットストリームは、同軸ケーブル、光ファイバケーブル、ツイスト・ペア、デジタル加入者線、又は赤外線、ラジオ、マイクロ波、WIFI、Bluetooth、LTE 若しくは 5G の無線技術を使用して、ウェブサイト、サーバ、又は他のリモート・ソースから送信されてもよい。

【0014】

一実施形態では、ビットストリームは、例えば、1 つ以上のコーディングされたビデオ・シーケンス (CVS) を形成するアクセス・ユニット (AU) のシーケンスの表現を形成するネットワーク抽象化レイヤ (NAL) ユニット・ストリーム又はバイト・ストリームの形態のビットのシーケンスである。

【0015】

いくつかの実施形態では、復号プロセスの場合、復号器側がビットストリームを読み、ビットストリームから復号されたピクチャを導出し、符号化プロセスの場合、符号化器側がビットストリームを生成する。

【0016】

通常、ビットストリームは、構文構造によって形成される構文要素を含むだろう。

構文要素：ビットストリームで表現されるデータの要素

構文構造：指定された順序でビットストリーム内に一緒に存在する 0 個以上の構文要素

【0017】

特定の例では、ビットストリームフォーマットは、ネットワーク抽象化レイヤ (NAL

10

20

30

40

50

）ユニット・ストリームとバイト・ストリームとの間の関係を指定し、これらはいずれもビットストリームと呼ばれる。

【0018】

ビットストリームは、例えば、NALユニット・ストリーム・フォーマット又はバイト・ストリーム・フォーマットの2つのフォーマットのいずれかであり得る。NALユニット・ストリーム・フォーマットは概念的には、より「基本的な」タイプである。NALユニット・ストリーム・フォーマットは、NALユニットと呼ばれる構文構造のシーケンスを含む。このシーケンスは、復号順序で順序付けされる。NALユニット・ストリームにおけるNALユニットの復号順序（及び）に課される制約がある。

【0019】

バイト・ストリーム・フォーマットは、NALユニットを復号順に順序付け、各NALユニットに開始コードプレフィックスとゼロ以上のゼロ値バイトをプレフィックスしてバイトのストリームを形成することによって、NALユニット・ストリーム・フォーマットから構成されてもよい。NALユニット・ストリーム・フォーマットは、このバイトのストリームにおけるユニークな開始コード・プレフィックス・パターンの位置をサーチすることによって、バイト・ストリーム・フォーマットから抽出され得る。

【0020】

この項は、ビットストリームを介して与えられる、ソースと復号されたピクチャとの間の関係の一実施形態を指定する。

【0021】

ビットストリームによって表されるビデオ・ソースは、復号順序のピクチャのシーケンスである。

【0022】

通常、merge_gpm_partition_idx[x0][y0]の値は、ビットストリームから復号される。一例では、merge_gpm_partition_idx[] []の値の範囲は、0と63を含む0～63である。一例では、merge_gpm_partition_idx[] []の復号プロセスは、「バイパス」である。

【0023】

merge_gpm_partition_idx[x0][y0]が存在しないときに、それは0に等しいと推論される。

【0024】

一実装形式では、角度インデックス値angleIdxは、現在のコーディング・ブロックの幾何学的パーティションに使用される。

【0025】

一例では、角度インデックス値angleIdxは、幾何学的パーティションの角度インデックスを指定する。以下、角度インデックス値angleIdxは、パーティション角度変数angleIdxとも呼ばれる。

【0026】

現在のブロックに対する角度パラメータの値は、分裂モードのインデックス値と予め定義されたルックアップ・テーブルの値に従って取得される。

【0027】

一実施形態では、幾何学的パーティショニング・モードのパーティション角度変数angleIdx（角度パラメータ）及び距離変数distanceIdxは、以下の表に指定されるmerge_gpm_partition_idx[xCb][yCb]（インジケータ）の値に従ってセットされる。実装形式では、この関係は、表1に従って、又は機能に従って実装され得る。

10

20

30

40

50

【表 1】

表 1 : merge_gpm_partition_idx に基づく angleIdx 及び distanceIdx の仕様

merge_gpm_partition_idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
angleIdx	0	0	2	2	2	2	3	3	3	3	4	4	4	4	5	5
distanceIdx	1	3	0	1	2	3	0	1	2	3	0	1	2	3	0	1
merge_gpm_partition_idx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
angleIdx	5	5	8	8	11	11	11	11	12	12	12	12	13	13	13	13
distanceIdx	2	3	1	3	0	1	2	3	0	1	2	3	0	1	2	3
merge_gpm_partition_idx	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
angleIdx	14	14	14	14	16	16	18	18	18	19	19	19	20	20	20	21
distanceIdx	0	1	2	3	1	3	1	2	3	1	2	3	1	2	3	1
merge_gpm_partition_idx	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
angleIdx	21	21	24	24	27	27	27	28	28	28	29	29	29	30	30	30
distanceIdx	2	3	1	3	1	2	3	1	2	3	1	2	3	1	2	3

10

【0028】

一実装形式では、インデックス値 `partIdx` は、以下を満たす。

20

【数 1】

$$\text{partIdx} = (\text{angleIdx} \geq \text{threshold1} \ \&\& \ \text{angleIdx} \leq \text{threshold2}) ? 1 : 0,$$

【0029】

ここで、`threshold1` 及び `threshold2` は整数値であり、`threshold1` は `threshold2` より小さい。

【0030】

したがって、角度インデックス値 `angleIdx` が `threshold1` と `threshold2` との間の区間にあるとき、又は角度インデックス値 `angleIdx` が `threshold1` 又は `threshold2` に等しいときに、インデックス値 `partIdx` が 1 にセットされ、そうではなく、角度インデックス値 `angleIdx` がこの区間の外にあるときに、インデックス値 `partIdx` が 0 にセットされる。インデックス値 `partIdx` に従って、現在のコーディング・ブロックが復号される。例えば、インデックス値 `partIdx` は、2つのサブブロックのうちのどちらが第1のサブブロック (`partIdx = 0`) で、どちらが復号される第2のサブブロック (`partIdx = 1`) であるかを定義する。

30

【0031】

一例では、`threshold1` が 13 であり、`threshold2` が 27 である。

【0032】

一例では、インデックス値は `isFlip` として表される分裂インデックスであってもよく、インデックス値 `isFlip` は、以下も満たす。

40

【数 2】

$$\text{isFlip} = (\text{angleIdx} \geq 13 \ \&\& \ \text{angleIdx} \leq 27) ? 1 : 0.$$

【0033】

復号プロセスは、サンプル重み導出プロセス、運動情報記憶プロセス、モーション・ベクトル導出プロセスなどであってもよい。したがって、バッファ利用と復号効率が改善されている。

50

【0034】

一実装形式では、コーディング・ブロックを復号することは、インデックス値 $partIdx$ に従って、現在のブロックのためのモーション情報を記憶することを含む。

【0035】

本発明の第2の態様は、ビデオ復号器であって、現在のコーディング・ブロックに対する分裂モード・インデックス値を取得するように構成されている解析ユニットと、分裂モード・インデックス値、及び分裂モード・インデックス値に基づいて角度インデックス値 $angleIdx$ を指定するテーブルに従って、現在のコーディング・ブロックに対する角度インデックス値 $angleIdx$ を取得するように構成されている角度インデックス値取得ユニットと、角度インデックス値 $angleIdx$ に従って、インデックス値 $partIdx$ をセットするように構成されているセッティング・ユニットと、インデックス値 $partIdx$ に従って、現在のコーディング・ブロックを復号するように構成されている処理ユニットと、含む、ビデオ復号器を提供する。

10

【0036】

一実装形式では、インデックス値 $partIdx$ は、以下を満たす。

【数3】

$$partIdx = (angleIdx \geq threshold1 \ \&\& \ angleIdx \leq threshold2) ? 1 : 0,$$

ここで、 $threshold1$ 及び $threshold2$ は整数値であり、 $threshold1$ は $threshold2$ より小さい。

20

【0037】

一実装形式では、 $threshold1$ が13であり、 $threshold2$ が27である。

【0038】

一実装形式では、処理ユニットは、インデックス値 $partIdx$ に従って、現在のブロックのためのモーション情報を記憶するように構成されている。

【0039】

一実装形式では、分裂モード・インデックス値は、現在のコーディング・ブロックにどの幾何学的パーティション・モードが使用されるかを示すために使用される。

30

【0040】

一実装形式では、 $angleIdx$ は、現在のコーディング・ブロックの幾何学的パーティションに使用される。

【0041】

本発明の第1の態様による方法は、本発明の第2の態様によるデバイスによって実行され得る。上記方法のさらなる特徴及び実装形式は、本発明の第2の態様による装置の特徴及び実装形式に対応する。一実施形態では、上記の実施形態及び実装のいずれかに1つよる方法を実施するための処理回路を含む復号器が開示される。

【0042】

一実施形態では、上記の実施形態及び実装のいずれかに1つよる方法を実行するためのプログラム・コードを含むコンピュータ・プログラム製品が開示される。

40

【0043】

一実施形態では、
1つ以上のプロセッサと、
プロセッサに結合され、プロセッサによる実行のためのプログラミング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、プログラミングは、プロセッサによって実行される場合に、上記の実施形態及び実装のいずれか1つによる方法を実施するように復号器又は符号化器を設定する、非一時的なコンピュータ可読記憶媒体と、を含む、復号器又は符号化器が提供される。

【0044】

50

一実施形態では、画像復号デバイスによって復号される符号化されたビットストリームを含む非一時的な記憶媒体であって、ビットストリームは、ビデオ信号又は画像信号のフレームを複数のブロックに分割することによって生成され、複数の構文要素を含み、複数の構文要素は、上記の実施形態及び実装のいずれか1つによるインジケータ（構文）を含む、非一時的な記憶媒体が提供される。

【0045】

1つ以上の実施形態の詳細は、添付の図面及び以下の説明に記載されている。他の特徴、目的、及び利点は、明細書、図面、及び特許請求の範囲から明らかであろう。

【図面の簡単な説明】

【0046】

以下、本発明の実施形態が、添付の図及び図面を参照して、より詳細に説明されている。

【0047】

【図1A】本発明の実施形態を実装するように構成されているビデオ・コーディング・システムの一例を示すブロック図である。

【図1B】本発明の実施形態を実装するように構成されているビデオ・コーディング・システムの一例を示すブロック図である。

【図2】本発明の実施形態を実装するように構成されているビデオ符号化器の一例を示すブロック図である。

【図3】本発明の実施形態を実装するように構成されているビデオ復号器の例示的な構造を示すブロック図である。

【図4】符号化装置又は復号装置の一例を示すブロック図である。

【図5】符号化装置又は復号装置の別例を示すブロック図である。

【図6a】共位置にあるブロックの一例を示した。

【図6b】空間的隣接ブロックの一例を示した。

【図7】三角形予測モードのいくつかの例を示した。

【図8】サブブロック予測モードのいくつかの例を示した。

【図9】ブロックのパーティションに関するいくつかの例を示す。

【図10】ブロックのパーティションに関するいくつかの例を示す。

【図11】ブロックのパーティションに関するいくつかの例を示す。

【図12】ブロックのパーティションに関するいくつかの例を示す。

【図13】step Dのための予め定義されたルックアップ・テーブルの例示的な実装形式を示した。

【図14】f（）のための予め定義されたルックアップ・テーブルの例示的な実装形式を示した。

【図15】step Dのための予め定義されたルックアップ・テーブルに関する量子化アスペクトの例を示した。

【図16】与えられたコーディング・ブロックに対して最大 p_{max} が定義される量子化スキームの一例を示した。

【図17】与えられたコーディング・ブロックに対して代替的な最大 p_{max} が定義される量子化スキームの一例を示した。

【図18】コンテンツ供給システムの一例の構造を示すブロック図である。

【図19】端末デバイスの一例の構造を示すブロック図である。

【図20】本発明の別の実施形態を示した。

【図21】本発明の別の実施形態を示した。

【図22】本発明による方法を示すフローチャートである。

【図23】本発明による装置を示すブロック図である。

【図24】本発明による復号器の一実施形態を示すブロック図である。

【0048】

以下、同一の参照符号は、他に明示的に指定されていない場合は、同一の又は少なくとも機能的に同等の特徴を指す。

10

20

30

40

50

【発明を実施するための形態】

【0049】

以下の説明では、本開示の一部を形成し、本発明の実施形態の特定の態様又は本発明の実施形態が使用され得る特定の態様を例示するために、添付の図面を参照する。本発明の実施形態は、他の態様で使用することができ、図に示されていない構造的又は論理的变化を含むことが理解される。したがって、以下の詳細な説明は、限定的な意味で解釈されるべきではなく、本発明の範囲は、添付の特許請求の範囲によって定義される。

【0050】

例えば、説明された方法に関連する開示は、当該方法を実行するように構成された対応するデバイス又はシステムについても真であり、その逆もまた真であり得ることが理解される。例えば、1つ又は複数の特定の方法ステップが説明される場合、対応するデバイスは、説明された1つ又は複数の方法ステップを実行する（例えば、1つのユニットが1つ又は複数のステップを実行するか、又は複数のユニットの各々が複数のステップのうちの1つ以上を実行する）ために、1つ又は複数のユニット、例えば、機能ユニットを、たとえそのような1つ以上のユニットが明示的に説明されるか、図に示されていなくても、含んでもよい。一方、例えば、1つ又は複数のユニット、例えば、機能ユニットに基づいて特定の装置が説明される場合、対応する方法は、1つ又は複数のユニットの機能性を実行する（例えば、1つのステップが1つ又は複数のユニットの機能性を実行するか、又は複数のユニットのうちの各々が複数のユニットのうちの1つ以上の機能性を実行する複数のステップ）ために、1つのステップを、たとえそのような1つ又は複数のステップが明示的に説明されるか、図に示されていなくても、含んでもよい。さらに、本明細書で説明された種々の例示的な実施形態及び／又は態様の特徴は、特に断らない限り、互いに組み合わせてもよいことが理解される。

【0051】

ビデオ・コーディングは、典型的には、ビデオ又はビデオ・シーケンスを形成する一連のピクチャの処理を指す。用語「ピクチャ」の代わりに、用語「フレーム」又は「画像」は、ビデオ・コーディングの分野では同義語として使用されてもよい。ビデオ・コーディング（又は一般的な符号化）は、2つの部分のビデオ符号化及びビデオ復号を含む。ビデオ・コーディングは、ソース側で実行され、典型的には、元のビデオ・ピクチャを（より効率的な記憶及び／又は伝送のために）表するのに必要なデータ量を低減するために、元のビデオ・ピクチャの（例えば、圧縮による）処理を含む。ビデオ復号は、宛先側で実行され、典型的には、ビデオ・ピクチャを再構成するために符号化器と比較して、逆方向の処理を含む。ビデオ・ピクチャ（又は、一般には、ピクチャ）の「コーディング」を参照する実施形態は、ビデオ・ピクチャ又はそれぞれのビデオ・シーケンスの「符号化」又は「復号」に関係するものと理解されたい。符号化ユニットと復号ユニットの組み合わせは、CODEC（Coding and Decoding）とも呼ばれる。

【0052】

可逆ビデオ・コーディングの場合、元のビデオ・ピクチャを再構成することができ、再構成されたビデオ・ピクチャは、元のビデオ・ピクチャと同じ品質を有する（記憶又は伝送の間に伝送損失又は他のデータ損失がないと仮定する）。非可逆ビデオ・コーディングの場合、ビデオ・ピクチャを表すデータの量を低減するために、例えば量子化によるさらなる圧縮が実行されるが、復号器で完全には再構成することができず、再構成されたビデオ・ピクチャの品質は、元のビデオ・ピクチャの品質より低い又は悪い。

【0053】

いくつかのビデオ・コーディング標準は、「非可逆ハイブリッド・ビデオ・コーデック」のグループに属する（例えば、サンプル・ドメインにおける空間的及び時間的予測と、変換ドメインにおける量子化を適用するための2D変換コーディングを組み合わせる）。ビデオ・シーケンスの各ピクチャは、典型的には、重複しないブロックのセットにパーティショニングされ、コーディングは、典型的には、ブロックレベルで実行される。言い換えれば、符号化器では、ビデオは、典型的には、ブロック（ビデオ・ブロック）レベルで

、例えば、空間的（ピクチャ内）予測及び／又は時間的（ピクチャ間）予測を使用して予測ブロックを生成し、現在のブロック（現在処理されている／処理されるブロック）から予測ブロックを減算して残差ブロックを得て、残差ブロックを変換し、変換ドメイン内の残差ブロックを量子化して、送信されるデータ量を低減（圧縮）することによって、処理される、例えば符号化される一方、復号器では、符号化器と比較した逆処理が符号化されたか、又は圧縮されたブロックに適用されて、提示のための現在のブロックを再構成する。さらに、符号化器は復号器処理ループを複製し、両者が同一の予測（例えば、イントラ予測及びインター予測）を生成し、及び／又は処理、例えば、後続のブロックをコーディングするための再構成を生成するようにする。ビデオ・コーディング・システム１０の以下の実施形態では、図１～図３に基づいて、ビデオ符号化器２０及びビデオ復号器３０が説明される。

10

【００５４】

図１Ａは、例示的なコーディング・システム１０、例えば、本出願の技術を利用することができるビデオ・コーディング・システム１０（又は省略してコーディング・システム１０）を示す概略ブロック図である。ビデオ・コーディング・システム１０のビデオ符号化器２０（又は省略して符号化器２０）及びビデオ復号器３０（又は省略して復号器３０）は、本出願で説明される種々の例にしたがって技術を実行するように構成され得るデバイスの例を表す。図１Ａに示すように、コーディング・システム１０は、例えば、符号化されたピクチャ・データ１３を復号するための宛先デバイス１４に、符号化されたピクチャ・データ２１を提供するように構成されているソース・デバイス１２を含む。

20

【００５５】

ソース・デバイス１２は、符号化器２０を含み、追加的に、例えば任意選択で、ピクチャ・ソース１６、プリプロセッサ（又は前処理ユニット）１８、例えばピクチャ・プリプロセッサ１８、及び通信インターフェース又は通信ユニット２２を含んでもよい。

【００５６】

ピクチャ・ソース１６は、任意の種類のピクチャ捕捉デバイス、例えば、実世界ピクチャを捕捉するためのカメラ、及び／又は任意の種類のピクチャ生成デバイス、例えば、コンピュータ・アニメーション・ピクチャを生成するためのコンピュータ・グラフィック・プロセッサ、又は、実世界ピクチャ、コンピュータ生成ピクチャ（例えば、スクリーン・コンテンツ、バーチャル・リアリティ（ＶＲ）ピクチャ）、及び／又はそれらの任意の組み合わせ（例えば、拡張現実（ＡＲ）ピクチャ）を得る、及び／又は提供するための任意の種類の他のデバイスを含んでもよい。

30

【００５７】

ピクチャ・ソースは、上述のピクチャのいずれかを記憶する任意の種類のメモリ又はストレージであってもよい。

【００５８】

プリプロセッサ１８と、前処理ユニット１８によって実行される処理とを区別して、ピクチャ又はピクチャ・データ１７もまた、生のピクチャ又は生のピクチャ・データ１７と呼ばれてもよい。プリプロセッサ１８は、（生の）ピクチャ・データ１７を受信し、ピクチャ・データ１７に前処理を実行して、前処理されたピクチャ・データ１９又は前処理されたピクチャ・データ１９を得るように構成されている。プリプロセッサ１８によって実行される前処理は、例えば、トリミング、色フォーマット変換（例えば、ＲＧＢからＹＣｂＣｒへ）、色補正、又はノイズ除去を含んでもよい。前処理ユニット１８は、任意のコンポーネントであってもよいと理解され得る。ビデオ符号化器２０は、前処理されたピクチャ・データ１９を受信し、符号化されたピクチャ・データ２１を提供するように構成されている（例えば、図２に基づいて、さらなる詳細を以下に説明する）。ソース・デバイス１２の通信インターフェース２２は、符号化されたピクチャ・データ２１を受信し、記憶又は直接再構成のために、通信チャネル１３を介して、符号化されたピクチャ・データ２１（又はその任意のさらなる処理されたバージョン）を他のデバイス、例えば宛先デバイス１４又は任意の他のデバイスに送信するように構成されてもよい。

40

50

【0059】

宛先デバイス14は、復号器30（例えば、ビデオ復号器30）を含み、追加的、例えば任意選択で、通信インターフェース又は通信ユニット28、ポストプロセッサ32（又は後処理ユニット32）、及び表示デバイス34を含んでもよい。

【0060】

宛先デバイス14の通信インターフェース28は、符号化されたピクチャ・データ21（又はそのさらなる処理されたバージョン）を、例えばソース・デバイス12から直接、又は任意の他のソース、例えば記憶デバイス、例えば符号化されたピクチャ・データ記憶デバイスから受信し、符号化されたピクチャ・データ21を復号器30に提供するように構成されている。

10

【0061】

通信インターフェース22及び通信インターフェース28は、ソース・デバイス12と宛先デバイス14との間の直接通信リンク、例えば直接有線若しくは無線接続、又は任意の種類のネットワーク、例えば有線若しくは無線ネットワーク若しくはそれらの任意の組み合わせ、若しくは任意の種類のプライベート及びパブリック・ネットワーク、又はそれらの任意の種類の組み合わせを介して、符号化されたピクチャ・データ21又は符号化されたデータ13を送信又は受信するように構成されてもよい。

【0062】

通信インターフェース22は、例えば、符号化されたピクチャ・データ21を適切なフォーマット、例えば、パケットにパッケージングし、及び／又は通信リンク又は通信ネットワークを介した伝送のための任意の種類の伝送符号化又は処理を使用して符号化されたピクチャ・データを処理するように構成されてもよい。

20

【0063】

通信インターフェース22のカウンタパートを形成する通信インターフェース28は、例えば、送信されたデータを受信し、任意の種類の対応する伝送復号又は処理及び／又はパッケージング解除を使用して伝送データを処理して、符号化されたピクチャ・データ21を得るように構成されてもよい。

【0064】

通信インターフェース22及び通信インターフェース28は両方とも、ソース・デバイス12から宛先デバイス14へ指し示す図1Aの通信チャネル13の矢印によって示されるように、一方向通信インターフェース、又は双方向通信インターフェースとして構成されてもよく、例えば、通信リンク及び／又はデータ伝送例えば、符号化されたピクチャ・データ伝送に関係する任意の他の情報を確認応答及び交換するために、メッセージを送信及び受信する、例えばコネクションをセット・アップするように構成されてもよい。

30

【0065】

復号器30は、符号化されたピクチャ・データ21を受信し、復号されたピクチャ・データ31又は復号されたピクチャ31を提供するように構成されている（例えば、図3又は図5に基づいて、さらなる詳細を以下に説明する）。

【0066】

宛先デバイス14のポストプロセッサ32は、復号されたピクチャ・データ31、例えば復号されたピクチャ31を後処理して、後処理されたピクチャ・データ33、例えば後処理されたピクチャ33を得るように構成されている。後処理ユニット32によって実行される後処理は、例えば、色フォーマット変換（例えば、YCbCrからRGBへ）、色補正、トリミング、又は再サンプリング、又は、例えば、表示デバイス34による表示のために、復号されたピクチャ・データ31を準備するための任意の他の処理を含むことができる。宛先デバイス14の表示デバイス34は、例えばユーザ又はビューアに、ピクチャを表示するために後処理されたピクチャ・データ33を受信するように構成されている。表示デバイス34は、再構成されたピクチャを表すための任意の種類のディスプレイ、例えば、統合されたディスプレイ又は外部ディスプレイ若しくはモニタであってもよく、又はそれらを含んでもよい。ディスプレイは、例えば、液晶ディスプレイLCD）、有機

40

50

発光ダイオード（O L E D）ディスプレイ、プラズマ・ディスプレイ、プロジェクタ、マイクロLEDディスプレイ、液晶オン・シリコン（L C o S）、デジタル光プロセッサ（D L P）、又は任意の種類の他のディスプレイを含むことができる。

【0067】

図1Aは、ソース・デバイス12及び宛先デバイス14を別個のデバイスとして描いているのであるが、デバイスの実施形態は、両方又は両方の機能、ソース・デバイス12又は対応する機能性、及び宛先デバイス14又は対応する機能性を含んでもよい。そのような実施形態では、ソース・デバイス12又は対応する機能性と宛先デバイス14又は対応する機能は、同じハードウェア及び／又はソフトウェア、別個のハードウェア及び／又はソフトウェア又はそれらの任意の組み合わせによって実装されてもよい。

10

【0068】

説明に基づいて当業者には明らかなように、図1Aに示すように、ソース・デバイス12及び／又は宛先デバイス14内の異なるユニット又は機能性の存在及び機能性の（正確な）分裂は、実際のデバイス及び用途に依存して変動し得る。

【0069】

符号化器20（例えば、ビデオ符号化器20）、復号器30（例えば、ビデオ復号器30）、又は符号化器20と復号器30の両方は、図1Bに示すような処理回路、1つ以上のマイクロプロセッサ、デジタル信号プロセッサ（D S P）、特定用途向け集積回路（A S I C）、フィールドプログラマブル・ゲート・アレイ（F P G A）、ディスクリート論理、ハードウェア、専用のビデオ・コーディング、又はそれらの任意の組み合わせによって実装されてもよい。符号化器20は、図2の符号化器20及び／又は本明細書で説明される任意の他の符号化器システム又はサブシステムに関して説明したように、種々のモジュールを具体化するために、処理回路46を介して実装されてもよい。復号器30は、図3の復号器30及び／又は本明細書で説明される任意の他の復号器システム又はサブシステムに関して説明されるように、種々のモジュールを具体化するために、処理回路46を介して実装されてもよい。処理回路は、後述するように、種々の演算を実行するように構成されてもよい。図5に示すように、技術が部分的にソフトウェアで実装される場合、デバイスは、好適な非一時的なコンピュータ可読記憶媒体にソフトウェアのための命令を記憶し、本開示の技術を実行するために1つ以上のプロセッサを使用してハードウェアで命令を実行することができる。

20

ビデオ符号化器20とビデオ復号器30のいずれかは、例えば図1Bに示すように、単一のデバイス内の組み合わされた符号化器／復号器（C O D E D）の一部として一体化されてもよい。

30

【0070】

ソース・デバイス12及び宛先デバイス14は、ノートブック又はラップトップ・コンピュータ、携帯電話、スマートフォン、タブレット又はタブレット・コンピュータ、カメラ、デスクトップ・コンピュータ、セット・トップ・ボックス、テレビ、表示デバイス、デジタル・メディア・プレーヤ、ビデオゲーム機、ビデオ・ストリーミング装置（コンテンツ・サービス・サーバ又はコンテンツ配信サーバなど）、放送受信機デバイス、放送送信機装置など、任意の種類のハンドヘルド・デバイス又は固定デバイスを含む広範囲のデバイスのいずれも含んでもよく、オペレーティング・システムを全く使用しないか、又は任意の種類のものを使用してもよい。

40

【0071】

場合によっては、ソース・デバイス12及び宛先デバイス14は、無線通信に装備されてもよい。したがって、ソース・デバイス12及び宛先デバイス14は、無線通信デバイスであってもよい。場合によっては、図1Aに示されたビデオ・コーディング・システム10は、単なる一例に過ぎず、本出願の技術は、必ずしも符号化デバイスと復号デバイスとの間の任意のデータ通信を含まないビデオ・コーディング・セッティング（例えば、ビデオ符号化又はビデオ復号）に適用されてもよい。他の例では、データはローカルメモリから検索され、ネットワークを介してストリーミングされるなどである。ビデオ符号化

50

デバイスは、データを符号化し、メモリに記憶してもよく、及び／又はビデオ復号デバイスは、メモリからデータを検索し、復号してもよい。いくつかの例では、符号化及び復号は、互いに通信せず、単にデータをメモリに符号化し、及び／又はメモリからデータを検索及び復号するデバイスによって実行される。

【0072】

説明の便宜上、本発明の実施形態は、例えば、高効率ビデオ・コーディング（HEVC）又は汎用ビデオ・コーディング（VVC）の参照ソフトウェア、ITU-Tビデオ・コーディング・エキスパート・グループ（VCEG）及びISO/IECモーション・ピクチャ・コーディング・エキスパート・グループ（MPEG）のジョイント・コラボレーション・チーム・オン・ビデオ・コーディング（JCT-VC）によって開発された次世代ビデオ符号化規格を参照して、本明細書に説明される。当業者は、本発明の実施形態がHEVC又はVVCに限定されないことを理解するであろう。

【0073】

符号化器及び符号化方法

【0074】

図2は、本出願の技術を実装するように構成されている例示的なビデオ符号化器20の概略ブロック図を示す。図2の例では、ビデオ符号化器20は、入力201（又は入力インターフェース201）、残差計算ユニット204、変換処理ユニット206、量子化ユニット208、逆量子化ユニット210、逆変換処理ユニット212、再構成ユニット214、ループ・フィルタ・ユニット220、復号されたピクチャ・バッファ（DPB）230、モード選択ユニット260、エントロピー符号化ユニット270、出力272（又は出力インターフェース272）を含む。モード選択ユニット260は、インター予測ユニット244と、イントラ予測ユニット254と、パーティショニング・ユニット262とを含んでもよい。インター予測ユニット244は、モーション推定ユニットと、モーション補償ユニット（図示せず）とを含んでもよい。図2に示すビデオ符号化器20は、ハイブリッド・ビデオ符号化器又はハイブリッド・ビデオ・コーデックにしたがったビデオ符号化器とも呼ばれることがある。

【0075】

残差演算ユニット204、変換処理ユニット206、量子化ユニット208、モード選択ユニット260は符号化器20の順方向信号経路を形成するものとして呼ばれることがあるが、逆量子化ユニット210、逆変換処理ユニット212、再構成ユニット214、バッファ216、ループ・フィルタ220、復号されたピクチャ・バッファ（DPB）230、インター予測ユニット244、及びイントラ予測ユニット254は、ビデオ符号化器20の逆方向信号経路を形成するものとして呼ばれることがあり、ビデオ符号化器20の逆方向信号経路は復号器の信号経路に対応する（図3のビデオ復号器30を参照）。また、逆量子化ユニット210、逆変換処理ユニット212、再構成ユニット214、ループ・フィルタ220、復号されたピクチャ・バッファ（DRB）230、インター予測ユニット244、及びイントラ予測ユニット254も、ビデオ符号化器20の「内蔵復号器」を形成すると呼ばれる。

【0076】

ピクチャ&ピクチャのパーティショニング（ピクチャ&ブロック）

【0077】

符号化器20は、例えば入力201を介してピクチャ17（又はピクチャ・データ17）、例えばビデオ又はビデオ・シーケンスを形成するピクチャのシーケンスのピクチャを受信するように構成されてもよい。受信されたピクチャ又はピクチャ・データは、前処理されたピクチャ19（又は前処理されたピクチャ・データ19）でもよい。簡単のために、以下の説明は、ピクチャ17を指す。また、ピクチャ17は、現在のピクチャ又は（特に、現在のピクチャを他のピクチャ、例えば同じビデオ・シーケンス、例えば現在のピクチャも含むビデオ・シーケンスの前に符号化及び／又は復号されたピクチャから区別するためのビデオ・コーディングにおいて）符号化されるピクチャと呼ばれることがある。

【0078】

(デジタル) ピクチャは、強度値を有するサンプルの二次元配列又はマトリックスであるか、又はこれとみなされることがある。アレイ中のサンプルは、画素(ピクチャ要素の省略した形式)又はペルとも呼ばれることがある。アレイ又はピクチャの水平方向及び垂直方向(又は軸)のサンプル数は、ピクチャのサイズ及び／又は解像度を定義する。色の表現には、典型的には、3つの色コンポーネントが使用され、例えばピクチャは、3つのサンプル・アレイを表現するか、又は含んでもよい。RGBフォーマット又は色空間において、ピクチャは、対応する赤、緑及び青のサンプル・アレイを含む。しかしながら、ビデオ・コーディングにおいては、各画素は、典型的には、輝度及び色度フォーマット又は色空間、例えば、Yで示される輝度コンポーネント、及びCb及びCrで示される2つの色度コンポーネントを含むYCbCrで表わされる。輝度(又は省略してルマ)コンポーネントYは、輝度又はグレーレベル強度(例えば、グレースケール・ピクチャなど)を表し、2つの色度(又は省略してクロマ)成分Cb及びCrは、色度又は色情報コンポーネントを表す。したがって、YCbCrフォーマットのピクチャは、輝度サンプル値(Y)の輝度サンプル・アレイと、色度値(Cb及びCr)の2つの色度サンプル・アレイとを含む。RGB形式のピクチャはYCbCr形式に転換又は変換されてもよく、その逆も可能であり、プロセスは色変換又は転換としても知られている。ピクチャがモノクロである場合、ピクチャは、輝度サンプル・アレイのみを含んでもよい。したがって、ピクチャは、例えば、モノクロフォーマットのルマ・サンプルのアレイ、又はルマ・サンプルのアレイ、及び4:2:0、4:2:2、及び4:4:4色フォーマットのクロマ・サンプルの2つの対応するアレイであり得る。

10

20

【0079】

ビデオ符号化器20の実施形態は、ピクチャ17を複数の(典型的には重複しない)ピクチャ・ブロック203にパーティショニングするように構成されているピクチャ・パーティショニング・ユニット(図2には示さず)を含んでもよい。これらのブロックは、ルート・ブロック、マクロ・ブロック(H.264/AVC)又はコーディング・ツリー・ブロック(CTB)若しくはコーディング・ツリー・ユニット(CTU)(H.265/HEVC及びVVC)とも呼ばれることがある。ピクチャ・パーティショニング・ユニットは、ビデオ・シーケンスのすべてのピクチャに対して同じブロックサイズと、ブロックサイズを定義する対応するグリッドを使用するか、又はピクチャ間、ピクチャのサブセット又はグループ間でブロックサイズを変更し、各ピクチャを対応するブロックに分割するように構成されてもよい。

30

【0080】

さらなる実施形態では、ビデオ符号化器は、ピクチャ17のブロック203、例えば、ピクチャ17を形成する1つ、複数の、又はすべてのブロックを直接受信するように構成されてもよい。ピクチャ・ブロック203は、現在のピクチャ・ブロック又は符号化されるピクチャ・ブロックとも呼ばれることがある。

【0081】

ピクチャ17と同様に、ピクチャ・ブロック203は、再度ピクチャ17より小さい寸法ではあるが、強度値(サンプル値)を有するサンプルの2次元アレイ又はマトリックスであるか、又はそのように見なされる。言い換えれば、ブロック203は、例えば、1つのサンプル・アレイ(例えば、モノクロ・ピクチャ17の場合はルマ・アレイ、色ピクチャの場合はルマ又はクロマ・アレイ)又は3つのサンプル・アレイ(例えば、色ピクチャ17の場合はルマ及び2つのクロマ・アレイ)又は適用される色フォーマットに依存した任意の他の数及び／又は種類のアレイを含んでもよい。ブロック203の方向及び垂直方向(又は軸)のサンプル数は、ブロック203のサイズを定義する。したがって、ブロックは、例えば、サンプルのM×N(M列×N行)アレイ、又は変換係数のM×Nアレイであってもよい。

40

【0082】

図2に示すビデオ符号化器20の実施形態は、ブロック毎にピクチャ17を符号化する

50

ように構成されてもよく、例えば、符号化及び予測はブロック 203 毎に実行される。

【0083】

残差計算

【0084】

残差計算ユニット 204 は、サンプル毎（画素毎）に、例えば、ピクチャ・ブロック 203 のサンプル値から予測ブロック 265 のサンプル値を減算することにより、ピクチャ・ブロック 203 及び予測ブロック 265（予測ブロック 265 についてのさらなる詳細は後述する）に基づいて残差ブロック 205（残差 205 とも呼ばれる）を計算して、サンプル・ドメインにおける残差ブロック 205 を得るように構成されてもよい。

【0085】

変換

【0086】

変換処理ユニット 206 は、残差ブロック 205 のサンプル値に変換、例えば離散コサイン変換（DCT）又は離散サイン変換（DST）を適用して、変換ドメインにおける変換係数 207 を得るように構成されてもよい。変換係数 207 は、変換残差係数とも呼ばれ、変換ドメインにおける残差ブロック 205 を表してもよい。

【0087】

変換処理ユニット 206 は、H. 265 / HEVC に対して指定された変換など、DCT / DST の整数近似を適用するように構成されてもよい。直交 DCT 変換と比較して、そのような整数近似は、典型的には、特定のファクタによってスケールされる。順変換と逆変換によって処理される残差ブロックのノルムを保存するために、変換プロセスの一部として付加的なスケール・ファクタが適用される。スケール・ファクタは、典型的には、シフト演算のための 2 のべき乗であるスケール・ファクタ、変換係数のビット深度、精度と実装コストの間のトレードオフなどのような特定の制約に基づいて選択される。特定のスケール・ファクタは、例えば、逆変換処理ユニット 212（及び、例えばビデオ復号器 30 における逆変換処理ユニット 312 による対応する逆変換）によって、逆変換に対して指定され、符号化器 20 において、例えば、変換処理ユニット 206 によって、順変換に対する対応するスケール係数が、それに従って指定されてもよい。

【0088】

ビデオ符号化器 20（それぞれ、変換処理ユニット 206）の実施形態は、例えば、エントロピー符号化ユニット 270 を介して直接又は符号化もしくは圧縮された変換パラメータ、例えば、変換パラメータを出力するように構成されてもよく、これにより、例えば、ビデオ復号器 30 は、復号のために変換パラメータを受信及び使用してもよい。

【0089】

量子化

【0090】

量子化ユニット 208 は、例えばスカラー量子化又はベクトル量子化を適用することによって、変換係数 207 を量子化して量子化係数 209 を得るように構成されてもよい。量子化係数 209 は、量子化変換係数 209 又は量子化残差係数 209 とも呼ばれることがある。

【0091】

量子化プロセスは、変換係数 207 の一部又は全部に関連するビット深度を低減してもよい。例えば、 n ビット変換係数は、量子化の間、 m ビット変換係数に丸められてもよく、 n は、 m より大きい。量子化の程度は、量子化パラメータ（QP）を調整することによって修正されてもよい。例えば、スカラー量子化では、異なるスケールを適用して、より微細又はより粗大な量子化を達成してもよい。より小さい量子化ステップ・サイズはより微細な量子化に対応し、より大きい量子化ステップ・サイズはより粗大な量子化に対応する。適用可能な量子化ステップ・サイズは、量子化パラメータ（QP）によって示されてもよい。量子化パラメータは、例えば、適用可能な量子化ステップ・サイズの事前定義

10

20

30

40

50

されたセットに対するインデックスであってもよい。例えば、小さな量子化パラメータは、微細な量子化（小さな量子化ステップ・サイズ）に対応してもよく、大きな量子化パラメータは、粗大な量子化（大きな量子化ステップ・サイズ）に対応してもよく、又はその逆に対応してもよい。量子化は、量子化ステップ・サイズによる除算を含んでもよく、例えば、逆量子化ユニット 210 による、対応する及び／又は逆の量子化解除は、量子化ステップ・サイズによる乗算を含んでもよい。いくつかの規格、例えば H E V C による実施形態は、量子化ステップ・サイズを判定するために量子化パラメータを使用するように構成されてもよい。一般に、量子化ステップ・サイズは、除算を含む式の固定点近似を使用して量子化パラメータに基づいて計算されてもよい。量子化ステップ・サイズ及び量子化パラメータの方程式の固定点近似に使用されるスケーリングのために修正され得る残差ブロックのノルムを復元するために、量子化及び量子化解除のために追加のスケーリング・ファクタを導入してもよい。一実施例では、逆変換及び脱量子化のスケーリングが組み合わせられてもよい。代替的には、カスタマイズされた量子化テーブルが使用され、符号化器から復号器へ、例えばビットストリームでシグナリングされてもよい。量子化は、損失が量子化ステップ・サイズの増加に伴って増加する不可逆演算である。ビデオ符号化器 20（それぞれ、量子化ユニット 208）の実施形態は、例えば、直接又はエントロピー符号化ユニット 270 を介して符号化された量子化パラメータを出力するように構成されてもよく、これにより、例えば、ビデオ復号器 30 は、復号のために量子化パラメータを受信及び適用してもよい。

【0092】

逆量子化

【0093】

逆量子化ユニット 210 は、例えば、量子化ユニット 208 と同じ量子化ステップ・サイズに基づいて、又は同じ量子化ステップ・サイズを使用して、量子化ユニット 208 によって適用される量子化方式の逆を適用することによって、量子化ユニット 208 の逆量子化を量子化係数に適用して、量子化解除係数 211 を得るように構成される。また、量子化解除係数 211 は、量子化解除残差係数 211 と呼ばれ、典型的には、量子化による損失のために変換係数と同一ではないが、変換係数 207 に対応する。

【0094】

逆変換

【0095】

逆変換処理ユニット 212 は、変換処理ユニット 206 によって適用された変換の逆変換、例えば逆離散余弦変換（D C T）、逆離散正弦変換、又は他の逆変換を適用して、サンプル・ドメイン内の再構成された残差ブロック 213（又は対応する量子化解除係数 213）を得るように構成されている。再構成された残差ブロック 213 は、変換ブロック 213 とも呼ばれる。

【0096】

再構成

【0097】

再構成ユニット 214（例えば、加算器又は合計器 214）は、変換ブロック 213（再構成された残差ブロック 213）を予測ブロック 365 に加算し、例えば、再構成された残差ブロック 213 のサンプル値及び予測ブロック 265 のサンプル値をサンプル毎に加算することによって、サンプル・ドメイン内の再構成されたブロック 215 を得るように構成されている。

【0098】

フィルタリング

【0099】

ループ・フィルタ・ユニット 220（又は略して「ループ・フィルタ」220）は、再構成されたブロック 215 をフィルタリングしてフィルタリングされたブロック 221 を得るか、又は一般に、再構成されたサンプルをフィルタリングしてフィルタリングされた

10

20

30

40

50

サンプルを得るように構成されている。ループ・フィルタ・ユニットは、例えば、画素遷移を平滑化するように、又は、そうでなければ、ビデオ品質を改善するように構成されている。ループ・フィルタ・ユニット 220 は、ブロッキング解除フィルタ、サンプル・アダプティブ・オフセット (SAO) フィルタ、若しくは 1 つ以上の他のフィルタ、例えばバイラテラル・フィルタ、アダプティブ・ループ・フィルタ (ALF)、鮮鋭化フィルタ、平滑化フィルタ、若しくは共同フィルタ、又はそれらの任意の組み合わせなどの 1 つ以上のループ・フィルタを含んでもよい。ループ・フィルタ・ユニット 220 は、図 2 ではループ内フィルタとして示されているが、他の構成では、ループ・フィルタ・ユニット 220 は、ポスト・ループ・フィルタとして実装されてもよい。フィルタリングされたブロック 221 はまた、フィルタリングされた再構成されたブロック 221 と呼ばれることがある。ビデオ符号化器 20 (それぞれ、ループ・フィルタ・ユニット 220) の実施形態は、例えば、直接又はエントロピー符号化ユニット 270 を介して符号化されたループ・フィルタ・パラメータを出力するように構成されてもよく、これにより、例えば、復号器 30 は、復号のために同じループ・フィルタ・パラメータ又はそれぞれのループ・フィルタを受信及び適用してもよい。

【0100】

復号されたピクチャ・バッファ

【0101】

復号されたピクチャ・バッファ (DRB) 230 は、ビデオ符号化器 20 によってビデオ・データを符号化するために、参照ピクチャ又は一般に、参照ピクチャ・データを記憶するメモリであってもよい。DPB 230 は、同期 DRAM (SDRAM)、磁気抵抗 RAM (MRAM)、抵抗 RAM (RRAM)、又は他のタイプのメモリ・デバイスを含む、ダイナミック・ランダム・アクセス・メモリなどの種々のメモリ・デバイスのいずれかによって形成されてもよい。復号されたピクチャ・バッファ 230 は、1 つ以上のフィルタリングされたブロック 221 を記憶するように構成されてもよい。復号されたピクチャ・バッファ 230 は、同じ現在のピクチャ又は異なるピクチャ、例えば以前に再構成されたピクチャの、他の以前にフィルタリングされたブロック、例えば以前に再構成され、フィルタリングされたブロック 221 を記憶するようにさらに構成されてもよく、例えば、インター予測のために、完全に以前に再構成された、例えば復号されたピクチャ (並びに対応する参照ブロック及びサンプル)) 及び／又は部分的に再構成された現在のピクチャ (並びに対応する参照ブロック及びサンプル) を提供してもよい。復号されたピクチャ・バッファ (DRB) 230 はまた、例えば、再構成されたブロック 215 がループ・フィルタ・ユニット 220 によってフィルタリングされたものではないか、又は再構成されたブロック又はサンプルの任意の他のさらなる処理バージョンではない場合、1 つ以上のフィルタリングされていない構成されたブロック 215、又は一般に、フィルタリングされていない再構成されたサンプルを記憶するように構成されてもよい。

【0102】

モード選択 (パーティショニング & 予測)

【0103】

モード選択ユニット 260 は、パーティショニング・ユニット 262 と、インター予測ユニット 244 と、イントラ予測ユニット 254 を含み、元のピクチャ・データ、例えば元のブロック 203 (現在のピクチャ 17 の現在のブロック 203) と、再構成されたピクチャ・データ、例えば、フィルタリングされた、又はフィルタリングされていない、同じ (現在の) ピクチャ、及び／又は、1 つ又は複数の以前に復号されたピクチャ、例えば、復号されたピクチャ・バッファ 230 又は他のバッファ (例えば、ラインバッファ、図示せず) からの再構成されたサンプル又はブロックと、を受信し得るようにより構成されている。再構成されたピクチャ・データは、予測、例えば、インター予測又はイントラ予測のための参照ピクチャ・データとして使用されて、予測ブロック 265 又は予測子 265 を得る。

【0104】

モード選択ユニット260は、現在のブロック予測モード（パーティショニングなしを含む）及び予測モード（例えば、イントラ又はインター予測モード）に対するパーティショニングを判定又は選択し、残差ブロック205の計算及び再構成されるブロック215の再構成のために使用される対応する予測ブロック265を生成するように構成されてもよい。

【0105】

モード選択ユニット260の実施形態は、（例えば、モード選択ユニット260によってサポートされる、又は利用可能なものから）パーティショニング及び予測モードを選択するように構成されてもよく、これは、最良の一致、言い換えれば、最小の残差（最小の残差は、伝送又は記憶のためのより良い圧縮を意味する）、又は最小のシグナリング・オーバーヘッド（最小のシグナリングは、伝送又は記憶のためのより良いの圧縮を意味する）、又は両方を考慮若しくはバランスをとったものを提供する。モード選択ユニット260は、レート歪み最適化（RDO）に基づいて、パーティショニング及び予測モードを判定する、例えば最小レート歪みを提供する予測モードを選択するように構成されてもよい。この文脈における「最良」、「最低」、「最適」などの用語は、必ずしも全体的な「最良」、「最低」、「最適」などを指すのではなく、終了基準若しくは例えば、閾値を超えるか、又は下回る値のような選択基準、又は「準最適の選択」につながる可能性があるが複雑性及び処理時間を短縮する他の制約条件の達成を指してもよい。

【0106】

言い換えれば、パーティショニング・ユニット262は、ブロック203を、例えば、クワッド・ツリー・パーティショニング（QT）、バイナリ・パーティショニング（BT）、トリプル・ツリー・パーティショニング（TT）又はそれらの任意の組み合わせを反復的に使用して、より小さなブロック・パーティション又はサブブロック（再度ブロックを形成する）にパーティショニングし、例えば、ブロック・パーティション又はサブブロックごとに予測を実行するように構成されてもよく、モード選択は、パーティショニングされたブロック203のツリー構造の選択を含み、予測モードは、ブロック・パーティション又はサブブロックの各々に適用される。

【0107】

以下、例示的なビデオ符号化器20によって実行されるパーティショニング（例えば、パーティショニング・ユニット260による）及び予測処理（インター予測ユニット244及びイントラ予測ユニット254による）をより詳細に説明する。

【0108】

パーティショニング

【0109】

パーティショニング・ユニット262は、現在のブロック203を、例えば正方形又は長方形のサイズのより小さいブロックのような、より小さなパーティションにパーティショニング（又は分裂）することができる。これらのより小さなブロック（サブブロックとも呼ばれる）は、さらに小さなパーティションにパーティショニングされてもよい。これは、ツリー・パーティショニング又は階層ツリー・パーティショニングとも呼ばれ、ルート・ブロック、例えば、ルート・ツリー・レベル0（階層レベル0、深度0）では、再帰的にパーティショニングされてもよく、例えば、ツリー・レベル1（階層レベル1、深度1）でのノードのような次の下位ツリー・レベルの2つ以上のブロックにパーティショニングされてもよく、これらのブロックは、例えば、最大ツリー深度又は最小ブロックサイズに達するなど、例えば、終了基準が満たされるために、パーティショニングが終了するまで、例えば、ツリー・レベル2（階層レベル2、深度2）のような次の下位レベルの2つ以上のブロックに再度パーティショニングされてもよい。それ以上パーティショニングされないブロックは、ツリーのリーフ・ブロック又はリーフ・ノードとも呼ばれる。2つのパーティションへのパーティショニングを使用するツリーはバイナリ・ツリー（BT）、3つのパーティションへのパーティショニングを使用するツリーはターナリ・ツリー（TT）、4つのパーティションへのパーティショニングを使用するツリーはクワッド・

ツリー（QT）と呼ばれる。前述のように、本明細書で使用される場合、用語「ブロック」は、ピクチャの一部分、特に正方形又は長方形の部分であってもよい。例えば、HEVC及びVVCを参照すると、ブロックは、コーディング・ツリー・ユニット（CTU）、コーディング・ユニット（CU）、予測ユニット（PU）、及び変換ユニットに対応してもよく、及び／又は対応するブロック、例えば、コーディング・ツリー・ブロック（CTB）、コーディング・ブロック（CB）、変換ブロック（TB）、又は予測ブロック（PB）に対応してもよい。

【0110】

例えば、コーディング・ツリー・ユニット（CTU）は、ルマ・サンプルのCTB、3つのサンプル・アレイを有するピクチャのクロマ・サンプルの2つの対応するCTB、又はモノクロ・ピクチャ若しくはサンプルをコーディングするために使用される3つの別々の色プレーン及び構文構造を使用してコーディングされるピクチャのサンプルのCTBであってもよく、又は、これらを含んでもよい。これに対応して、コーディング・ツリー・ブロック（CTB）は、あるコンポーネントのCTBへの分割がパーティショニングであるように、ある値のNのサンプルのN×Nブロックであってもよい。コーディング・ユニット（CU）は、ルマ・サンプルのCTB、3つのサンプル・アレイを有するピクチャのクロマ・サンプルの2つの対応するコーディング・ブロック、又はモノクロ・ピクチャ若しくはサンプルをコーディングするために使用される3つの別々の色プレーン及び構文構造を使用してコーディングされるピクチャのサンプルのコーディング・ブロックであってもよく、又は、これらを含んでもよい。これに対応して、コーディング・ブロック（CB）は、あるCTBのコーディング・ブロックへの分割がパーティショニングであるように、ある値のM及びNのサンプル値のサンプルのN×Nブロックであってもよい。

【0111】

実施形態において、例えば、HEVCにしたがって、コーディング・ツリー・ユニット（CTU）は、コーディング・ツリーとして示されるクワッド・ツリー構造を使用することによって、CUに分裂されてもよい。ピクチャ間（時間的）又はピクチャ内（空間的）予測を使用してピクチャ・エリアをコーディングするかどうかの判定は、CUレベルで行われる。各CUはさらに、PU分裂タイプに従って、1つ、2つ、又は4つのPUに分裂させることができる。1つのPU内では、同じ予測プロセスが適用され、関連情報がPUベースで復号器に送信される。PU分裂タイプに基づく予測プロセスを適用することによって残差ブロックを得た後、CUを、CUのコーディング・ツリーに類似した別のクワッド・ツリー構造に従って変換ユニット（TU）にパーティショニングすることができる。実施形態では、例えば、現在開発中の最新のビデオ・コーディング規格、汎用ビデオ・コーディング（VVC）と呼ばれるものにしたがって、クワッド・ツリー及びバイナリ・ツリー（QTB T）パーティショニングが、例えば、コーディング・ブロックをパーティショニングするために使用される。QTB Tブロック構造では、CUは正方形又は長方形のいずれかの形状を有することができる。例えば、コーディング・ツリー・ユニット（CTU）は、最初にクワッド・ツリー構造によってパーティショニングされる。クワッド・ツリー・リーフ・ノードは、バイナリ・ツリー又はターナリ（若しくはトリプル）ツリー構造によってさらにパーティショニングされる。パーティショニング・ツリー・リーフ・ノードは、コーディング・ユニット（CU）と呼ばれ、セグメントは、それ以上のパーティショニングなしで予測及び変換処理のために使用される。これは、CU、PU、TUがQTB Tコーディング・ブロック構造において同じブロックサイズを有することを意味する。並列に、複数のパーティション、例えば、トリプル・ツリー・パーティションが、QTB Tブロック構造と共に使用されることも提案されている。

【0112】

一例では、ビデオ符号化器20のモード選択ユニット260は、本明細書で説明されるパーティショニング技術の任意の組み合わせを実行するように構成されてもよい。

【0113】

上述のように、ビデオ符号化器20は、（予め判定された）予測モードのセットから最

10

20

30

40

50

良又は最適な予測モードを判定又は選択するように構成されている。

予測モードのセットは、例えば、イントラ予測モード及び／又はインター予測モードを含んでもよい。

【0114】

イントラ予測

【0115】

イントラ予測モードのセットは、35の異なるイントラ予測モード、例えば、DC（又は平均）モード及び平面モードのような非方向モード、又は、例えば、HEVCで定義されているような方向モードを含むことができ、又は、67の異なるイントラ予測モード、例えば、DC（又は平均）モード及び平面モードのような非方向モード、又は、例えば、VV Cに対して定義されているような方向モードを含むことができる。

10

【0116】

イントラ予測ユニット254は、イントラ予測モードのセットのイントラ予測モードに従って、同じの現在のピクチャの隣接ブロックの再構成されたサンプルを使用して、イントラ予測ブロック265を生成するように構成されている。

【0117】

イントラ予測ユニット254（又は、一般に、モード選択ユニット260）は、符号化されたピクチャ・データ21に含めるための構文要素266の形式で、エントロピー符号化ユニット270にイントラ予測パラメータ（又は、一般に、ブロックに対して選択されたイントラ予測モードを示す情報）を出力するようにさらに構成されており、これにより、例えば、ビデオ復号器30は、復号のために予測パラメータを受信し、使用することができる。

20

【0118】

インター予測

【0119】

インター予測のセット（又は可能なインター予測モード）は、利用可能な参照ピクチャ（例えば、DBP230に記憶された、例えば少なくとも部分的に復号されたピクチャ）及び他のインター予測パラメータ、例えば、参照ピクチャの全体又は参照ピクチャの一部のみ、例えば、現在のブロックのエリアの周囲のサーチ・ウィンドウ・エリアが、最良一致の参照ブロックをサーチするために使用されるかどうか、及び／又は、例えば、画素補間、例えば、半分／セミペル及び／又はクォータ・ペル補間が適用されるかどうか依存する。

30

上記の予測モードに加えて、スキップ・モード及び／又は直接モードが適用されてもよい。インター予測ユニット244は、モーション推定（ME）ユニットと、モーション補償（MC）ユニット（両方とも図2には示さず）を含んでもよい。モーション推定ユニットは、モーション推定のために、ピクチャ・ブロック203（現在のピクチャ17の現在のピクチャ・ブロック203）及び復号されたピクチャ231、又は少なくとも1つ若しくは複数の以前に再構成されたブロック、例えば、1つ又は複数の他の／異なる以前に復号されたピクチャ231の再構成されたブロックを受信又は取得するように構成されてもよい。例えば、ビデオ・シーケンスは、現在のピクチャと以前に復号されたピクチャ231とを含んでもよく、言い換えれば、現在のピクチャと以前に復号されたピクチャ231は、ビデオ・シーケンスを形成するピクチャの一部であってもよく、又は、ピクチャのシーケンスを形成してもよい。

40

【0120】

符号化器20は、例えば、複数の他のピクチャの同じ又は異なるピクチャの複数の参照ブロックから参照ブロックを選択し、参照ブロックの位置（x、y座標）と現在のブロックの位置との間の参照ピクチャ（又は参照ピクチャ・インデックス）及び／又はオフセット（空間オフセット）を、モーション推定ユニットへのインター予測パラメータとして提供するように構成されてもよい。このオフセットはモーション・ベクトル（MV）とも呼ばれる。

50

【0121】

モーション補償ユニットは、インター予測パラメータを取得、例えば、受信し、インター予測パラメータに基づいて、又はこれを使用して、インター予測を実行して、インター予測ブロック265を取得するように構成されている。モーション補償ユニットによって実行されるモーション補償は、モーション推定によって判定されたモーション／ブロック・ベクトルに基づいて予測ブロックをフェッチ又は生成することを伴ってもよく、おそらくサブピクセル精度までの補間を実行する。補間フィルタリングは、既知の画素サンプルから追加の画素サンプルを生成することができ、したがって、ピクチャ・ブロックをコーディングするために使用され得る候補予測ブロックの数を潜在的に増加させる。現在のピクチャ・ブロックのPUに対するモーション・ベクトルを受信すると、モーション補償ユニットは、参照ピクチャ・リストのうちの1つにおいてモーション・ベクトルがポイントする予測ブロックを位置付けてもよい。

10

【0122】

モーション補償ユニットはまた、ビデオ・スライスのピクチャ・ブロックを復号する際にビデオ復号器30によって使用するために、ブロック及びビデオ・スライスに関連する構文要素を生成してもよい。

【0123】

エントロピー・コーディング

【0124】

エントロピー符号化ユニット270は、例えば、エントロピー符号化アルゴリズム又は方式（例えば、可変長符号化（VLC）方式、コンテキスト適応VLC方式（CAVLC）、算術コーディング方式、バイナリ化、コンテキスト適応バイナリ算術コーディング（CABAC）、構文ベースのコンテキスト適応バイナリ算術コーディング（SBAC）、確率間隔パーティショニング・エントロピー（PIPE）コーディング、若しくは別のエントロピー・コーディング方法論又は技法）又はバイパス（非圧縮）を、量子化係数209、インター予測パラメータ、イントラ予測パラメータ、ループ・フィルタ・パラメータ及び／又は他の構文要素に適用して、例えば、符号化ビットストリーム21の形式で出力272を介して出力され得る符号化されたピクチャ・データ21を取得して、これにより、例えば、ビデオ復号器30は、復号のためのパラメータを受信及び使用することができる。符号化されたビットストリーム21は、ビデオ復号器30に送信されてもよいし、後の伝送又はビデオ復号器30による検索のためにメモリに記憶されてもよい。ビデオ符号化器20の他の構造的変形が、ビデオ・ストリームを符号化するために使用することができる。例えば、非変換ベースの符号化器20は、特定のブロック又はフレームに対して変換処理ユニット206なしで残差信号を直接量子化することができる。別の実装では、符号化器20は、単一ユニットに組み合わされた量子化ユニット208及び逆量子化ユニット210を有することができる。

20

30

【0125】

復号器及び復号方法

【0126】

図3は、本出願の技術を実装するように構成されたビデオ復号器30の例を示す。ビデオ復号器30は、例えば、符号化器20によって符号化されたピクチャ・データ21（例えば、符号化されたビットストリーム21）を受信して、復号されたピクチャ331を取得するように構成されている。符号化されたピクチャ・データ又はビットストリームは、符号化されたピクチャ・データを復号するための情報、例えば、符号化されたビデオ・スライスのピクチャ・ブロック及び関連する構文要素を表すデータを含む。

40

【0127】

図3の例では、復号器30は、エントロピー復号ユニット304と、逆量子化ユニット310と、逆変換処理ユニット312と、再構成ユニット314（例えば、合計器314）と、ループ・フィルタ320と、復号されたピクチャ・バッファ（DBP）330と、インター予測ユニット344と、イントラ予測ユニット354とを含む。インター予測ユ

50

ニット 344 は、モーション補償ユニットであってもよいし、又はこれを含んでもよい。ビデオ復号器 30 は、いくつかの例において、図 2 からのビデオ符号化器 100 に関して説明された符号化パスと概ね逆の復号パスを実行してもよい。

【0128】

符号化器 20 に関して説明したように、逆量子化ユニット 210、逆変換処理ユニット 212、再構成ユニット 214、ループ・フィルタ 220、復号されたピクチャ・バッファ (DRB) 230、インター予測ユニット 344、及びイントラ予測ユニット 354 も、ビデオ符号化器 20 の「内蔵復号器」を形成すると呼ばれる。したがって、逆量子化ユニット 310 は、逆量子化ユニット 110 と機能的に同一であってもよく、逆変換処理ユニット 312 は、逆変換処理ユニット 212 と機能的に同一であってもよく、再構成ユニット 314 は、再構成ユニット 214 と機能的に同一であってもよく、ループ・フィルタ 320 は、ループ・フィルタ 220 と機能的に同一であってもよく、復号されたピクチャ・バッファ 330 は、復号されたピクチャ・バッファ 230 と機能的に同一であってもよい。したがって、ビデオ復号器 30 のそれぞれのユニット及び機能には、ビデオ 20 符号化器のそれぞれのユニット及び機能に対して提供された説明が適用される。

【0129】

エントロピー復号

【0130】

エントロピー復号ユニット 304 は、ビットストリーム 21 (又は一般に、符号化されたピクチャ・データ 21) を解析し、例えば、符号化されたピクチャ・データ 21 へのエントロピー復号を実行して、例えば、量子化係数 309 及び／又は復号されたコーディング・パラメータ (図 3 には示さず)、例えば、インター予測パラメータ (例えば、参照ピクチャ・インデックス及びモーション・ベクトル)、イントラ予測パラメータ (例えば、イントラ予測モード又はインデックス)、変換パラメータ、量子化パラメータ、ループ・フィルタ・パラメータ、及び／又は他の構文要素のいずれか又は全てを取得するように構成されている。エントロピー復号ユニット 304 は、符号化器 20 のエントロピー符号化ユニット 270 に関して説明したように、符号化方式に対応する復号アルゴリズム又は方式を適用するように構成されてもよい。エントロピー復号ユニット 304 は、インター予測パラメータ、イントラ予測パラメータ及び／又は他の構文要素をモード選択ユニット 360 に提供し、他のパラメータを復号器 30 の他のユニットに提供するようにさらに構成されてもよい。ビデオ復号器 30 は、ビデオ・スライス・レベル及び／又はビデオ・ブロックレベルで構文要素を受信してもよい。

【0131】

逆量子化

【0132】

逆量子化ユニット 310 は、(例えば、エントロピー復号ユニット 304 による、例えば、解析及び／又は復号による) 符号化されたピクチャ・データ 21 からの量子化パラメータ (QP) (又は、一般には、逆量子化に関係する情報) 及び量子化係数を受信し、量子化パラメータに基づいて逆量子化を復号された量子化係数 309 に適用して、変換係数 311 とも呼ばれることがある量子化解除係数 311 を取得するように構成されてもよい。逆量子化プロセスは、量子化の程度、及び同様に適用されるべき逆量子化の程度を判定するために、ビデオ・スライスにおけるビデオ・ブロックごとにビデオ符号化器 20 によって判定された量子化パラメータを使用することを含んでもよい。

【0133】

逆変換

【0134】

逆変換処理ユニット 312 は、変換係数 311 とも呼ばれる量子化解除係数 311 を受信し、サンプル・ドメインにおいて再構成された残差ブロック 213 を取得するために、量子化解除係数 311 に変換を適用するように構成されてもよい。再構成された残差ブロック 213 は、変換ブロック 213 とも呼ばれることがある。変換は、逆変換、例えば、

逆 D C T、逆 D S T、逆整数変換、又は概念的に類似した逆変換プロセスであってもよい。逆変換処理ユニット 3 1 2 は、変換パラメータ又は対応する情報を符号化されたピクチャ・データ 2 1 から受信して（例えば、エントロピー復号ユニット 3 0 4 による、例えば、解析及び／又は復号によって）、量子化解除係数 3 1 1 に適用される変換を判定するようにさらに構成されてもよい。

【0 1 3 5】

再構成

【0 1 3 6】

再構成ユニット 3 1 4（例えば、加算器又は合計器 3 1 4）は、再構成された残差ブロック 3 1 3 を予測ブロック 3 6 5 に加算し、例えば、再構成された残差ブロック 3 1 3 のサンプル値及び予測ブロック 3 6 5 のサンプル値を追加することによって、サンプル・ドメインにおける再構成されたブロック 3 1 5 を取得するように構成されてもよい。

【0 1 3 7】

フィルタリング

【0 1 3 8】

ループ・フィルタ・ユニット 3 2 0（コーディング・ループ内又はコーディング・ループ後のいずれか）は、再構成されたブロック 3 1 5 をフィルタリングして、フィルタリングされたブロック 3 2 1 を取得して、例えば、画素遷移を平滑化するか、又は、そうでなければ、ビデオ品質を改善するように構成されている。ループ・フィルタ・ユニット 3 2 0 は、ブロック解除フィルタ、サンプル・アダプティブ・オフセット（S A O）フィルタ、若しくは 1 つ以上の他のフィルタ、例えばバイラテラル・フィルタ、アダプティブ・ループ・フィルタ（A L F）、鮮鋭化フィルタ、平滑化フィルタ、若しくは共同フィルタ、又はそれらの任意の組み合わせなどの 1 つ以上のループ・フィルタを含んでもよい。ループ・フィルタ・ユニット 3 2 0 は、図 3 ではループ内フィルタとして示されているが、他の構成では、ループ・フィルタ・ユニット 3 2 0 は、ポスト・ループ・フィルタとして実装されてもよい。

【0 1 3 9】

復号されたピクチャ・バッファ

【0 1 4 0】

次いで、ピクチャの復号されたビデオ・ブロック 3 2 1 は、復号されたピクチャ・バッファ 3 3 0 に記憶され、復号されたピクチャ・バッファ 3 3 0 は、他のピクチャ及び／又はそれぞれ出力表示するためのその後のモーション補償のための参照ピクチャとして記憶される。復号器 3 0 は、ユーザへの提示又は視認のために、例えば出力 3 1 2 を介して復号されたピクチャ 3 1 1 を出力するように構成されている。

【0 1 4 1】

予測

【0 1 4 2】

インター予測ユニット 3 4 4 は、インター予測ユニット 2 4 4（特に、モーション補償ユニット）と同一であってもよく、イントラ予測ユニット 3 5 4 は、機能において、インター予測ユニット 2 5 4 と同一であってもよく、（例えば、エントロピー復号ユニット 3 0 4 による、例えば、解析及び／又は復号による）復号されたピクチャ・データ 2 1 から受信されたパーティショニング及び／又は予測パラメータ又はそれぞれの情報に基づいて、分裂又はパーティショニング決定及び予測を実行する。モード選択ユニット 3 6 0 は、再構成されたピクチャ、ブロック又はそれぞれのサンプル（フィルタリングされた又はフィルタリングされていない）に基づいてブロックごとに予測（イントラ予測又はインター予測）を実行して、予測ブロック 3 6 5 を取得するように構成されてもよい。

【0 1 4 3】

ビデオ・スライスがイントラ・コーディング（I）されたスライスとしてコーディングされるときに、モード選択ユニット 3 6 0 のイントラ予測ユニット 3 5 4 は、シグナリングされたイントラ予測モードと、現在のピクチャの以前に復号されたブロックからのデー

10

20

30

40

50

タとに基づいて、現在のビデオ・スライスのピクチャ・ブロックのための予測ブロック 365 を生成するように構成されている。ビデオ・ピクチャがインターコーディング（例えば、B、又は P）されたスライスとしてコーディングされるときに、モード選択ユニット 360 のインター予測ユニット 344（例えば、モーション補償ユニット）は、エントロピー復号ユニット 304 から受信されたモーション・ベクトル及び他の構文要素に基づいて、現在のビデオ・スライスのビデオ・ブロックのための予測ブロック 365 を生成するように構成されている。インター予測のために、予測ブロックは、参照ピクチャ・リストのうちの 1 つ内の参照ピクチャのうちの 1 つから生成されてもよい。ビデオ復号器 30 は、DPB 330 に記憶された参照ピクチャに基づくデフォルトの構成技術を使用して、参照フレームリスト、リスト 0 及びリスト 1 を構成してもよい。

10

【0144】

モード選択ユニット 360 は、モーション・ベクトル及び他の構文要素を解析することによって、現在のビデオ・スライスのビデオ・ブロックのための予測情報を判定するように構成されており、予測情報を使用して、復号される現在のビデオ・ブロックの予測ブロックを生成する。例えば、モード選択ユニット 360 は、受信された構文要素のいくつかを使用して、ビデオ・スライスのビデオ・ブロックをコーディングするために使用される予測モード（例えば、イントラ又はインター予測）、インター予測スライス・タイプ（例えば、B スライス、P スライス、又は GPB スライス）、スライスのための参照ピクチャ・リストのうちの 1 つ以上の構成情報、スライスの各インター符号化されたビデオ・ブロックのためのモーション・ベクトル、スライスの各インターコーディングされたビデオ・ブロックのためのインター予測ステータス、及び現在のビデオ・スライスにおけるビデオ・ブロックを復号するための他の情報を判定する。

20

【0145】

ビデオ復号器 30 の他の変形を使用して、符号化されたピクチャ・データ 21 を復号することができる。例えば、復号器 30 は、ループ・フィルタリング・ユニット 320 なしで出力ビデオ・ストリームを生成することができる。例えば、非変換ベースの復号器 30 は、特定のブロック又はフレームに対して、逆変換処理ユニット 312 なしで残差信号を直接逆量子化することができる。別の実装では、ビデオ復号器 30 は、単一のユニットに組み合わせられた逆量子化ユニット 310 及び逆変換処理ユニット 312 を有することができる。

30

【0146】

符号化器 20 及び復号器 30 では、現在のステップの処理結果をさらに処理し、次のステップに出力してもよいと理解されたい。例えば、補間フィルタリング、モーション・ベクトル導出、又はループ・フィルタリングの後に、クリップ又はシフトなどのさらなる演算を、補間フィルタリング、モーション・ベクトル導出、又はループ・フィルタリングの処理結果に対して実行してもよい。

【0147】

さらなる演算は、現在のブロックの導出されたモーション・ベクトル（アフィン・モードの制御点モーション・ベクトル、アフィン、平面、ATMVP モードのサブブロック・モーション・ベクトル、時間的モーション・ベクトルなどを含むが、これらに限定されない）に適用され得ると留意されたい。例えば、モーション・ベクトルの値は、その表現ビットにしたがって予め定義された範囲に制約される。モーション・ベクトルの表現ビットが bitDepth である場合、範囲は、 $-2^{(\text{bitDepth}-1)} \sim 2^{(\text{bitDepth}-1)} - 1$ であり、ここで、「 $\wedge$ 」はべき乗を意味する。例えば、 bitDepth が 16 にセットされる場合、範囲は $-32768 \sim 32767$ であり、 bitDepth が 18 にセットされる場合、 $-131072 \sim 131071$ である。ここでは、モーション・ベクトルを制約するための 2 つの方法を提供する。

40

【0148】

方法 1：フロー演算によりオーバーフロー MSB（最上位ビット）を除去する。

【数 4】

50

$$ux = (mvx + 2^{\text{bitDepth}}) \% 2^{\text{bitDepth}} \quad (1)$$

$$mvx = (ux \geq 2^{\text{bitDepth}-1}) ? (ux - 2^{\text{bitDepth}}) : ux \quad (2)$$

$$uy = (mvy + 2^{\text{bitDepth}}) \% 2^{\text{bitDepth}} \quad (3)$$

$$mvy = (uy \geq 2^{\text{bitDepth}-1}) ? (uy - 2^{\text{bitDepth}}) : uy \quad (4)$$

10

たとえば、 mvx の値が -32769 である場合、式 (1) 及び (2) を適用した後、結果の値は 32767 である。コンピュータ・システムでは、10進数は2の補数として記憶される。 -32769 の2の補数は $1, 0111, 1111, 1111$ (17ビット) であり、MSBが破棄されるため、結果として得られる2つの補数は $0111, 1111, 1111, 1111$ (10進数は 32767) となり、これは、式 (1) と (2) を適用した出力と同じである。

【数5】

$$ux = (mvpx + mvdx + 2^{\text{bitDepth}}) \% 2^{\text{bitDepth}} \quad (5)$$

$$mvx = (ux \geq 2^{\text{bitDepth}-1}) ? (ux - 2^{\text{bitDepth}}) : ux \quad (6)$$

20

$$uy = (mvpy + mvdy + 2^{\text{bitDepth}}) \% 2^{\text{bitDepth}} \quad (7)$$

$$mvy = (uy \geq 2^{\text{bitDepth}-1}) ? (uy - 2^{\text{bitDepth}}) : uy \quad (8)$$

演算は、式 (5) ~ (8) に示すように、 mvp と mvd の和の間に適用されてもよい。

【0149】

方法2：値をクリッピングすることによりオーバーフローMSBを除去する。

30

【数6】

$$vx = \text{Clip3}(-2^{\text{bitDepth}-1}, 2^{\text{bitDepth}-1}-1, vx)$$

$$vy = \text{Clip3}(-2^{\text{bitDepth}-1}, 2^{\text{bitDepth}-1}-1, vy)$$

ここで、機能 Clip3 の定義は以下のようである。

【数7】

$$\text{Clip3}(x, y, z) = \begin{cases} x & ; \quad z < x \\ y & ; \quad z > y \\ z & ; \quad \text{otherwise} \end{cases}$$

40

【0150】

図4は、本開示の一実施形態によるビデオ・コーディング・デバイス400の概略図である。ビデオ・コーディング・デバイス400は、本明細書で説明される開示される実施形態を実装するのに好適である。一実施形態では、ビデオ・コーディング・デバイス400は、図1Aのビデオ復号器30のような復号器、又は図1Aのビデオ符号化器20のような符号器であってもよい。

【0151】

ビデオ・コーディング・デバイス400は、データを受信するための入口ポート410

50

(又は入力ポート410)及び受信機ユニット(Rx)420と、データを処理するためのプロセッサ、論理ユニット、又は中央処理ユニット(CPU)430と、データを送信するための送信機ユニット(Tx)440及び出口ポート450(又は出力ポート450)と、データを記憶するためのメモリ460を含む。ビデオ・コーディング・デバイス400はまた、光信号又は電気信号の出入りのために、入力ポート410、受信機ユニット420、送信機ユニット440、及び出口ポート450に結合された光から電気(OE)コンポーネント及び電気から光(EO)コンポーネントを含んでもよい。

【0152】

プロセッサ430は、ハードウェア及びソフトウェアによって実装される。プロセッサ430は、1つ以上のCPUチップ、コア(例えば、マルチコアプロセッサ)、FPGA、ASIC、及びDSPとして実装されてもよい。プロセッサ430は、入力ポート410、受信機ユニット420、送信機ユニット440、出口ポート450、及びメモリ460と通信している。プロセッサ430は、コーディング・モジュール470を含む。コーディング・モジュール470は、上述の開示された実施形態を実装する。例えば、コーディング・モジュール470は、種々のコーディング演算を実装、処理、準備、又は提供する。したがって、符号化モジュール470を含めることは、ビデオ・コーディング・デバイス400の機能性への実質的な改善を提供し、ビデオ・コーディング・デバイス400の異なる状態への変換をもたらす。代替的には、コーディング・モジュール470は、メモリ460に記憶され、プロセッサ430によって実行される命令として実装される。

【0153】

メモリ460は、1つ又は複数のディスク、テープ・ドライブ、及びソリッドステート・ドライブを含んでもよく、オーバ・フロー・データ記憶デバイスとして使用され、そのようなプログラムが実行のために選択されたときにプログラムを記憶し、プログラム実行中に読み出された命令及びデータを記憶してもよい。メモリ460は、例えば、揮発性及び／又は不揮発性であってもよく、リード・オンリー・メモリ(ROM)、ランダム・アクセス・メモリ(RAM)、ターナリー・コンテンツ・アドレス指定可能メモリ(TCAM)、及び／又はスタティック・ランダム・アクセス・メモリ(RSAM)であってもよい。

【0154】

図5は、例示的な実施形態による図1からのソース・デバイス12及び宛先デバイス14のいずれか又は両方として使用されてもよい装置500の簡略ブロック図である。

【0155】

装置500内のプロセッサ502は、中央処理ユニットとすることができる。代替的には、プロセッサ502は、現在存在する、又は今後開発される情報を操作又は処理することが可能な、任意の他のタイプのデバイス又は複数のデバイスとすることができる。開示された実装は、示すように単一のプロセッサ、例えば、プロセッサ502を用いて実施することができるが、複数のプロセッサを使用して、速度及び効率における利点を達成することができる。装置500内のメモリ504は、実装におけるリード・オンリー・メモリ(ROM)デバイス又はランダム・アクセス・メモリ(RAM)デバイスとすることができる。メモリ504として、任意の他の好適なタイプの記憶デバイスを使用することができる。メモリ504は、バス512を使用して、プロセッサ502によってアクセスされるコード及びデータ506を含むことができる。メモリ504は、オペレーティング・システム508及びアプリケーション・プログラム510をさらに含むことができ、アプリケーション・プログラム510は、プロセッサ502が本明細書で説明される方法を実行することを許可する少なくとも1つのプログラムを含む。例えば、アプリケーション・プログラム510は、アプリケーション1～Nを含むことができ、これは、本明細書で説明される方法を実行するビデオ・コーディング・アプリケーションをさらに含む。装置500はまた、ディスプレイ518などの1つ以上の出力デバイスをさらに含むことができる。ディスプレイ518は、一例では、ディスプレイを、タッチ入力感知するように動作可能なタッチ・センシティブ要素と組み合わせるタッチ・センシティブ・ディスプレイで

10

20

30

40

50

あってもよい。ディスプレイ 518 は、バス 512 を介してプロセッサ 502 に接続することができる。

【0156】

ここでは、単一のバスとして示されているが、装置 500 のバス 512 は、複数のバスから構成することができる。さらに、二次ストレージ 514 は、装置 500 の他のコンポーネントに直接結合することができ、又はネットワークを介してアクセスすることができ、メモリカードなどの単一の一体化ユニット又は複数のメモリカードなどの複数のユニットを含むことができる。したがって、装置 500 は、広範な構成で実装することができる。

【0157】

図 18 は、コンテンツ配信サービスを実現するためのコンテンツ供給システム 3100 を示すブロック図である。このコンテンツ供給システム 3100 は、捕捉デバイス 3102、端末デバイス 3106 を含み、任意選択で、ディスプレイ 3126 を含む。捕捉デバイス 3102 は、通信リンク 3104 を介して端末デバイス 3106 と通信する。通信リンクは、上述の通信チャネル 13 を含んでもよい。通信リンク 3104 は、WIFI、イーサネット、ケーブル、無線（3G／4G／5G）、USB、又はそれらの任意の種類の組み合わせなどを含むが、これらに限定されない。

【0158】

捕捉デバイス 3102 は、データを生成し、上記の実施形態に示す符号化方法によってデータを符号化してもよい。代替的には、捕捉デバイス 3102 は、ストリーミングサーバ（図示せず）にデータを配信することができ、サーバは、データを符号化し、符号化されたデータを端末デバイス 3106 に送信する。捕捉デバイス 3102 は、カメラ、スマートフォン又はパッド、コンピュータ若しくはラップトップ、ビデオ会議システム、PDA、車載デバイス、又はそれらのいずれかの組み合わせなどを含むが、これらに限定されない。例えば、捕捉デバイス 3102 は、上述のようにソース・デバイス 12 を含んでもよい。データがビデオを含む場合に、捕捉デバイス 3102 に含まれるビデオ符号化器 20 は、実際にビデオ符号化処理を実行することができる。データがオーディオ（例えば、音声）を含む場合、捕捉デバイス 3102 に含まれるオーディオ符号化器は、実際にオーディオ符号化処理を実行することができる。いくつかの実践的なシナリオでは、捕捉デバイス 3102 は、符号化されたビデオ及びオーディオ・データを、それらを一緒に多重化することによって配布する。他の実践的なシナリオ、例えば、ビデオ会議システムにおいては、符号化されたオーディオ・データ及び符号化されたビデオ・データは多重化されない。捕捉デバイス 3102 は、符号化されたオーディオ・データ及び符号化されたビデオ・データを、端末デバイス 3106 に別々に配信する。

【0159】

コンテンツ供給システム 3100 では、端末デバイス 310 は、符号化されたデータを受信及び再生する。端末デバイス 3106 は、スマートフォン又は Pad 3108、コンピュータ若しくはラップトップ 3110、ネットワーク・ビデオ・レコーダ（NVR）／デジタル・ビデオ・レコーダ（DVR） 3112、TV 3114、セット・トップ・ボックス 3116、ビデオ会議システム 3118、ビデオ監視システム 3120、パーソナル・デジタル・アシスタント（PDA） 3122、車載デバイス 3124、又はこれらのいずれかの組み合わせなどの、データ受信及び復元能力を有し、前述の符号化されたデータを復号することが可能なデバイスとすることができる。例えば、捕捉デバイス 3106 は、上述のようにソース・デバイス 14 を含んでもよい。符号化されたデータがビデオを含む場合に、端末デバイスに含まれるビデオ復号器 30 は、ビデオ復号を実行するように優先される。符号化されたデータがオーディオを含む場合に、端末デバイスに含まれるオーディオ復号器は、オーディオ復号処理を実行するように優先される。

【0160】

ディスプレイを有する端末デバイス、例えば、スマートフォン又はパッド 3108、コンピュータ又はラップトップ 3110、ネットワーク・ビデオ・レコーダ（NVR）／デジタル・ビデオ・レコーダ（DVR） 3112、TV 3114、パーソナル・デジタル

10

20

30

40

50

・アシスタント 3 1 2 2、又は車載デバイス 3 1 2 4 では、端末デバイスは、復号されたデータをそのディスプレイに供給することができる。ディスプレイを搭載しないデバイス、例えば、S T B 3 1 1 6、ビデオ会議システム 3 1 1 8、又はビデオ監視システム 3 1 2 0 では、外部ディスプレイ 3 1 2 6 がそこに接触されて、復号されたデータを受信し示す。

【0 1 6 1】

このシステムの各デバイスが符号化又は復号を行う場合に、前述の実施形態に示すように、ピクチャ符号化デバイス又はピクチャ復号デバイスを使用することができる。図 1 9 は、端末デバイス 3 1 0 6 の一例の構成を示す図である。端末デバイス 3 1 0 6 が、捕捉デバイス 3 1 0 2 からストリームを受信した後、プロトコル進行ユニット 3 2 0 2 は、ストリームの伝送プロトコルを分析する。このプロトコルは、リアルタイム・ストリーミング・プロトコル (R T S P)、ハイパー・テキスト・トランスファー・プロトコル (H T T P)、H T T P ライブ・ストリーミング・プロトコル (H L S)、M P E G - D A S H、リアルタイム・トランスポート・プロトコル (R T P)、リアルタイム・メッセージング・プロトコル (R T M P)、又はそれらの任意の種類の組み合わせなどを含むが、これらに限定されない。プロトコル進行ユニット 3 2 0 2 がストリームを処理した後に、ストリームファイルは、生成される。ファイルは、逆多重化ユニット 3 2 0 4 に出力される。逆多重化ユニット 3 2 0 4 は、多重化されたデータを符号化されたオーディオ・データと符号化されたビデオ・データとに分離することができる。上述したように、いくつかの実際的なシナリオ、例えば、ビデオ会議システムにおいては、符号化されたオーディオ・データ及び符号化されたビデオ・データは多重化されない。この状況では、符号化データは、逆多重化部 3 2 0 4 を介することなく、ビデオ復号器 3 2 0 6 及びオーディオ復号器 3 2 0 8 に送信される。

【0 1 6 2】

逆多重化処理を介して、ビデオ・エレメンタリー・ストリーム (E S)、オーディオ E S、及び任意選択で字幕が生成される。前述の実施形態で説明されたビデオ復号器 3 0 を含むビデオ復号器 3 2 0 6 は、上述の実施形態に示す復号方法によってビデオ E S を復号して、ビデオ・フレームを生成し、このデータを同期ユニット 3 2 1 2 に送信する。オーディオ復号器 3 2 0 8 は、オーディオ E S を復号して、オーディオ・フレームを生成し、このデータを同期ユニット 3 2 1 2 に送信する。代替的には、ビデオ・フレームは、同期ユニット 3 2 1 2 に供給する前に、バッファ (図 Y には示さず) に記憶させてもよい。代替的には、オーディオ・フレームは、同期ユニット 3 2 1 2 に供給する前に、バッファ (図 Y には示さず) に記憶させてもよい。

【0 1 6 3】

同期ユニット 3 2 1 2 は、ビデオ・フレームとオーディオ・フレームを同期させ、ビデオ／オーディオ・ディスプレイ 3 2 1 4 にビデオ／オーディオを供給する。例えば、同期ユニット 3 2 1 2 は、ビデオ情報及びオーディオ情報の提示を同期させる。情報は、符号化されたオーディオ及びビジュアル・データの提示に関するタイム・スタンプ及びデータストリーム自体の送達に関するタイム・スタンプを使用して、構文でコーディングすることができる。

【0 1 6 4】

字幕がストリームに含まれる場合、字幕復号器 3 2 1 0 は、字幕を復号し、それをビデオ・フレーム及びオーディオ・フレームと同期させ、ビデオ／オーディオ／字幕をビデオ／オーディオ／字幕ディスプレイ 3 2 1 6 に供給する。

【0 1 6 5】

I T U - T H . 2 6 5 によるマージ候補リスト構成に関する例では、マージ候補リストが、以下の候補に基づいて構成される。

1. 5 つの空間的隣接ブロックから導出される最大 4 つの空間的候補 M V P
2. 2 つの時間的に共位置にあるブロックから導出された 1 つの時間的候補、
3. 組み合わされたバイ予測候補を含む追加候補

4. ゼロ・モーション・ベクトル候補

【0166】

空間的候補

【0167】

空間的隣接ブロックのモーション情報は、最初に、モーション情報候補としてマージ候補リストに追加される（一例では、マージ候補リストは、最初のモーション・ベクトルがマージ候補リストに追加される前に、空のリストであってもよい）。ここで、マージリストに挿入されると考慮される隣接ブロックが図6bに示される。インター予測ブロックマージの場合、A1、B1、B0、A0、B2を順次その順序でチェックすることにより、最大4つの候補がマージリストに挿入される。

10

【0168】

モーション情報は、1つ又は2つの参照ピクチャ・リストが使用されるかどうかの情報、ならびに各参照ピクチャ・リストに対する参照インデックス及びモーション・ベクトルを含む全てのモーション・データを含んでもよい。

【0169】

一例では、隣接ブロックが利用可能であり、モーション情報を含んでいるかどうかをチェックした後、隣接ブロックのすべてのモーション・データをモーション情報候補として取り出す前に、いくつかの追加の冗長性チェックが実行される。

【0170】

これらの冗長性チェックは、2つの異なる目的のために2つのカテゴリに分割され得る。カテゴリ1は、リスト内に冗長なモーション・データを持つ候補を持つことを避ける。カテゴリ2は、冗長な構文を生成する他の手段によって表現され得る2つのパーティションをマージすることを防止する。

20

【0171】

時間的候補

【0172】

図6aは、時間的モーション情報候補が検索されるブロックの座標を示した。共位置にあるブロックは、現在のブロックと同じx、y座標を有するが、異なるピクチャ（参照ピクチャのうちの1つ）にあるブロックである。時間的モーション情報候補は、リストが満杯でない場合、マージリストに追加される（一例では、マージリスト内の候補の量が閾値より小さいときに、マージリストが満杯でなく、例えば、閾値は、4、5、6などであってもよい）。

30

【0173】

生成された候補

【0174】

空間的及び時間的運動情報候補の挿入後、マージリストが依然として満杯でない場合、生成された候補がリストを充填するように追加される。リストサイズは、シーケンスパラメータセットにおいて示され、コーディングされたビデオ・シーケンス全体を通して固定される。

【0175】

40

ITU-T H. 265及びVVCにおけるマージリスト構成プロセスは、モーション情報候補のリストを出力する。VVCのマージリスト構成プロセスは、文書JVET-L1001_v2 Versatile Video Coding (Draft 3)の「8.3.2.2 Derivation process for luma motion vectors for merge mode」のセクションに説明されており、これは、<http://phenix.it-sudparis.eu/jvet/>で公衆に利用可能である。用語モーション情報は、モーション補償予測プロセスを行うために必要なモーション・データを指す。

モーション情報は通常、以下の情報を指す。

・ブロックがユニ予測、バイ予測のどちらを適用するか

50

- ・予測に使用される参照ピクチャのID（ブロックがバイ予測を適用する場合、2つのID）。
- ・モーション・ベクトル（ブロックがバイ予測される場合、2つのモーション・ベクトル）
- ・追加情報

【0176】

VVC及びH.265では、マージリスト構成の出力である候補のリストは、N個の候補モーション情報を含む。数Nは、典型的にはビットストリームに含まれ、5、6などの正の整数であってもよい。構成されたマージリストに含まれる候補は、ユニ予測情報又はバイ予測情報を含むかもしれない。これは、マージリストから選択される候補がバイ予測動作を示すかもしれないことを意味する。

【0177】

バイ予測

【0178】

インター予測の特別なモードは、「バイ予測」と呼ばれ、ブロックを予測するために2つのモーション・ベクトルが使用される。モーション・ベクトルは、同じ又は異なる参照ピクチャを指し示すことができ、参照ピクチャは、参照ピクチャ・リストID及び参照ピクチャ・インデックスによって示され得る。例えば、第1のモーション・ベクトルは、参照ピクチャ・リストL0内の最初のピクチャを指し示すかも知れないし、第2のモーション・ベクトルは、参照ピクチャ・リストL1内の最初のピクチャを指し示すかもしれない。2つの参照ピクチャ・リスト（例えば、L0及びL1）が維持されるかも知れず、第1のモーション・ベクトルによって指し示されるピクチャがリストL0から選択され、第2のモーション・ベクトルによって指し締めされるピクチャがリストL1から選択される。

【0179】

一例では、モーション情報がバイ予測を示す場合、モーション情報は2つの部分を含む。

- ・L0部：参照ピクチャ・リストL0の中のエントリを指し示すモーション・ベクトルと参照ピクチャ・インデックス。
- ・L1部：参照ピクチャ・リストL1の中のエントリを指し示すモーション・ベクトルと参照ピクチャ・インデックス。

【0180】

ピクチャ・オーダ・カウント（POC）：各ピクチャに関連付けられた変数は、CVS（Coded Video Sequence）内のすべてのピクチャの中で関連するピクチャを一意に識別し、関連するピクチャが復号されたピクチャ・バッファから出力されるときに、復号されたピクチャ・バッファから出力される同じCVS内の他のピクチャの出力オーダ位置に対する出力オーダの関連するピクチャの位置を示す。

【0181】

参照ピクチャ・リストL0及びL1の各々は、各々がPOCで識別される1つ以上の参照ピクチャを含むかもしれない。各参照インデックス及びPOC値との関連は、ビットストリームにおいてシグナリングされるかもしれない。例として、L0及びL1参照ピクチャ・リストは、以下の参照ピクチャを含むかもしれない。

【表2】

参照ピクチャリスト	参照インデックス	POC
L0	0	12
L0	1	13
L1	0	13
L1	1	14

【0182】

上記の例では、参照ピクチャ・リストL1の最初のエントリ（参照インデックス0で示される）は、POC値13の参照ピクチャである。参照ピクチャ・リストL1の2番目のエントリ（参照インデックス1で示される）は、POC値14の参照ピクチャである。

【0183】

三角形予測モード

【0184】

三角形予測モードの概念は、モーション補償予測のための三角形パーティションを開示することである。図7に示すように、対角又は逆対角方向のいずれかにおいて2つの三角形予測単位がCUに使用される。CUにおける各三角形予測ユニットは、ユニ予測候補リストから導出されるユニ予測モーション・ベクトルと基準フレームインデックスを用いて、インター予測される。各三角形予測ユニットに関連するサンプルが動き補償又はピクチャ内予測によって予測された後、適応重み付けプロセスが対角エッジに対して実行される。次いで、変換と量子化プロセスがCUに適用される。このモードは、スキップ・モード及びマージ・モードに適用されることに留意する。

【0185】

三角形予測モードでは、ブロックは2つの三角形部分に分裂され（図7のように）、各部分は1つのモーション・ベクトルを使用して予測され得る。1つの三角部（PU1で示される）を予測するために使用されるモーション・ベクトルは、他の三角部（PU2で示される）を予測するために使用されるモーション・ベクトルとは異なってもよい。一例では、三角形予測モードを実行する複雑さを低減するために、各部分が単一のモーション・ベクトル（ユニ予測）を使用して予測され得ることに留意する。言い換えれば、PU1及びPU2は、2つのモーション・ベクトルを含むバイ予測を使用して予測されなくてもよい。

【0186】

サブブロック予測モード

【0187】

三角形予測モードは、サブブロック予測の特殊なケースである。サブブロック予測では、ブロックは、2つのブロックに分割される。上記の例では、2つのブロック分割方向（45度及び135度パーティション）が示されており、他のパーティション角度及びパーティション比も可能である（図8の例）。

【0188】

いくつかの例では、ブロックは2つの部分に分裂され、各部分はユニ予測によって適用される。サブブロック予測は、一般化バージョンの三角形予測を表す。

【0189】

三角形予測モードは、ブロックが2つのブロックに分割されるサブブロック予測モードの特殊な場合である。上記の例では、2つのブロック分割方向（45度及び135度パーティション）が示されている。他のパーティション角度及びパーティション比も可能である（例えば、図8の例）。

【0190】

いくつかの例では、ブロックは2つのサブブロックに分裂され、各部分（サブブロック）はユニ予測で予測される。

【0191】

一例では、サブブロック・パーティション・モードに従って、予測サンプルを取得するために以下のステップが適用される。

・ステップ1：コーディング・ブロックは、幾何学的モデルに従って、2つのサブブロックに分割されると考慮される。このモデルは、図9～12に例示されているように、分離線（例えば、直線）によるブロックの分裂をもたらす。このステップでは、幾何学的モデルに従って、コーディング・ブロックにおけるサンプルは、2つのサブブロックに位置すると見なされる。サブブロックA又はサブブロックBは、現在のコーディング・ブ

10

20

30

40

50

ックにおけるサンプルの一部（すべてではない）を含む。サブブロック A 又はサブブロック B の概念は、分離線に関するパラメータによって示されてもよい。

・ステップ 2：第 1 のサブブロックのための第 1 の予測モードと第 2 のサブブロックのための第 2 の予測モードを取得する。一例では、第 1 の予測モードは第 2 の予測モードと同一ではない。一例では、予測モード（第 1 の予測モード又は第 2 の予測モード）は、インター予測モードであってもよく、相互予測モードのための情報は、参照ピクチャ・インデックス及びモーション・ベクトルを含んでもよい。別の例では、予測モードは、イントラ予測モードであってもよく、イントラ予測モードのための情報は、イントラ予測モード・インデックスを含んでもよい。

・ステップ 3：第 1 予測モードと第 2 予測モードを使用して、第 1 の予測値と第 2 の予測値をそれぞれ生成する。

・ステップ 4：第 1 の予測値と第 2 の予測値の組み合わせに従って、予測サンプルの組み合わせ値を取得する。一例として、ステップ 1 では、コーディング・ブロックは、様々な方法で 2 つのサブブロックに分割されると考慮される。

【0192】

図 9 は、コーディング・ブロックのパーティションに関する例を示し、分離線 1250 は、ブロックを 2 つのサブブロックに分割する。線 1250 を説明するために、2 つのパラメータがシグナリングされ、一方のパラメータは角度 α_{1210} であり、他方のパラメータは距離 d_{1230} である。

【0193】

いくつかの実施形態では、図 9 に示すように、角度は、x 軸と分離線との間で測定され、距離は、分離線に垂直で、現在のブロックの中心を通過するベクトルの長さによって測定される。別の例では、図 10 は、分離線を表す別の方法を示しており、角度と距離の例は、図 9 に示されている例とは異なる。いくつかの例では、ステップ 4 において、ステップ 1 で開示された分割を、第 1 の予測値と第 2 の予測値との組み合わせに使用して、最終的な予測値を取得する。一例では、任意のアーチファクト（分離線に沿ったエッジ状又はジャグ状の外観）を除去するために、混合動作がステップ 4 において適用される。混合動作は、分離ラインに沿ったフィルタリング動作として説明されてもよい。

【0194】

符号化器側において、分離線（線を定義するパラメータ、例えば角度及び距離）は、レート歪みベースのコスト関数に基づいて判定される。判定された線パラメータは、ビットストリームに符号化される。復号器側において、線パラメータはビットストリームに従って復号（取得）される。

【0195】

一例では、輝度コンポーネントと 2 つの色度コンポーネントを含む 3 つのビデオ・チャネルの場合、各チャネルに対して第 1 の予測と第 2 の予測が生成される。

【0196】

コーディング・ブロックを 2 つのサブブロックに分割する可能性が多いため、分割のシグナリング（コーディング）にはビット数が多すぎる。角度値と距離値は、ビットストリームにシグナリングされるにはあまりにも多くのサイド情報を必要とする多くの異なる値を有するため、コーディング効率を改善するために、量子化スキームが角度と距離サイド情報に適用される。

【0197】

一例では、量子化された角度パラメータ α_{Idx} 及び量子化された距離パラメータ d_{Idx} が、パーティショニング・パラメータのためにシグナリングされる。

【0198】

一例では、量子化スキームにおいて、角度値及び距離値は、以下に従って、線形一様量子化器によって量子化され得る。

【数 8】

10

20

30

40

50

$$\text{alphaIdx} = \text{Round}(\text{alpha} / \Delta\text{alpha}),$$

$$\text{distanceIdx} = \text{Round}(\text{dist} / \Delta\text{dist}),$$

ここで、変数 Δalpha 及び Δdist は、分離線の角度パラメータ及び距離パラメータの量子化ステップ・サイズを表す。したがって、角度 alpha 及び距離 dist は、以下のように、ビットストリームから取得された alphaIdx 及び distanceIdx の値を使用して、再構成されてもよい。

【数 9】

$$\text{alpha} = \text{alphaIdx} * \Delta\text{alpha},$$

$$\text{dist} = \text{distanceIdx} * \Delta\text{dist}.$$

一例では、角度量子化ステップ・サイズが $\Delta\text{alpha} = 11.25^\circ$ に等しい場合、 alphaIdx の値は、ユークリッド平面内で許容される角度をカバーするために、0 ~ 31 (0 及び 31 を含む) の範囲であり得る。別の例では、角度量子化ステップ・サイズの値は、 $\Delta\text{alpha} = 22.5^\circ$ に等しく、それによって、 alphaIdx の値は、0 ~ 15 (0 及び 15 を含む) の範囲であり得る。いくつかの例では、角度に依存して距離情報を量子化する方法が開示されており、角度を量子化するために線形一様量子化器が使用される場合、与えられたコーディング（復号又は符号化）ブロックについて、 Δalpha の値が一定であることを意味する。開示される量子化スキームの目標は、異なるサイズ及びアスペクト比を有することができる所与のコーディング・ブロックに対して、許容される分離線を均等に分布させることである。量子化された角度及び距離パラメータの均等に分布した空間は、非矩形コンポーネントのコーディング効率を改善し、分離線パラメータのコンパクトな表現を可能にする。

【0199】

いくつかの例では、直線によって矩形コーディング・ブロックをパーティショニングする方法が開示されており、ここで、直線は、量子化された角度及び量子化された距離の値を表す一対のパラメータによってパラメータ化され、量子化された距離の値は、コーディング・ブロックの角度の値及びアスペクト比に依存して、量子化プロセスに従って導出される。

【0200】

一例では、距離は、 distanceIdx の所与の値範囲が満たされるように、例えば、0 ~ 3 の値範囲 (0 及び 3 を含む) で量子化されてもよい。別の例では、距離は、所与の angleIdx 及び distanceIdx に対する分離線が均等に分布し、分離線が所与のコーディング・ブロックの領域の外側に存在しないように、所与のブロックに対して量子化されてもよい。

【0201】

最初のステップでは、最大距離 p_{max} を角度に依存して導出され得、距離 $0 < \text{dist} < p_{\text{max}}$ のすべての分離線がコーディング・ブロックに閉じ込められるようにする（例えば、コーディング・ブロックの境界と交差する）。これは、サイズ 16×4 の輝度サンプルのコーディング・ブロックに対して図 15 に例示されている。

【0202】

一例では、最大距離 p_{max} が、角度 alpha_R とコーディング・ブロックのサイズに依存する関数として、以下に従って導出され得る。

【数 10】

10

20

30

40

50

$$\rho_{\max}(\alpha R, w, h) = \cos(\alpha R) \left(\frac{h}{2 \tan\left(\frac{\pi}{2} - \alpha R\right)} + \frac{w}{2} \right), 0 \leq \alpha R < \frac{\pi}{2},$$

ここで、 αR は、放射ユニットに関して α として前もって示された角度であり、 w は、幅であり、 h は、輝度サンプルに関してブロックの高さである。

この例では、角度に依存の距離量子化ステップ・サイズ $\Delta \text{dist}(\alpha)$ の値は、以下のように導出され得る。

【数 1 1】

$$\Delta \text{dist}(\alpha R, w, h) = \rho_{\max}(\alpha R, w, h) / N,$$

10

ここで、 $N-1$ は、 distanceIdx の最大値である。これは、図 16において、 $\Delta \text{angle} = 22.5^\circ$ 、 $N = 4$ の値のものが例示されている。

【0 2 0 3】

別の例では、角度に依存の距離量子化ステップ・サイズの値は、以下のように導出されてもよい。

【数 1 2】

$$\Delta \text{dist}(\alpha R, w, h) = (\rho_{\max}(\alpha R, w, h) - \rho_{\text{th}}) / N,$$

20

ここで、 ρ_{th} は、コーディング・ブロックための分離ライン間の間隔を修正するために使用されるパラメータである。例では、 ρ_{th} の値は、 $\rho_{\text{th}} = 1.5$ にセットされる。

【0 2 0 4】

別の例では、最大距離 $\rho_{\max}$ は、コーディング・ブロックの角度及びサイズに依存する関数として、以下に従って導出され得る。

【数 1 3】

$$\rho_{\max}(\alpha R, w, h) = \frac{1}{2 \sqrt{\left(\frac{\cos(\alpha R)}{w}\right)^2 + \left(\frac{\sin(\alpha R)}{h}\right)^2}}, 0 \leq \alpha R < \frac{\pi}{2},$$

30

ここで、 αR は、放射ユニットに関する角度であり、 w は、幅であり、 h は、輝度サンプルに関してブロックの高さである。これは、図 17において、 $\Delta \text{angle} = 22.5^\circ$ 、 $N = 4$ の値のものが例示されている。

【0 2 0 5】

一例では、 Δdist の値は、角度を表す値に依存し、幅を表すパラメータ及び高さを表すパラメータは、符号化又は復号処理の間に Δdist の繰り返される計算を避けるために、事前に計算されたルックアップ・テーブルに記憶されてもよい。

【0 2 0 6】

一例では、 Δdist の値は、整数演算を使用する目的として、以下のようにスケーリング及び丸められ得る。

【数 1 4】

$$\text{stepD} = \text{Round}(\Delta \text{dist} * 2^{\text{prec}}),$$

40

ここで、 stepD は、線形スケーリングされた距離ステップ・サイズを表し、 prec は、スケーリング・プロセスの精度を調整するために使用される変数である。一例では、 prec の値は、 $\text{prec} = 7$ にセットされる。

【0 2 0 7】

一例では、それは、コーディング・ブロックの幅及び高さに依存する、 $w h R a t i o$

50

を示すアスペクト比に基づいて $stepD$ の予め計算された値をさらに記憶することである。さらに、 $stepD$ の予め計算された値は、ユークリッド平面の第1象限の角度（例えば、 $0 \leq angleN * \Delta alpha \leq 90^\circ$ ）に関するインデックス値である（正規化された）角度値 $angleN$ に基づいて記憶される。そのようなルックアップ・テーブルが適用された例を図13に示す。一例では、以下のステップを適用して、コーディング・ブロックのサンプルに対する予測値を取得する。

【0208】

ステップ1：現在のコーディング・ブロック（復号ブロック又は符号化ブロック）内のサンプルに対して、サンプル距離（ $sample_dist$ ）が計算される。

【0209】

いくつかの例では、サンプル距離は、分離線（分離線は、コーディング・ブロックが2つのサブブロックに分割されることを示すために使用される）へのサンプルの水平距離若しくは垂直距離、又は垂直距離と水平距離の組み合わせを表してもよい。サンプルは、コーディング・ブロックの左上のサンプルに関する座標（ x, y ）によって表される。サンプル座標と $sample_dist$ は、図11と図12に例示される。サブブロックは、必ずしも矩形ではなく、三角形又は台形の形状を有するかもしれない。一例では、第1のパラメータは量子化された角度値（ $angleIdx$ ）を表し、第2のパラメータは量子化された距離値（ $distanceIdx$ ）を表す。2つのパラメータは、直線方程式を表す。一例では、距離1230は、 $distanceIdx$ （第2のパラメータ）に従って得られ得、角度 $alpha$ （1210）は、 $angleIdx$ （第1のパラメータ）に従って得られ得る。距離1230は、コーディング・ブロックの中心までの距離とすることができ、角度は、分離線と、コーディング・ブロックの中心点を通過する水平（又は同等の垂直）線との間の角度とすることができる。

【0210】

一例では、ステップ1において、コーディング・ブロックは、様々な方法で2つのサブブロックに分割されると考慮される。図9は、コーディング・ブロックのパーティションに関する例を示し、分離線1250は、ブロックが2つのサブブロックに分割されていることを示すために使用される。線1250を説明するために、1つのパラメータ角 $alpha$ 1210は、ビットストリームにおいてシグナリングされる。

【0211】

いくつかの実施形態では、図9に示すように、角度は、 x 軸と分離線との間で測定され、距離は、分離線に垂直で、現在のブロックの中心を通過するベクトルの長さによって測定される。別の例では、図10は、分離線を表す別の方法を示しており、角度と距離の例は、図9に示されている例とは異なる。

【0212】

ステップ2：算出された $sample_dist$ を使用して重み付け係数を計算し、その重み付け係数は、サンプルに対応する第1の予測値と第2の予測値との組み合わせのために使用される。一例では、重み付け係数は、 $sampleWeight1$ 及び $sampleWeight2$ として示され、これらは、第1の予測値に対応する重み及び第2の予測値に対応する重みを指している。

【0213】

ステップ3：サンプル座標（ x, y ）での予測サンプルの合計値は、座標（ x, y ）での第1の予測値、座標（ x, y ）での第2の予測値、 $sampleWeight1$ 及び $sampleWeight2$ に従って計算される。

【0214】

一例では、上記の例における $Step1$ は、以下のステップを含んでもよい。

【0215】

ステップ1.1：現在のブロックに対する角度パラメータ（ $alphaN$ ）のインデックス値、現在のブロックの幅の値（ W ）、現在のブロックの高さの値（ H ）を取得する。 W と H は、サンプル数の現在のブロックの幅と高さである。例えば、幅と高さの両方が8

10

20

30

40

50

に等しいコーディング・ブロックは、64のサンプルを含む正方形ブロックである。別の例では、W及びHは、ルマ・サンプルの数における現在のブロックの幅及び高さである。

【0216】

ステップ1.2: Wの値とHの値に従って、比 $whRatio$ の値を取得し、 $whRatio$ の値は、現在のコーディング・ブロックの幅と高さの比を表す。

【0217】

ステップ1.3: ルックアップ・テーブル、 $alpha$ の値、及び $whRatio$ の値に従って、 $stepD$ 値を取得し、一例では、図13に示すように、 $alpha$ の値及び $whRatio$ の値が、ルックアップ・テーブルのインデックス値として使用される。

【0218】

ステップ1.4: $sample_dist$ の値は、 $stepD$ の値に従って計算される。

【0219】

別の例では、上記の例におけるStep1は、以下のステップを含んでもよい。

【0220】

ステップ1.1: 現在のブロックに対する角度パラメータ($alphaN$)、距離インデックスの値($distanceIdx$)、現在のブロックの幅の値(W)、現在のブロックの高さの値(H)を取得する。

【0221】

ステップ1.2: Wの値とHの値に従って、比 $whRatio$ の値を取得し、 $whRatio$ の値は、現在のコーディング・ブロックの幅と高さの比を表す。

【0222】

ステップ1.3: ルックアップ・テーブル、 $alpha$ の値、及び $whRatio$ の値に従って、 $stepD$ 値を取得し、一例では、図13に示すように、 $alphaN$ の値及び $whRatio$ の値が、ルックアップ・テーブルのインデックス値として使用される。一例では、 $stepD$ 値は、サンプル距離計算プロセスのための量子化ステップ・サイズを表す。

【0223】

ステップ1.4: $sample_dist$ の値は、 $stepD$ の値、 $distanceIdx$ の値、角度の値($alphaN$)、Wの値、Hの値に従って、計算される。一例では、 $whRatio$ の値が式を使用して取得される。

【数15】

$$whRatio = (wIdx \geq hIdx) ? wIdx - hIdx : hIdx - wIdx$$

ここで、 $wIdx$ の値は、 $\log_2(W) - 3$ であり、 $hIdx$ の値は、 $\log_2(H) - 3$ である。

【0224】

別の例では、 $whRatio$ の値は、 $whRatio = (W \geq H) ? W/H : H/W$ として計算される。一例では、角度 $alpha$ の値を(復号器において)ビットストリームから取得することができる。一例では、角度の値範囲は、 $angleIdx$ として示される0~31(0と31を含む)の量子化された値範囲である。一例では、量子化された角度値は、32の異なる値をのみを取る(したがって、0~31の値は、どの角度値が選択されるかを表すのに十分である)。別の例では、角度値の値範囲は、0~15の間であってもよく、これは、16の異なる量子化された角度値が選択され得ることを意味する。一般に、角度値は、ゼロ以上の整数値であってもよいことに留意する。

【0225】

一例では、 $alphaN$ の値は、ビットストリームから得られるインデックス値であるか、又は $alpha$ の値は、ビットストリームから取得されるインジケータの値に基づいて計算される。例えば、 $alphaN$ の値は、以下の式に従って計算されてもよい。

【数16】

10

20

30

40

50

$\alpha N = \text{angleIdx}$	if $\text{angleIdx} \geq 0 \ \&\& \ \text{angleIdx} \leq \text{Lim}$, or
$\alpha N = 2 * \text{Lim} - \text{angleIdx}$	if $\text{angleIdx} > \text{Lim} \ \&\& \ \text{angleIdx} \leq 2 * \text{Lim}$, or
$\alpha N = \text{angleIdx} - 2 * \text{Lim}$	if $\text{angleIdx} > 2 * \text{Lim} \ \&\& \ \text{angleIdx} \leq 3 * \text{Lim}$,
or	
$\alpha N = 4 * \text{Lim} - \text{angleIdx}$	otherwise.

一方、 angleIdx は、ビットストリームから得られるインジケータの値である。 angleIdx の値は整数値で、 angleIdx の値の範囲は0～31（0と31を含む）である。

10

【0226】

別の例では、値 αN は、以下の式の1つに従って計算され得る。

【数17】

$\alpha N = \text{angleIdx}$	, or
$\alpha N = 2 * \text{Lim} - \text{angleIdx}$	, or
$\alpha N = \text{angleIdx} - 2 * \text{Lim}$	, or
$\alpha N = 4 * \text{Lim} - \text{angleIdx}$	.

20

上記の式では、 Lim は量子化された角度値の数に關係する一定の予め定義された値である。例えば、32の異なる量子化された角度値がある場合、数 Lim は8に等しくてもよい（ $4 * \text{Lim}$ は、32に等しい）。別の例では、 Lim は、4であり得、これは、合計16の異なる量子化角度値に対応する。

【0227】

一例では、 sample_dist の値が以下の式に従って取得される。

【数18】

$\text{sample_dist} = ((2 * x) - W + K) * f_1(\alpha) + ((2 * y) - H + K) * f_2(\alpha) - \text{distanceIdx} * \text{stepD} * \text{scaleStep}$, where:

30

・ここで、 $f_1()$ 及び $f_2()$ は関数であり、 α は、関数に対する入力である。一例では、 $f_1()$ と $f_2()$ は、ルックアップ・テーブルとして実装され、 α の値はルックアップ／テーブルのインデックス値である。

・ scaleStep の値は、コーディング（復号又は符号化）ブロックの幅の値、又はコーディング・ブロックの高さの値のいずれかに等しい。

・ x 及び y は、コーディング・ブロックの左上のサンプルに対するサンプルの座標である。

・ K は、整数のオフセット値である。一例では、 K の値は、1である。

【0228】

40

一例では、関数 $f_1()$ 及び $f_2()$ は、ルックアップ・テーブルとして実装される。一例では、関数 $f_1()$ と関数 $f_2()$ は、 x と y の値の変化に対する sample_dist の値の増分変化を表す。いくつかの例では、 $f_1(\text{index})$ は、 sample_dist の値が x の値の1ユニットの増加で変化する（ユニットは1に等しい増加かもしれない）ことを表すが、 $f_2(\text{index})$ は、 sample_dist の値が y の値の1ユニットの増加で変化することを表す。インデックスの値は、ビットストリームにおけるインジケータの値から取得され得る。

【0229】

別の例では、 sample_dist の値が式に従って取得される。

【数19】

50

$\text{sample_dist} = ((2 * x) - W + K) * f(\text{dispIdxN}) + ((2 * y) - H + K) * f(\text{dispIdxS}) - \text{distanceIdx} * \text{stepD} * \text{scaleStep}$, where:

ここで、

- ・ $f()$ は、関数であり、一例では、 $f()$ はルックアップ・テーブルとして実装される。
- ・ dispIdxN 又は dispIdxS は、ビットストリームから取得されるか、又はビットストリームにおけるインジケータの値に基づいて計算されるインデックス値である。
- ・ scaleStep の値は、コーディング（復号又は符号化）ブロックの幅の値、又はコーディング・ブロックの高さの値のいずれかに等しい。
- ・ x 及び y は、コーディング・ブロックの左上のサンプルに対するサンプルの座標である。
- ・ K は、整数のオフセット値である。一例では、 K の値は、1 である。

【0230】

一例では、関数 $f()$ は、ルックアップ・テーブルとして実装される。関数 $f()$ は、 x と y の値の変化に対する sample_dist の値の増分変化を表す。例では、 $f(\text{index1})$ は、 sample_dist の値が x の値の1ユニットの増加で変化し、 $f(\text{index2})$ は、 sample_dist の値が y の値の1ユニットの増加で変化することを表す。 index1 及び index2 の値は、ビットストリームにおけるインジケータの値に従って取得され得る（0以上の整数値を有する）テーブルへのインデックスである。

【0231】

一例では、関数 $f()$ の実装が図14に示す。この例では、 idx の値が入力パラメータ（ index1 又は index2 である）であり、関数の出力は $f(\text{idx})$ として示される。一例では、 $f()$ は、整数演算を使ったコサイン関数の実装形式であり、 idx （入力インデックス値）は、量子化された角度値を表す。

【0232】

一実施形態では、 stepD 値は、サンプル距離計算のための量子化された距離値を表す。

【0233】

一例では、図13に示すように、 whRatio の値及び角度の値（ α ）の値に従って、 stepD の値が取得される。一例では、 stepD の値は、 $\text{stepD} = \text{lookupTable}[\alpha N][\text{whRatio}]$ として取得することができ、ここで、 αN の値は、ビットストリームから取得されるインデックス値であるか、又は αN の値は、ビットストリームから取得されるインジケータの値に基づいて計算される。例えば、 α は、以下の式に従って計算され得る。

【数20】

$\alpha N = \text{angleIdx}$	if $\text{angleIdx} \geq 0 \ \&\& \ \text{angleIdx} \leq 8$, or
$\alpha N = 16 - \text{angleIdx}$	if $\text{angleIdx} > 8 \ \&\& \ \text{angleIdx} \leq 16$, or
$\alpha N = \text{angleIdx} - 16$	if $\text{angleIdx} > 16 \ \&\& \ \text{angleIdx} \leq 24$, or
$\alpha N = 32 - \text{angleIdx}$	otherwise,

ここで、 angleIdx は、ビットストリームに従って取得されるインデックス値である。

【0234】

別の例では、

【数21】

10

20

30

40

50

$\alpha N = \text{angleIdx}$	if $\text{angleIdx} \geq 0 \ \&\& \ \text{angleIdx} \leq \text{Lim}$, or
$\alpha N = 2 * \text{Lim} - \text{angleIdx}$	if $\text{angleIdx} > \text{Lim} \ \&\& \ \text{angleIdx} \leq 2 * \text{Lim}$, or
$\alpha N = \text{angleIdx} - 2 * \text{Lim}$	if $\text{angleIdx} > 2 * \text{Lim} \ \&\& \ \text{angleIdx} \leq 3 * \text{Lim}$,
or	
$\alpha N = 4 * \text{Lim} - \text{angleIdx}$	otherwise.

又は、別の例では、

【数 2 2】

$\alpha N = \text{angleIdx}$	, or	10
$\alpha N = 2 * \text{Lim} - \text{angleIdx}$	, or	
$\alpha N = \text{angleIdx} - 2 * \text{Lim}$	, or	
$\alpha N = 4 * \text{Lim} - \text{angleIdx}$		

上記の式では、 Lim は量子化された角度値の数に関係する一定の予め定義された値である。例えば、32の異なる量子化された角度値がある場合、数 Lim は8に等しくてもよい（ $4 * \text{Lim}$ は、32に等しい）。別の例では、 Lim は、4であり得、これは、合計16の異なる量子化角度値に対応する。

【0235】

一例では、 sample_dist の値は、 $\text{distanceIdx} * \text{stepD} * \text{scaleStep}$ に従って取得され、ここで、 distanceIdx は、ビットストリームに従って取得されるインデックス値であり、 scaleStep の値は、ブロックの幅の値又はブロックの高さの値のいずれかに従って取得される。乗算の結果は、コーディング・ブロック（座標 $x = W/2$ 及び $y = H/2$ を有する）の中心点までの分離線の距離を表す。

【0236】

一例では、ルックアップ・テーブルは、予め定義されたテーブルである。予め定義されたテーブルは、以下の利点を有する。

- ・サンプルの分離線までの距離を取得することは通常複雑であり、大量生産された消費者製品を対象とするビデオ・コーディング規格を実装するときには受け入れ可能ではない三角方程式を解く必要がある。

【0237】

いくつかの例では、サンプル距離は、ルックアップ・テーブル（予め定義されてもよい）に従って取得され、そのテーブルは、 whRatio 及び α に従って中間結果を予め計算しており、それらは、すでに整数演算（したがって、一例では、すべての stepD 値は、整数である）に従っている。ルックアップ・テーブルを使用して得られる中間結果は、以下の理由で慎重に選択される。

- ・ルックアップ・テーブルは、複素演算に対する中間計算結果（三角計算）を含むため、実装の複雑性は低減される。

- ・テーブルのサイズは小さく保たれる（これは、メモリを必要とする）。

【0238】

別の例では、 sample_dist の値は、 $\text{distanceIdx} * (\text{stepD} + T) * \text{scaleStep}$ に従って取得され、ここで、 T は、整数値を有するオフセット値である。一例では、 T の値は、32である。

【0239】

別の例では（復号器と符号化器の両方の観点から）、上記の例における sample_dist の値は、図11又は図12に示すように、現在のコーディング・サブブロックにおけるサンプルと分離線との距離を示す。

【0240】

10

20

30

40

50

2つのサブブロックは、0と1でインデックス付けされる。第1のサブブロックは0でインデックス付けされ、第2のサブブロックは1でインデックス付けされる。

【0241】

幾何学的パーティション・モードでは、2つのサブブロックの処理順序はサブブロックのインデックスに基づいて決定される。インデックス0のサブブロックが最初に処理され、その後にインデックス1のサブブロックが処理されるだろう。処理順序は、サンプル重み導出プロセス、モーション情報記憶プロセス、モーション・ベクトル導出プロセスなどにおいて使用されてもよい。

【0242】

一例では、サブブロックのインデックス値は、サブブロックにおけるサンプルの `sample_dist` 値に基づいて決定される。

【0243】

1つのサブブロックにおける各サンプルの `sample_dist` 値がゼロより小さい（又は等しい）場合、サブブロックに対するインデックス値は0にセットされる。

【0244】

1つのサブブロックにおける各サンプルの `sample_dist` 値がゼロより大きい（又は等しい）場合、サブブロックに対するインデックス値は1にセットされる。

【0245】

同じ例では、サブブロックに対するインデックス値は修正（例えば、反転）されてもよい。

【0246】

一例では、サブブロックのインデックス値は、予め定義されたサンプル位置に基づいて修正されてもよい。

【0247】

一例では、コーディング・ブロックの予め定義された位置（例えば、左下の位置）が位置するサブブロックに対するインデックス値は、常に0としてセットされる。

【0248】

一例では、予め定義されたサンプル位置がインデックス値1を有する第2のサブブロックに属する場合、このサブブロックに対するインデックス値は、0に修正されるだろう（他のサブブロックに対するインデックス値は、1に修正される）。

【0249】

一例では、予め定義されたサンプル位置がインデックス0を有する第1のサブブロックに属する場合、サブブロックのインデックスは変更されない。

【0250】

図20及び図21に示す例について。予め定義されたサンプル位置は、現在のコーディング・ブロックの左下のサンプル位置である。

【0251】

図20では、左下のサンプルはインデックス0を有する第1のサブブロックに属し、サブブロックのインデックスは変更されていない。

【0252】

図21では、左下のサンプルは、インデックス1（図21a）を有する第2のサブブロックに属し、したがって、2つのサブブロックのインデックスが、図21bに示されるように、交換又は反転され、プロセス順序は、交換又は反転されたインデックスに基づくだろう。

【0253】

一例では、インデックス割り当て及び反転処理は、以下のステップで行われ得る。
ステップ1：現在のコーディング・ブロックを分裂モードに基づいて2つのサブブロックに分裂させる。
ステップ2：1つのサブブロック内のサンプルに対して `sample_dist` 値を計算する。

10

20

30

40

50

【0254】

`sample_dist` 値が負の値かゼロに等しい場合、このサブブロックに対するインデックス値は0（第1のサブブロックに対応する）にセットされ、他のサブブロックに対するインデックス値は1（第2のサブブロックに対応する）にセットされる。

あるいは、

`sample_dist` 値が正の値に等しい場合、このサブブロックに対するインデックス値は1（第2のサブブロックに対応する）にセットされ、他のサブブロックに対するインデックス値は0（第1のサブブロックに対応する）にセットされる。

【0255】

ステップ3：予め定義されたサンプル位置の `sample_dist` 値が正の値である場合、2つのサブブロックのインデックスを交換する。

10

【0256】

定義されたサンプル位置の `sample_dist` 値が負の場合、2つのサブブロックのインデックスは変更されない。

【0257】

本発明の一実施形態では、復号デバイス又は符号化デバイスによって実装されるコーディングの方法が開示される。

【0258】

方法は、以下を含む。

【0259】

20

S101：現在のコーディング・ブロックに対する分裂（又はパーティション）モードを取得する。

【0260】

分裂モードは、コーディング・ブロックをサブブロックに分裂させるために使用される。分裂モードは、角度及び／又は距離を有するように定義された形式の例であってもよい。水平方向に関する角度は、分離線に垂直な方向として記述される。分割線は、現在のコーディング・ブロックが2つのサブブロックに分割されていることを示すために使用される。距離は、現在のコーディング・ブロックの中心位置から分離線までの距離を説明している。いくつかの例では、分裂モードは、ビットストリームにおけるインジケータの値に従って表される。インジケータの値は、角度パラメータ値と距離パラメータ値を取得するために使用される。一例によれば、対（第1のパラメータ（角度）、第2のパラメータ（距離））は、予め定義された方法又は関数から導出され得る。言い換えれば、パーティション・モードのインジケータは、復号器のビットストリームから解析される。このインジケータは、方法又は関数への入力として定義される。方法又は関数の出力は、対（第1パラメータと第2パラメータ）である。

30

【0261】

インジケータ（`geo_partition_idx`）と予め定義された方法又は関数の一例は、以下のようである。

【0262】

関数1

40

インジケータXは、入力（0～63）である。

【数23】

if (X<4)

Angle = 0, Dist =(X>=2)?X-4:X+1;

Else if (X>=60)

Angle = 8, Dist = (X>61)?X-64:X-59;

Else

Z=(X-4)/14;

K = Z+1+Z/3;

Angle = (X-4)%14<7 ? K: 16-K;

Dist = (X-4)%7 - 3.

インジケータXは、0～63の値を有し、これは、第1パラメータ (Angle) と第2パラメータ (Dist) を導出するために使用される。出力結果の例を表2に示す。テーブル2：関数1の入力及び出力テーブルであり、geo_partition_idx (例えば、インジケータX) は、テーブルのインデックスであり、angleIdx (例えば、angle) は第1のパラメータであり、distanceIdx (例えば、Dist) は第2のパラメータである。

10

20

30

40

50

【表 3 - 1】

geo_partition_idx	angleIdx	distanceIdx	
0	0	1	
1	0	2	
2	0	-2	
3	0	-1	
4	1	-3	
5	1	-2	
6	1	-1	10
7	1	0	
8	1	1	
9	1	2	
10	1	3	
11	15	-3	
12	15	-2	
13	15	-1	
14	15	0	
15	15	1	
16	15	2	20
17	15	3	
18	2	-3	
19	2	-2	
20	2	-1	
21	2	0	
22	2	1	
23	2	2	
24	2	3	
25	14	-3	30
26	14	-2	
27	14	-1	
28	14	0	
29	14	1	
30	14	2	
31	14	3	
32	3	-3	
33	3	-2	
34	3	-1	
35	3	0	40

【表 3 - 2】

geo_partition_idx	angleIdx	distanceIdx	
36	3	1	
37	3	2	
38	3	3	
39	13	-3	
40	13	-2	
41	13	-1	
42	13	0	10
43	13	1	
44	13	2	
45	13	3	
46	5	-3	
47	5	-2	
48	5	-1	
49	5	0	
50	5	1	20
51	5	2	
52	5	3	
53	11	-3	
54	11	-2	
55	11	-1	
56	11	0	
57	11	1	
58	11	2	
59	11	3	30
60	8	1	
61	8	2	
62	8	-2	
63	8	-1	

【 0 2 6 3 】

インジケータ (geo _ p a r t i t i o n _ i d x) と予め定義された方法又は関数の別の例は、以下のである。

【 0 2 6 4 】

関数 2

インジケータ X は、入力 (0 ~ 6 3) である。

【数 2 4】

10

20

30

40

50

if (X<4)

Angle = 0, Dist =(X>=2)?X-4:X+1;

Else if (X>=60)

Angle = 8, Dist = (X>61)?X-64:X-59;

Else

Z=(X-4)/14;

K = Z+1+Z/3;

Angle = (X-4)%14<7 ? K: 16-K;

Dist = (X-4)%7 - 3.

If (Dist < 0)

Angle = Angle + 16;

Dist = - Dist.

インジケータXは、0～63の値を有し、これは、第1パラメータ (A n g l e) と第2パラメータ (D i s t) を導出するために使用される。出力結果の例が表3に記述される。テーブル3：関数2の入力及び出力テーブルであり、g e o _ p a r t i t i o n _ i d x (例えば、インジケータX) は、テーブルのインデックスであり、a n g l e I d x (例えば、A n g l e) は第1のパラメータであり、d i s t a n c e I d x (例えば、D i s t) は第2のパラメータである。

10

20

30

40

50

【表 4－1】

geo_partition_idx	angleIdx	distanceIdx	
0	0	1	
1	0	2	
2	16	2	
3	16	1	
4	17	3	
5	17	2	
6	17	1	
7	1	0	10
8	1	1	
9	1	2	
10	1	3	
11	31	3	
12	31	2	
13	31	1	
14	15	0	
15	15	1	
16	15	2	20
17	15	3	
18	18	3	
19	18	2	
20	18	1	
21	2	0	
22	2	1	
23	2	2	
24	2	3	
25	30	3	
26	30	2	30
27	30	1	
28	14	0	
29	14	1	
30	14	2	
31	14	3	
32	19	3	
33	19	2	
34	19	1	
35	3	0	40

【表 4－2】

geo_partition_idx	angleIdx	distanceIdx	
36	3	1	
37	3	2	
38	3	3	
39	29	3	
40	29	2	
41	29	1	
42	13	0	10
43	13	1	
44	13	2	
45	13	3	
46	21	3	
47	21	2	
48	21	1	
49	5	0	
50	5	1	20
51	5	2	
52	5	3	
53	27	3	
54	27	2	
55	27	1	
56	11	0	
57	11	1	
58	11	2	
59	11	3	30
60	8	1	
61	8	2	
62	24	2	
63	24	1	

【0265】

S102：現在のコーディング・ブロックは、分裂モードに基づいて、2つのサブブロック（サブブロックA、サブブロックB）に分裂されると考慮される。

【0266】

このステップでは、分割モードに従って、コーディング・ブロックにおけるサンプルは、2つのサブブロックに位置すると考慮される。サブブロックA又はサブブロックBは、現在のコーディング・ブロックにおけるサンプルの一部（すべてではない）を含む。サブブロックA又はサブブロックBは、各サンプルのsample_distの符号に従って表されてもよい。sample_distは、上記の例及び実施形態に従って取得されてもよい。

【0267】

同じ例では、このステップは、コーディング・ブロックを2つのサブブロックに分割することに関する上述の実施形態又は例を指すことができる。例えば、コーディング・ブロックは、幾何学的モデルに従って2つのサブブロックに分割されてもよい。このモデルは

10

20

30

40

50

、図9～12に例示されているように、分離線（例えば、直線）によるブロックの分裂をもたらし得る。

【0268】

図9は、分離線1250に従って、コーディング・ブロックが2つのサブブロックに分割されていると考えられる、コーディング・ブロックのパーティションに関する例を示す。線1250を説明するために、2つのパラメータがシグナリングされ、一方のパラメータは角度 α 1210であり、他方のパラメータは距離 d 1230である。

【0269】

いくつかの実施形態では、図9に示すように、角度は、x軸と分離線との間で測定され、距離は、分離線に垂直で、現在のブロックの中心を通過するベクトルの長さによって測定される。

10

【0270】

別の例では、図10は、分離線を表す別の方法を示しており、角度と距離の例は、図9に示されている例とは異なる。符号化器側において、分離線（線を定義するパラメータ、例えば角度及び距離）は、レート歪みベースのコスト関数に基づいて判定される。判定された線パラメータは、ビットストリームに符号化される。復号器側において、線パラメータはビットストリームに従って復号（取得）される。

【0271】

一例では、輝度成分（ルマ）と2つのクロミナンス成分（クロマ）を含む3つのビデオ・チャンネルの場合である。この分裂プロセスは、ルマ又はクロマに対して行われてもよい。

20

【0272】

一実施形態では、コーディング・ブロック内のサンプルに対して、距離値は、このサンプルがどのサブブロック（サブブロックA又はサブブロックB）に位置するかを決定するために使用される。一例では、コーディング・ブロック内の各サンプルに対して、距離値が取得される。

【0273】

一例では、サンプルに対応する距離値が0以下であるときに、このサンプルはサブブロックA（又はサブブロックB）内にあり、サンプルに対応する距離値が0より大きいときに、このサンプルはサブブロックB（又はサブブロックA）内にある。

【0274】

別の例では、サンプルに対応する距離値が0より小さいときに、このサンプルはサブブロックA（又はサブブロックB）内にあり、サンプルに対応する距離値が0以上であるときに、このサンプルはサブブロックB（又はサブブロックA）内にある。

30

【0275】

サンプルに対応する距離値（例えば、 $sample_dist$ ）を取得するプロセスは、上記の例及び実施形態を参照することができ、例えば、 $sample_dist$ は、サンプルに対応する距離値を表す。

【0276】

S103：サブブロックAのインデックス値を、現在のコーディング・ブロックの予め定義されたサンプル位置に従ってセットする。

40

【0277】

いくつかの例では、所定のサンプル位置は、現在のコーディング・ブロックの左下、右下、左上、若しくは右上位置、又は中心位置であってもよい。

【0278】

予め定義されたサンプル位置がサブブロックA内に位置するときに、サブブロックAのインデックス値は第1の値に等しくセットされ、この例では、サブブロックBのインデックス値は第2の値に等しくセットされるだろう。

あるいは、

予め定義されたサンプル位置がサブブロックA内に位置するときに（例えば、予め定義されたサンプル位置がサブブロックB内にあるときに）、サブブロックAのインデックス値

50

は第2の値に等しくセットされ、この例では、サブブロックBのインデックス値は第1の値に等しくセットされるだろう。第1の値に対応する処理順序は、第2の値に対応する処理順序の前であり、予め定義されたサンプル位置が位置するサブブロックは、常に、他のサブブロックより前に処理される。処理は、サンプル重み導出プロセス、モーション情報記憶プロセス、モーション・ベクトル導出プロセスなどであってもよい。いくつかの例では、幾何学的パーティション・モードでは、2つのサブブロックの処理順序はサブブロックのインデックスに基づいて決定される。インデックス0のサブブロックが最初に処理され、その後にインデックス1のサブブロックが処理されるだろう。

【0279】

第1の値又は第2の値は整数値であってもよい。一例では、第1の値は、0もしくは1、又は他の値であってもよく、第2の値は、0もしくは1、又は他の値であってもよい。第1の値と第2の値が異なる。

【0280】

一例では、予め定義されたサンプル位置がサブブロックA内に位置するかどうかは、予め定義されたサンプル位置に対するサンプル距離値（例えば、`sample_dist`）に基づいて判定されてもよい。例えば、サブブロックA内のサンプルに対応するサンプル距離値が0より大きく、予め定義されたサンプル位置に対応するサンプル距離値が0より小さい場合、所定のサンプル位置はサブブロックA内ではなく、サブブロックB内にある。この例では、サブブロックB内の各サンプルは、0より小さいサンプル距離値を有する。いくつかの他の例では、サブブロックBの各サンプルは、0より小さいか、又は0に等しいサンプル距離値を有してもよく、それは、サブブロックBにおいて0に等しいサンプル距離値に対応するサンプルが考慮されるかどうかに依存する。別の例では、予め定義されたサンプル位置がサブブロックA内に位置するかどうかは、予め定義されたサンプル位置に対する座標値に基づいて判定されてもよい。

【0281】

実施形態2

【0282】

別の実施形態では、サブブロック又はブロックのインデックスは、予め定義された関数又はルックアップ・テーブルによってセットされ、予め定義された機能又はルックアップ・テーブルの入力値は、変数又はパラメータ値であってもよく、予め定義されたか、又はルックアップ・テーブルの出力は、サブブロック又はブロックのインデックス値である。

【0283】

一例では、ブロックのインデックスは`isFlip`として表される分裂インデックスであってもよく、インデックス値`isFlip`は、以下を満たす。

【数25】

$$\text{isFlip} = (\text{angleIdx} \geq 13 \ \&\& \ \text{angleIdx} \leq 27) ? 1 : 0.$$

【0284】

一例では、予め定義された関数又はルックアップ・テーブルの入力値は、ビットストリームから解析されてもよい。

【0285】

一例では、予め定義された関数又はルックアップ・テーブルの入力値は、指示情報の値に基づいて導出されてもよい。

【0286】

一例では、入力値は分裂境界の角度インデックスの値であり、これは、構文`geo_partition_idx`から導出される。

【0287】

一例では、幾何学的パーティションのサブブロックAのインデックス（`index`）値は、以下のルックアップ・テーブルから導出されてもよい。

【0288】

`partIdx[AngleIdx]` の値が 0 に等しい場合、幾何学的パーティションから分裂されたサブブロック A のパート・インデックスが 0 であることを意味し、`partIdx[AngleIdx]` の値が 1 に等しい場合、幾何学的パーティションから分裂されたサブブロック A のパート・インデックスが 1 であることを意味する。

【表 5】

表 4 : `AngleIdx` に基づくサブブロック A のためのルックアップ・テーブル `PartIdx`

<code>AngleIdx</code>	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
<code>partIdx[AngleIdx]</code>	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
<code>AngleIdx</code>	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
<code>partIdx[AngleIdx]</code>	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0

10

【0289】

テーブル 4 は、以下のように関数又は方程式の形式に変換され得る。

【数 26】

$$\text{partIdx} = \text{angleIdx} \geq 13 \ \&\& \ \text{angleIdx} \leq 27 ? 1 : 0.$$

20

`PartIdx` は、幾何学的パーティション・モード場合、部分反転を分裂させるために使用される。

【0290】

一例では、サブブロック A とサブブロック B が 1 つの幾何学的パーティションから分裂されるため、サブブロック B は、サブブロック A の反対のインデックス（例えば、値 0 に対応する値 1、又は値 1 に対応する値 0）を有するだろう。

【0291】

いくつかの例では、第 1 の値（上記の例では 0）に対応する処理順序は、第 2 の値（上記の例では 1）に対応する処理順序の前である。したがって、所定のサンプル位置が位置するサブブロックは、常に、他のサブブロックの前に処理される。処理は、サンプル重み導出プロセス、モーション情報記憶プロセス、モーション・ベクトル導出プロセスなどであってもよい。いくつかの例では、幾何学的パーティション・モードでは、2 つのサブブロックの処理順序はサブブロックのインデックスに基づいて決定される。インデックス 0 のサブブロックが最初に処理され、その後にインデックス 1 のサブブロックが処理されるだろう。

30

【0292】

いくつかの例では、幾何学的パーティション・モードでは、予測子の値は $\text{predictorA} * \text{Mask1} + \text{predictorB} * \text{Mask2}$ に等しく、`predictorA` の値又は `predictorB` の値は、他のモード（例えば、マージ・モード）から導出されてもよく、`Mask1` 及び `Mask2` は、幾何学的パーティションを形成するサブブロック分裂のための混合マスクである。混合マスクは、他のインジケータに基づいて事前に計算される。

40

【0293】

一例では、`angleIdx` 及び `distanceIdx` は `geo_partition_idx` から導出され、`geo_partition_idx` はビットストリームから解析される。

【0294】

サンプル位置と分裂境界の間の距離は、`angleIdx` 及び `distanceIdx`

50

に基づいて計算され、混合マスクは、事前に定義されたルックアップ・テーブルを使用して、この距離に依存する。

【0295】

一例では、幾何学的パーティションから分裂されたサブブロックのインデックス値（例えば、0又は1）が、サブブロックのインジケータ・マスク選択に使用されてもよい。

【0296】

一例では、サブブロックのインデックス値が第1の値（例えば0）である場合、Mask1は、予測子の値を生成するためにこのサブブロックに使用され、サブブロックのインデックス値が第2の値（例えば1）である場合、Mask2は、予測子の値を生成するためにこのサブブロックに使用される。

【0297】

一例では、サブブロックAのインデックス値が0である場合、サブブロックAにはMask1が使用され、サブブロックBにはMask2が使用される（この例では、サブブロックBのインデックス値は1である）。

【数27】

$$\text{Final predictor} = \text{predictorA} * \text{Mask1} + \text{predictorB} * \text{Mask2}.$$

【0298】

一例では、サブブロックAのインデックス値が1である場合、サブブロックAにはMask2が使用され、サブブロックBにはMask1が使用される（この例では、サブブロックBのインデックス値は0である）。

【数28】

$$\text{Final predictor} = \text{predictorA} * \text{Mask2} + \text{predictorB} * \text{Mask1}.$$

【0299】

一般形式では、以下のようなものである。

【数29】

$$\text{Final predictor} = \text{PartIdx}[\text{AngleIdx}] ?$$

$$\text{predictorA} * \text{Mask2} + \text{predictorB} * \text{Mask1} :$$

$$\text{predictorA} * \text{Mask1} + \text{predictorB} * \text{Mask2}.$$

【0300】

図22は、復号デバイスによって実装されるコーディングの方法を示し、方法は以下を含む。

【0301】

S2201：現在のコーディング・ブロックに対する分裂モード・インデックス値を取得する。

【0302】

一実装形式では、分裂モード・インデックス値は、どの幾何学的パーティション・モードが現在のコーディング・ブロックに使用されるかを示すために使用される。例えば、geo_partition_idx又はmerge_gpm_partition_idxである。

【0303】

一例では、merge_gpm_partition_idx[x0][y0]又はgeo_partition_idxは、幾何学的パーティショニング・マージ・モードの

10

20

30

40

50

パーティショニング形状を指定する。配列インデックス $x0$ 、 $y0$ は、ピクチャの左上のルマ・サンプルに対する、考慮されるコーディング・ブロックの左上のルマ・サンプルの位置 ($x0$, $y0$) を指定する。

【0304】

`merge_gpm_partition_idx[x0][y0]` 又は `go_partition_idx[x0][y0]` が存在しないときに、それは、0 に等しいと推論される。

【0305】

一例では、分裂モード・インデックス値は、ビデオ・ビットストリームにおいてコーディングされたインデックス値を解析することによって取得されてもよいし、分裂モード・インデックス値は、ビデオ・ビットストリームから解析される構文値に従って判定されてもよい。

【0306】

ビットストリームは、無線ネットワーク又は有線ネットワークに従って取得されてもよい。ビットストリームは、同軸ケーブル、光ファイバケーブル、ツイスト・ペア、デジタル加入者線、又は赤外線、ラジオ、マイクロ波、WIFI、Bluetooth、LTE 若しくは 5G の無線技術を使用して、ウェブサイト、サーバ、又は他のリモート・ソースから送信されてもよい。

【0307】

一実施形態では、ビットストリームは、例えば、1 つ以上のコーディングされたビデオ・シーケンス (CVS) を形成するアクセス・ユニット (AU) のシーケンスの表現を形成するネットワーク抽象化レイヤ (NAL) ユニット・ストリーム又はバイト・ストリームの形態のビットのシーケンスである。

【0308】

いくつかの実施形態では、復号プロセスの場合、復号器側がビットストリームを読み、ビットストリームから復号されたピクチャを導出し、符号化プロセスの場合、符号化器側がビットストリームを生成する。

【0309】

通常、ビットストリームは、構文構造によって形成される構文要素を含むだろう。

構文要素：ビットストリームで表現されるデータの要素

構文構造：指定された順序でビットストリーム内に一緒に存在する 0 個以上の構文要素

【0310】

特定の例では、ビットストリームフォーマットは、ネットワーク抽象化レイヤ (NAL) ユニット・ストリームとバイト・ストリームとの間の関係を指定し、これらはいずれもビットストリームと呼ばれる。

【0311】

ビットストリームは、例えば、NAL ユニット・ストリーム・フォーマット又はバイト・ストリーム・フォーマットの 2 つのフォーマットのいずれかであり得る。NAL ユニット・ストリーム・フォーマットは概念的には、より「基本的な」タイプである。NAL ユニット・ストリーム・フォーマットは、NAL ユニットと呼ばれる構文構造のシーケンスを含む。このシーケンスは、復号順序で順序付けされる。NAL ユニット・ストリームにおける NAL ユニットの復号順序 (及び) に課される制約がある。

【0312】

バイト・ストリーム・フォーマットは、NAL ユニットの復号順に順序付け、各 NAL ユニットに開始コードプレフィックスとゼロ以上のゼロ値バイトをプレフィックスしてバイトのストリームを形成することによって、NAL ユニット・ストリーム・フォーマットから構成されてもよい。NAL ユニット・ストリーム・フォーマットは、このバイトのストリームにおけるユニークな開始コード・プレフィックス・パターンの位置をサーチすることによって、バイト・ストリーム・フォーマットから抽出され得る。

【0313】

10

20

30

40

50

この項は、ビットストリームを介して与えられる、ソースと復号されたピクチャとの間の関係の一実施形態を指定する。

【0314】

ビットストリームによって表されるビデオ・ソースは、復号順序におけるピクチャのシーケンスである。

【0315】

通常、`merge_gpm_partition_idx[x0][y0]` の値は、ビットストリームから復号される。一例では、`merge_gpm_partition_idx[][]` の値の範囲は、0と63を含む0～63である。一例では、`merge_gpm_partition_idx[][]` の復号プロセスは、「バイパス」である。
`merge_gpm_partition_idx[x0][y0]` が存在しないときに、それは0に等しいと推論される。

【0316】

S2202：分裂モード・インデックス値とテーブルに従って、現在のコーディング・ブロックに対する角度インデックス値 `angleIdx` を取得する。

【0317】

一実装形式では、`angleIdx` は、現在のコーディング・ブロックの幾何学的パーティションに使用される。

【0318】

一例では、`angleIdx` は、幾何学的パーティションの角度インデックスを指定する。

【0319】

現在のブロックに対する角度パラメータの値は、分裂モードのインデックス値と予め定義されたルックアップ・テーブルの値に従って取得される。

【0320】

一実施形態では、幾何学的パーティショニング・モードのパーティション角度変数 `angleIdx` （角度パラメータ）及び距離変数 `distanceIdx` は、以下の表に指定される `merge_gpm_partition_idx[xCb][yCb]` （インジケータ）の値に従ってセットされる。実装形式では、この関係は、表1に従って、又は機能に従って実装され得る。

【表6】

表1：merge_gpm_partition_idxに基づく angleIdx 及び distanceIdx の仕様

merge_gpm_partition_idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
angleIdx	0	0	2	2	2	2	3	3	3	3	4	4	4	4	5	5
distanceIdx	1	3	0	1	2	3	0	1	2	3	0	1	2	3	0	1
merge_gpm_partition_idx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
angleIdx	5	5	8	8	11	11	11	11	12	12	12	12	13	13	13	13
distanceIdx	2	3	1	3	0	1	2	3	0	1	2	3	0	1	2	3
merge_gpm_partition_idx	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
angleIdx	14	14	14	14	16	16	18	18	18	19	19	19	20	20	20	21
distanceIdx	0	1	2	3	1	3	1	2	3	1	2	3	1	2	3	1
merge_gpm_partition_idx	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
angleIdx	21	21	24	24	27	27	27	28	28	28	29	29	29	30	30	30
distanceIdx	2	3	1	3	1	2	3	1	2	3	1	2	3	1	2	3

【0321】

S2203：角度インデックス値 `angleIdx` に従って、インデックス値 `part`

`I d x`をセットする。

【0322】

一実装形式では、インデックス値 `p a r t I d x` は、以下を満たす。

【数30】

$$partIdx = (angleIdx \geq threshold1 \ \&\& \ angleIdx \leq threshold2) ? 1 : 0$$

`t h r e s h o l d 1`及び`t h r e s h o l d 2`は整数値であり、`t h r e s h o l d 1`は`t h r e s h o l d 2`より小さい。

【0323】

一例では、`t h r e s h o l d 1`が13であり、`t h r e s h o l d 2`が27である。

【0324】

一例では、インデックス値は`i s F l i p`として表される分裂インデックスであってもよく、インデックス値 `i s F l i p` は、以下も満たす。

【数31】

$$isFlip = (angleIdx \geq 13 \ \&\& \ angleIdx \leq 27) ? 1 : 0.$$

【0325】

S2204：インデックス値 `p a r t I d x` に従って、現在のコーディング・ブロックを復号する。

【0326】

復号プロセスは、サンプル重み導出プロセス、運動情報記憶プロセス、モーション・ベクトル導出プロセスなどであってもよい。したがって、バッファ利用と復号効率が改善されている。

【0327】

一実装形式では、コーディング・ブロックを復号することは、インデックス値 `p a r t I d x` に従って、現在のブロックのためのモーション情報を記憶することを含む。

【0328】

上記の実施形態によれば、コーディング・ブロックは、インデックス値に従って復号される。復号プロセスは、サンプル重み導出プロセス、運動情報記憶プロセス、モーション・ベクトル導出プロセスなどであってもよい。したがって、バッファ利用と復号効率が改善されている。

【0329】

図23は、ビデオ復号器であって、現在のコーディング・ブロックに対する分裂モード・インデックス値を取得するように構成されている解析ユニット2301と、分裂モード・インデックス値、及びテーブルに従って、現在のコーディング・ブロックに対する角度インデックス値 `a n g l e I d x` を取得するように構成されている角度インデックス値取得ユニット2302と、角度インデックス値 `a n g l e I d x` に従って、インデックス値 `p a r t I d x` をセットするように構成されているセッティング・ユニット2303と、インデックス値 `p a r t I d x` に従って、現在のコーディング・ブロックを復号するように構成されている処理ユニット2304と、含む、ビデオ復号器を示す。

【0330】

図22による方法は、図23による復号器によって実行され得る。

【0331】

上記方法のさらなる特徴及び実装形式は、復号器の特徴及び実装形式に対応する。

【0332】

1つの具体的な例では、幾何学的パーティショニング・モードのためのモーション・ベクトル記憶プロセスについてである。

【0333】

このプロセスは、`M e r g e G p m F l a g [ x C b ] [ y C b ]` が1のコーディング

10

20

30

40

50

・ユニットを復号するとき呼び出される。

【0334】

このプロセスへの入力は、

現在のピクチャの左上のルマ・サンプルに対する現在のルマ・コーディング・ブロックの
左上のサンプルを指定するルマ位置 (x C b, y C b)、

ルマ・サンプルにおける現在のコーディング・ブロックの幅を指定する変数 c b W i d t h、

ルマ・サンプルにおける現在のコーディング・ブロックの高さを指定する変数 c b H e i g h t、

幾何学的パーティションの角度インデックスを指定する変数 a n g l e I d x、

幾何学的パーティションの距離インデックスを指定する可変 d i s t a n c e I d x、

1 / 16 分数サンプル精度におけるルマ・モーション・ベクトル m v A 及び m v B、

参照インデックス r e f I d x A 及び r e f I d x B、

予測リスト・フラグ p r e L i s t F l a g A 及び p r e L i s t F l a g B。

水平方向と垂直方向の現在のコーディング・ブロックの 4 × 4 ブロックの数を指定する変数
n u m S b X と n u m S b Y は、それぞれ c b W i d t h >> 2 と c b H e i g h t >> 2 に等しくセットされる。

変数、変位 X、変位 Y、i s F l i p、及び s h i f t H o r は、以下のように導出される。

【数32】

displacementX = angleIdx

displacementY = (angleIdx + 8) % 32

isFlip = (angleIdx >= 13 && angleIdx <= 27) ? 1 : 0

shiftHor = (angleIdx % 16 == 8 || (angleIdx % 16 != 0 && cbHeight >=

cbWidth)) ? 0 : 1

【0335】

変数 o f f s e t X と o f f s e t Y は、以下のように導出される。

S h i f t H o r が 0 に等しい場合、以下が適用される。

【数33】

offsetX = (-cbWidth) >> 1

offsetY = ((-cbHeight) >> 1) +

(angleIdx < 16 ? (distanceIdx * cbHeight) >> 3 : - ((distanceIdx * cbHeight) >> 3)

)

そうではない (S h i f t H o r が 1 に等しい) 場合、以下が適用される。

【数34】

10

20

30

40

50

```

offsetX = ( ( -cbWidth ) >> 1 ) +
    ( angleIdx < 16 ? ( distanceIdx * cbWidth ) >> 3 : - ( ( distanceIdx * cbWidth ) >> 3 )
)
offsetY = ( -cbHeight ) >> 1

```

$xSbIdx = 0 \dots numSbX - 1$ 、及び $ySbIdx = 0 \dots numSbY - 1$ であるサブブロック・インデックス $(xSbIdx, ySbIdx)$ での各 4×4 個のサブブロックに対して、以下が適用される：

10

【数 3 3 6】

変数 $motionIdx$ は、以下のように表 3 7 に指定されたアレイ $disLut$ に基づき、以下のように計算される。

【数 3 5】

```

motionIdx = ( ( ( 4 * xSbIdx + offsetX ) << 1 ) + 5 ) * disLut[ displacementX ] +
    ( ( ( 4 * ySbIdx + offsetY ) << 1 ) + 5 ) * disLut[ displacementY ]

```

20

変数 $sType$ は、以下のように導出される。

【数 3 6】

```

sType = abs( motionIdx ) < 32 ? 2 : ( motionIdx <= 0 ? ( 1 - isFlip ) : isFlip )

```

$sType$ の値に依存して、以下の割り当てが行われる。

【数 3 3 7】

$sType$ が 0 に等しい場合、以下が適用される。

【数 3 7】

```

predFlagL0 = ( predListFlagA == 0 ) ? 1 : 0
predFlagL1 = ( predListFlagA == 0 ) ? 0 : 1
refIdxL0 = ( predListFlagA == 0 ) ? refIdxA : -1
refIdxL1 = ( predListFlagA == 0 ) ? -1 : refIdxA
mvL0[ 0 ] = ( predListFlagA == 0 ) ? mvA[ 0 ] : 0
mvL0[ 1 ] = ( predListFlagA == 0 ) ? mvA[ 1 ] : 0
mvL1[ 0 ] = ( predListFlagA == 0 ) ? 0 : mvA[ 0 ]
mvL1[ 1 ] = ( predListFlagA == 0 ) ? 0 : mvA[ 1 ]

```

30

40

そうではなく、 $sType$ が 1 に等しいか、又は ($sType$ が 2 に等しく、 $predListFlagA + predListFlagB$ が 1 に等しくない) 場合、以下が適用される。

【数 3 8】

50

```

predFlagL0 = ( predListFlagB == 0 ) ? 1 : 0
predFlagL1 = ( predListFlagB == 0 ) ? 0 : 1
refIdxL0 = ( predListFlagB == 0 ) ? refIdxB : -1
refIdxL1 = ( predListFlagB == 0 ) ? -1 : refIdxB
mvL0[ 0 ] = ( predListFlagB == 0 ) ? mvB[ 0 ] : 0
mvL0[ 1 ] = ( predListFlagB == 0 ) ? mvB[ 1 ] : 0
mvL1[ 0 ] = ( predListFlagB == 0 ) ? 0 : mvB[ 0 ]
mvL1[ 1 ] = ( predListFlagB == 0 ) ? 0 : mvB[ 1 ]

```

10

そうではない（`sType`が2に等しく、`predListFlagA+predListFlagB`が1に等しくない）場合、以下が適用される。

【数39】

```

predFlagL0 = 1
predFlagL1 = 1
refIdxL0 = ( predListFlagA == 0 ) ? refIdxA : refIdxB
refIdxL1 = ( predListFlagA == 0 ) ? refIdxB : refIdxA
mvL0[ 0 ] = ( predListFlagA == 0 ) ? mvA[ 0 ] : mvB[ 0 ]
mvL0[ 1 ] = ( predListFlagA == 0 ) ? mvA[ 1 ] : mvB[ 1 ]
mvL1[ 0 ] = ( predListFlagA == 0 ) ? mvB[ 0 ] : mvA[ 0 ]
mvL1[ 1 ] = ( predListFlagA == 0 ) ? mvB[ 1 ] : mvA[ 1 ]

```

20

30

【0338】

$x = 0 \dots 3$ 及び $y = 0 \dots 3$ に対して、以下の割り当てが行われる。

【数40】

40

50

$MvL0[(xSbIdx \ll 2) + x][(ySbIdx \ll 2) + y] = mvL0$

$MvL1[(xSbIdx \ll 2) + x][(ySbIdx \ll 2) + y] = mvL1$

$MvDmvrL0[(xSbIdx \ll 2) + x][(ySbIdx \ll 2) + y] = mvL0$

$MvDmvrL1[(xSbIdx \ll 2) + x][(ySbIdx \ll 2) + y] = mvL1$

$RefIdxL0[(xSbIdx \ll 2) + x][(ySbIdx \ll 2) + y] = refIdxL0$

$RefIdxL1[(xSbIdx \ll 2) + x][(ySbIdx \ll 2) + y] = refIdxL1$

$PredFlagL0[(xSbIdx \ll 2) + x][(ySbIdx \ll 2) + y] = predFlagL0$

$PredFlagL1[(xSbIdx \ll 2) + x][(ySbIdx \ll 2) + y] = predFlagL1$

$BcwIdx[(xSbIdx \ll 2) + x][(ySbIdx \ll 2) + y] = 0$

【0339】

図24は、本発明による復号器の一実施形態を示すブロック図である。デコーダ2400は、プロセッサ2401と、プロセッサ2401に結合され、プロセッサ2401による実行のためのプログラミング命令を記憶する非一時的なコンピュータ可読記憶媒体2402と、を含み、プログラミング命令は、プロセッサによって実行されるときに、復号器に、第1の態様又はその任意の実装形式もしくは実施形態に従って、方法を実行させる。

【0340】

例1. 復号デバイス又は符号化デバイスによって実装されるコーディングの方法であって、

現在のコーディング・ブロックに対する分裂モードを取得することと、

分裂モードに基づいて現在のコーディング・ブロックを2つのサブブロック（サブブロックA、サブブロックB）に分裂させることと、

予め定義された関数又は予め定義されたテーブルに従って、サブブロックAのインデックス値をセットすることと、を含む。

【0341】

例2. 予め定義された関数又は予め定義されたテーブルに対する入力値は、現在のコーディング・ブロックに対する角度インデックス値である（例えば、角度値は、現在のコーディング・ブロックの幾何学的パーティションに使用される）、例1に記載の方法。

【0342】

例3. 予め定義された関数は、

【数41】

$partIdx = angleIdx \geq threshold1 \ \&\& \ angleIdx \leq threshold2 ? 1 : 0$

ここで、 $AngleIdx$ は、現在のコーディング・ブロックに対する角度インデックス値であり、 $partIdx[AngleIdx]$ は、サブブロックAのインデックス値であり、 $threshold1$ 及び $threshold2$ は、整数値であり、 $threshold1$ （ $threshold1$ の値は13）は、 $threshold2$ （閾値2の値は27）より小さい、例1又は2に記載の方法。

【0343】

10

20

30

40

50

【表 7】

例 4：例 1～3 のいずれか 1 つの方法であって、事前定義されたテーブルは、

AngleIdx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
partIdx[A ngleIdx]	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
AngleIdx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
partIdx[A ngleIdx]	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0

10

ここで、AngleIdx は、現在のコーディング・ブロックに対する角度インデックス値であり、partIdx [AngleIdx] は、サブブロック A のインデックス値である。

【0344】

例 5．復号デバイス又は符号化デバイスによって実装されるコーディングの方法であって、

現在のブロックに対する角度パラメータの値を取得することと、

現在のブロックに対する距離パラメータの値を取得することと、

角度パラメータ及び距離パラメータの値に従って、現在のブロックにおける所定のサンプル（例えば、現在のブロックにおける左下、右下、左上又は右上の位置に位置するサンプル）に対するサンプル距離値を計算することと、

20

予め定義されたサンプルに対するサンプル距離値に基づいて、現在のブロック内の 2 つのサブブロックに対する 2 つのインデックス値をセットすることと、を含む、方法。

【0345】

例 6．予め定義されたサンプルに対するサンプル距離の値に基づいて、現在のブロック内の 2 つのサブブロックに対する 2 つのインデックス値をセットすることは、

第 1 の値に等しい、定義されたサンプルが位置する 1 つのサブブロックに対するインデックス値をセットすることと、

30

第 2 の値に等しい、他のサブブロックに対するインデックス値セットすることと、を含み、第 1 の値に対応する処理順序は、第 2 の値に対応する処理順序の前である、例 5 に記載の方法。

【0346】

例 7．サブブロックに対するインデックス値に基づいて、値又は混合マスク（例えば、Mask 1 又は Mask 2）を選択することをさらに含む、例 1～6 のいずれか 1 つに記載の方法。

【0347】

例 8．例 1～7 のいずれか 1 つに記載の方法を実施するための処理回路を含む、復号器（30）又は符号化器。

【0348】

40

例 9．請求項 1～7 のいずれか一項に記載の方法を実行するためのプログラム・コードを含む、コンピュータ・プログラム製品。

【0349】

例 10．復号器又は符号化器であって、

1 つ以上のプロセッサと、

プロセッサに結合され、プロセッサによる実行のためのプログラミング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、プログラミングは、プロセッサによって実行される場合に、例 1～7 のいずれか一項に記載の方法を実施するように復号器を設定する、非一時的なコンピュータ可読記憶媒体と、を含む、復号器又は符号化器。

【0350】

50

数学演算子

この出願で使用される数学演算子は、Cプログラミング言語で使用されるものに類似する。しかし、整数除算演算と算術シフト演算の結果は、より正確に定義され、べき乗や実数値除算などの追加演算が定義される。番号付けとカウントの規則は一般に0から開始する。例えば、「第1」は、0番に等価であり、「第2」は、1番に等価であるなどである。

【0351】

算術演算子

次の算術演算子は、次のように定義される。

+ 加算

− 減算（2引数演算子として）又は否定（単項接頭演算子として）

10

* 行列乗算を含む乗算

x^y べき乗。 x の指数 y のべき乗を指定する。他の文脈では、そのような表記は、べき乗として解釈することを意図しない上付き文字として使用される。

/ 結果がゼロに向かって切り捨てられる整数除算。たとえば、 $7/4$ と $-7/-4$ は1に切り捨てられ、 $-7/4$ と $7/-4$ は−1に切り捨てられる。

÷ 切り捨てや丸めが意図されていない数式の除算を示すために使用される。

【0352】

【数42】

$$\frac{x}{y}$$

20

切り捨てや丸めが意図されていない数式の除算を示すために使用される。

【0353】

【数43】

$$\sum_{i=x}^y f(i)$$

30

x から y までのすべての整数値をとる i の $f(i)$ の和である。

$x \% y$ 剰余。 x の残りを y で割った余りであり、 $x \geq 0$ 及び $y > 0$ の整数 x 及び y に対してのみ定義される。

【0354】

論理演算子

次の論理演算子は、次のように定義される。

$x \& \& y$ x と y のブール論理「and」

$x \mid \mid y$ x と y のブール論理「or」

! ブール論理「否定」

$x ? y : z$ x がTRUE又は0でない場合、 y の値、そうでない場合は z の値で評価される。

40

【0355】

関係演算子

次の関係演算子は、次のように定義される。

> より大きい

>= 以上

< より小さい

<= 以下

== 等しい

!= 等しくない

50

値「na」（適用不可）が割り当てられた構文要素又は変数に関係演算子が適用される場合に、値「na」はその構文要素又は変数の重複しない値として扱われる。値「na」は他の値と等しくないとみなされる。

【0356】

ビット単位演算子

次のビット単位演算子は、次のように定義される。

& ビット単位の「and」である。整数引数について操作する場合に、整数値の2の補数表現について操作する。別の引数よりも少ないビットを含むバイナリ引数について操作する場合に、より短い引数は、0に等しいより大きなビットを加えることによって拡張される。

| ビット単位の「or」である。整数引数について操作する場合に、整数値の2の補数表現について操作する。別の引数よりも少ないビットを含むバイナリ引数について操作する場合に、より短い引数は、0に等しいより大きなビットを加えることによって拡張される。

^ ビット単位の「exclusive or」である。整数引数について操作する場合に、整数値の2の補数表現について操作する。別の引数よりも少ないビットを含むバイナリ引数について操作する場合に、より短い引数は、0に等しいより大きなビットを加えることによって拡張される。

$x \gg y$ x を y バイナリ・ディジットの2の補数整数表現の算術右シフトである。この関数は、 y の負でない整数値に対してのみ定義される。右シフトの結果として最上位ビット(MSB)にシフトされたビットは、シフト演算の前に x のMSBに等しい値を持つ。
 $x \ll y$ x を y バイナリ・ディジットの2の補数整数表現の算術左シフトである。この関数は、 y の負でない整数値に対してのみ定義される。左シフトの結果として最下位ビット(LSB)にシフトされたビットは、0に等しい値を持つ。

【0357】

割り当て演算子

次の代入演算子は、次のように定義される。

= 代入演算子

++ インクリメント、すなわち $x++$ は $x = x + 1$ と等価であり、配列インデックスで 사용되는場合に、インクリメント演算の前に変数の値で評価される。

-- デクリメント、すなわち $x--$ は $x = x - 1$ と等価であり、配列インデックスで 사용되는場合に、デクリメント演算の前に変数の値で評価される。

+= 指定された量だけインクリメントする、すなわち $x += 3$ は $x = x + 3$ と等価であり、 $x += (-3)$ は $x = x + (-3)$ と等価である。

-- 指定された量だけデクリメントする、すなわち $x -= 3$ は $x = x - 3$ と等価であり、 $x -= (-3)$ は $x = x - (-3)$ と等価である。

【0358】

範囲表記

値の範囲を指定するには、次の表記が使用される。

$x = y \dots z$ x は、 y から z までの整数値を取り、 x 、 y 及び z は、整数であり、 z は、 y より大きい。

【0359】

数学関数

次の数学関数が定義される。

【0360】

【数44】

$$\text{Abs}(x) = \begin{cases} x & ; \quad x \geq 0 \\ -x & ; \quad x < 0 \end{cases}$$

10

20

30

40

50

A s i n (x) 正弦の逆三角関数であって、 $-1.0 \sim 1.0$ の範囲の引数 x について操作し、

$-\pi \div 2 \sim \pi \div 2$ (ラジアン単位) の範囲の出力値である。

A t a n (x) 正接の逆三角関数であって、引数 x について操作し、 $-\pi \div 2 \sim \pi \div 2$ (ラジアン単位) の範囲の出力値である。

【0361】

【数45】

$$\text{Atan2}(y, x) = \begin{cases} \text{Atan}\left(\frac{y}{x}\right) & ; \quad x > 0 \\ \text{Atan}\left(\frac{y}{x}\right) + \pi & ; \quad x < 0 \ \&\& \ y \geq 0 \\ \text{Atan}\left(\frac{y}{x}\right) - \pi & ; \quad x < 0 \ \&\& \ y < 0 \\ +\frac{\pi}{2} & ; \quad x = 0 \ \&\& \ y \geq 0 \\ -\frac{\pi}{2} & ; \quad \text{それ以外} \end{cases} \quad 10$$

C e i l (x) x 以上の最小整数である。

【0362】

【数46】

$$\text{Clip1}_Y(x) = \text{Clip3}(0, (1 \ll \text{BitDepth}_Y) - 1, x) \quad 20$$

$$\text{Clip1}_C(x) = \text{Clip3}(0, (1 \ll \text{BitDepth}_C) - 1, x)$$

$$\text{Clip3}(x, y, z) = \begin{cases} x & ; \quad z < x \\ y & ; \quad z > y \\ z & ; \quad \text{それ以外} \end{cases}$$

C o s (x) ラジアン単位の引数 x について操作する余弦の三角関数である。 30

F l o o r (x) x 以下の最大整数である。

【0363】

【数47】

$$\text{GetCurrMsb}(a, b, c, d) = \begin{cases} c + d & ; \quad b - a \geq d/2 \\ c - d & ; \quad a - b > d/2 \\ c & ; \quad \text{それ以外} \end{cases}$$

L n (x) x の自然対数 (底 e 対数、 e は自然基本定数 $2.718281828\dots$)。 40

l o g 2 (x) x の底 2 の対数

L o g 1 0 (x) x の底 10 の対数

【0364】

【数48】

$$\text{Min}(x, y) = \begin{cases} x & ; \quad x \leq y \\ y & ; \quad x > y \end{cases}$$

$$\text{Max}(x, y) = \begin{cases} x & ; \quad x \geq y \\ y & ; \quad x < y \end{cases}$$

$$\text{Round}(x) = \text{Sign}(x) * \text{Floor}(\text{Abs}(x) + 0.5)$$

10

$$\text{Sign}(x) = \begin{cases} 1 & ; \quad x > 0 \\ 0 & ; \quad x == 0 \\ -1 & ; \quad x < 0 \end{cases}$$

S i n (x) ラジアン単位の引数 x について操作する正弦の三角関数である。

【 0 3 6 5 】

【数 4 9 】

20

$$\text{Sqrt}(x) = \sqrt{x}$$

$$\text{Swap}(x, y) = (y, x)$$

T a n (x) ラジアン単位の引数 x について操作する正接の三角関数である。

【 0 3 6 6 】

演算優先度順位

式の優先順位が丸かっこを使用して明示的に示されない場合、次の規則が適用される。

- － より高い優先度の演算は、より低い優先度の演算の前に評価される。
- － 同じ優先度の演算は、左から右へ順次評価される。

30

以下の表は、最高から最低までの演算の優先度を指定しており、表中のより高い位置は、より高い優先度を示す。

Cプログラミング言語でも使用されるこれらの演算子については、本明細書で使用される優先順位は、Cプログラミング言語で使用される優先順位と同じである。

【 0 3 6 7 】

40

50

【表 8】

演算 (オペランド x, y , 及び z)	
" $x++$ ", " $x--$ "	
" $!x$ ", " $-x$ " (単項前置演算子として)	
x^y	10
" $x * y$ ", " x / y ", " $x \div y$ ", " $\frac{x}{y}$ ", " $x \% y$ "	
" $x + y$ ", " $x - y$ " (2つの引数の演算子として), " $\sum_{i=x}^y f(i)$ "	
" $x << y$ ", " $x >> y$ "	20
" $x < y$ ", " $x \leq y$ ", " $x > y$ ", " $x \geq y$ "	
" $x == y$ ", " $x != y$ "	
" $x \& y$ "	
" $x y$ "	
" $x \&\& y$ "	30
" $x y$ "	
" $x ? y : z$ "	
" $x..y$ "	
" $x = y$ ", " $x += y$ ", " $x -= y$ "	40

表：（表の上部）最高から（表の下部）最低までの演算優先度

【0 3 6 8】

論理演算のテキスト記載

本テキストにおいて、次の形式で数学的に記載される論理演算の記述

【0 3 6 9】

【数 5 0】

```

if( condition 0 )
    statement 0
else if( condition 1 )
    statement 1
...
else /* informative remark on remaining condition */
    statement n

```

10

は、以下の方式で記載されてもよい。

【 0 3 7 0 】

【数 5 1】

... as follows / ... the following applies:

- If condition 0, statement 0
- Otherwise, if condition 1, statement 1
- ...
- Otherwise (informative remark on remaining condition), statement n

20

本テキストにおいて、各「If . . . Otherwise, if . . . Otherwise, . . .」文は、直後に「If . . . 」が続く、「. . . as follows」又は「. . . the following applies」で導入される。「If . . . Otherwise, if . . . Otherwise, . . .」の最後の条件は、常に「Otherwise, . . .」である。「If . . . Otherwise, if . . . Otherwise, . . .」文は、「. . . as follows」又は「. . . the following applies」を終わりの「Otherwise, . . .」と一致させることによって識別することができる。本テキストにおいて、次の形式で数学的に記載される論理演算の記述

30

【 0 3 7 1 】

【数 5 2】

```

if( condition 0a  &&  condition 0b )
    statement 0
else if( condition 1a  ||  condition 1b )
    statement 1
...
else
    statement n

```

40

は、以下の方式で記載されてもよい。

【 0 3 7 2 】

50

【数 5 3】

... as follows / ... the following applies:

- If all of the following conditions are true, statement 0:
 - condition 0a
 - condition 0b
- Otherwise, if one or more of the following conditions are true, statement 1:
 - condition 1a
 - condition 1b
- ...
- Otherwise, statement n

10

本テキストにおいて、次の形式で数学的に記載される論理演算の記述

【 0 3 7 3】**【数 5 4】**

20

```
if( condition 0 )  
    statement 0  
if( condition 1 )  
    statement 1
```

は、以下の方式で記載されてもよい。

【 0 3 7 4】

30

【数 5 5】

```
When condition 0, statement 0  
When condition 1, statement 1.
```

【 0 3 7 5】

図 1 8 は、コンテンツ配信サービスを実現するためのコンテンツ供給システム 3 1 0 0 を示すブロック図である。このコンテンツ供給システム 3 1 0 0 は、捕捉デバイス 3 1 0 2、端末デバイス 3 1 0 6 を含み、任意選択で、ディスプレイ 3 1 2 6 を含む。捕捉デバイス 3 1 0 2 は、通信リンク 3 1 0 4 を介して端末デバイス 3 1 0 6 と通信する。通信リンクは、上述の通信チャネル 1 3 を含んでもよい。通信リンク 3 1 0 4 は、W I F I、イーサネット、ケーブル、無線（3 G / 4 G / 5 G）、U S B、又はそれらの任意の種類の組み合わせなどを含むが、これらに限定されない。捕捉デバイス 3 1 0 2 は、データを生成し、上記の実施形態に示す符号化方法によってデータを符号化してもよい。代替的には、捕捉デバイス 3 1 0 2 は、ストリーミングサーバ（図示せず）にデータを配信することができ、サーバは、データを符号化し、符号化されたデータを端末デバイス 3 1 0 6 に送信する。

40

【 0 3 7 6】

捕捉デバイス 3 1 0 2 は、カメラ、スマートフォン又はパッド、コンピュータ若しくは

50

ラップトップ、ビデオ会議システム、PDA、車載デバイス、又はそれらのいずれかの組み合わせなどを含むが、これらに限定されない。例えば、捕捉デバイス3102は、上述のようにソース・デバイス12を含んでもよい。データがビデオを含む場合に、捕捉デバイス3102に含まれるビデオ符号化器20は、実際にビデオ符号化処理を実行することができる。データがオーディオ（すなわち、音声）を含む場合、捕捉デバイス3102に含まれるオーディオ符号化器は、実際にオーディオ符号化処理を実行することができる。いくつかの実際的なシナリオでは、捕捉デバイス3102は、符号化されたビデオ及びオーディオ・データを、それらを一緒に多重化することによって配布する。他の実際的なシナリオ、例えば、ビデオ会議システムにおいては、符号化されたオーディオ・データ及び符号化されたビデオ・データは多重化されない。捕捉デバイス3102は、符号化されたオーディオ・データ及び符号化されたビデオ・データを、端末デバイス3106に別々に配信する。

10

【0377】

コンテンツ供給システム3100では、端末デバイス310は、符号化されたデータを受信及び再生する。端末デバイス3106は、スマートフォン又はPad3108、コンピュータ若しくはラップトップ3110、ネットワーク・ビデオ・レコーダ（NVR）／デジタル・ビデオ・レコーダ（DVR）3112、TV3114、セット・トップ・ボックス3116、ビデオ会議システム3118、ビデオ監視システム3120、パーソナル・デジタル・アシスタント（PDA）3122、車載デバイス3124、又はこれらのいずれかの組み合わせなどの、データ受信及び復元能力を有し、前述の符号化されたデータを復号することが可能なデバイスとすることができる。例えば、捕捉デバイス3106は、上述のようにソース・デバイス14を含んでもよい。符号化されたデータがビデオを含む場合に、端末デバイスに含まれるビデオ復号器30は、ビデオ復号を実行するように優先される。符号化されたデータがオーディオを含む場合に、端末デバイスに含まれるオーディオ復号器は、オーディオ復号処理を実行するように優先される。

20

【0378】

ディスプレイを有する端末デバイス、例えば、スマートフォン又はパッド3108、コンピュータ又はラップトップ3110、ネットワーク・ビデオ・レコーダ（NVR）／デジタル・ビデオ・レコーダ（DVR）3112、TV3114、パーソナル・デジタル・アシスタント（PDA）3122、又は車載デバイス3124では、端末デバイスは、復号されたデータをそのディスプレイに供給することができる。ディスプレイを搭載しないデバイス、例えば、STB3116、ビデオ会議システム3118、又はビデオ監視システム3120では、外部ディスプレイ3126がそこに接触されて、復号されたデータを受信し、示す。

30

【0379】

このシステムの各デバイスが符号化又は復号を行う場合に、前述の実施形態に示すように、ピクチャ符号化デバイス又はピクチャ復号デバイスを使用することができる。

【0380】

図19は、端末デバイス3106の一例の構成を示す図である。端末デバイス3106が、捕捉デバイス3102からストリームを受信した後、プロトコル進行ユニット3202は、ストリームの伝送プロトコルを分析する。このプロトコルは、リアルタイム・ストリーミング・プロトコル（RTSP）、ハイパー・テキスト・トランスファー・プロトコル（HTTP）、HTTPライブ・ストリーミング・プロトコル（HLS）、MPEG-DASH、リアルタイム・トランスポート・プロトコル（RTP）、リアルタイム・メッセージング・プロトコル（RTMP）、又はそれらの任意の種類の組み合わせなどを含むが、これらに限定されない。

40

【0381】

プロトコル進行ユニット3202がストリームを処理した後に、ストリームファイルは、生成される。ファイルは、逆多重化ユニット3204に出力される。逆多重化ユニット3204は、多重化されたデータを符号化されたオーディオ・データと符号化されたビデ

50

オ・データとに分離することができる。上述したように、いくつかの実際的なシナリオ、例えば、ビデオ会議システムにおいては、符号化されたオーディオ・データ及び符号化されたビデオ・データは多重化されない。この状況では、符号化データは、逆多重化部 3204 を介することなく、ビデオ復号器 3206 及びオーディオ復号器 3208 に送信される。

【0382】

逆多重化処理を介して、ビデオ・エレメントリー・ストリーム (ES)、オーディオ ES、及び任意選択で字幕が生成される。前述の実施形態で説明されたビデオ復号器 30 を含むビデオ復号器 3206 は、上述の実施形態に示す復号方法によってビデオ ES を復号して、ビデオ・フレームを生成し、このデータを同期ユニット 3212 に送信する。オーディオ復号器 3208 は、オーディオ ES を復号して、オーディオ・フレームを生成し、このデータを同期ユニット 3212 に送信する。代替的には、ビデオ・フレームは、同期ユニット 3212 に供給する前に、バッファ (図 19 には示さず) に記憶させてもよい。代替的には、オーディオ・フレームは、同期ユニット 3212 に供給する前に、バッファ (図 19 には示さず) に記憶させてもよい。

【0383】

同期ユニット 3212 は、ビデオ・フレームとオーディオ・フレームを同期させ、ビデオ／オーディオ・ディスプレイ 3214 にビデオ／オーディオを供給する。例えば、同期ユニット 3212 は、ビデオ情報及びオーディオ情報の提示を同期させる。情報は、符号化されたオーディオ及びビジュアル・データの提示に関するタイム・スタンプ及びデータストリーム自体の送達に関するタイム・スタンプを使用して、構文でコーディングすることができる。

【0384】

字幕がストリームに含まれる場合、字幕復号器 3210 は、字幕を復号し、それをビデオ・フレーム及びオーディオ・フレームと同期させ、ビデオ／オーディオ／字幕をビデオ／オーディオ／字幕ディスプレイ 3216 に供給する。

【0385】

本発明は、前述のシステムに限定されるものではなく、前述の実施形態におけるピクチャ符号化デバイス又はピクチャ復号デバイスのいずれかを、他のシステム、例えば、自動車システムに組み込むことができる。

【0386】

本発明の実施形態は、主にビデオ・コーディングに基づいて説明されてきたが、コーディング・システム 10、符号化器 20 及び復号器 30 (並びに対応するシステム 10) の実施形態と、本明細書に説明される他の実施形態もまた、静止ピクチャ処理又はコーディング、例えば、ビデオ・コーディングにおけるように任意の先行又は連続するピクチャから独立した個々のピクチャの処理又はコーディングのために構成されてもよいことに留意されたい。一般に、ピクチャ処理コーディングが単一のピクチャ 17 に限定される場合、インター予測ユニット 244 (符号化器) 及び 344 (復号器) のみが利用可能ではないことがある。ビデオ符号化器 20 及びビデオ復号器 30 の他の全ての機能 (ツール又は技術とも呼ばれる) は、例えば、残留計算 204 / 304、変換 206、量子化 208、逆量子化 210 / 310、(逆) 変換 212 / 312、パーティショニング 262 / 362、フレーム内予測 254 / 354、及び／又はループ・フィルタリング 220、320、及びエントロピー符号化 270 及びエントロピー復号 304 は、静止ピクチャ処理のために等しく使用されてもよい。

【0387】

例えば、符号化器 20 及び復号器 30 の実施形態、並びに、例えば、符号化器 20 及び復号器 30 を参照して、本明細書で説明される機能は、ハードウェア、ソフトウェア、ファームウェア、又はそれらの任意の組み合わせで実装されてもよい。ソフトウェアで実装される場合、機能は、コンピュータ可読媒体に記憶されるか、又は 1 つ以上の命令又はコードとして通信媒体を介して送信され、ハードウェアベースの処理ユニットによって実行

される。

【0388】

コンピュータ可読媒体は、データ記憶媒体のような有形媒体に対応するコンピュータ可読記憶媒体、又は、例えば通信プロトコルにしたがって、ある場所から他の場所へのコンピュータ・プログラムの転送を容易にする任意の媒体を含む通信媒体を含んでもよい。このようにして、コンピュータ可読媒体は、一般に、(1) 非一時的である有形のコンピュータ可読記憶媒体、又は(2) 信号又は搬送波などの通信媒体に対応し得る。データ記憶媒体は、本開示で説明される技術の実装のための命令、コード及び／又はデータ構造を検索するために、1つ以上のコンピュータ又は1つ以上のプロセッサによってアクセス可能な任意の利用可能な媒体であり得る。コンピュータ・プログラム製品は、コンピュータ可読記憶媒体を含んでもよい。

10

【0389】

例により、限定するものではないが、このようなコンピュータ可読記憶媒体は、RAM、ROM、EEPROM、CD-ROM又は他の光ディスク記憶デバイス、磁気ディスクストレージ、又は他の磁気記憶デバイス、フラッシュメモリ、又は命令又はデータ構造の形式で所望のプログラム・コードを記憶するために使用することができ、コンピュータによってアクセスすることができる他の任意の媒体を含むことができる。また、任意の接続は、適切にコンピュータ可読媒体とも称される。例えば、同軸ケーブル、光ファイバケーブル、ツイスト・ペア、デジタル加入者線(DSL)、又は赤外線、無線、及びマイクロ波などの無線技術を用いて、ウェブサイト、サーバ、又は他のリモート・ソースから命令が送信される場合、同軸ケーブル、光ファイバケーブル、ツイスト・ペア、又は赤外線、無線、及びマイクロ波のような無線技術は、媒体の定義に含まれる。しかしながら、コンピュータ可読記憶媒体及びデータ記憶媒体は、接続、搬送波、信号、又は他の一時的な媒体を含まず、代わりに非一時的で有形の記憶媒体に向けられると理解されたい。ディスクは、本明細書で使用する場合、コンパクトディスク(CD)、レーザディスク、光ディスク、デジタル多用途ディスク(DVD)、フロッピー・ディスク及びブルーレイ・ディスクを含み、ディスクは、通常、磁氣的にデータを再生し、ディスクは光学的にレーザでデータを再生する。上記の組み合わせはまた、コンピュータ可読媒体の範囲に含まれるべきである。

20

【0390】

命令は、1つ以上のデジタル信号プロセッサ(DSP)、汎用マイクロプロセッサ、特定用途向け集積回路(ASIC)、フィールド・プログラマブル論理アレイ(FPGA)、又は他の同等の集積又は個別論理回路などの1つ以上のプロセッサによって実行されてもよい。したがって、本明細書で使用する用語「プロセッサ」は、前述の構造のいずれか、又は本明細書で説明される技術の実装に好適な他の構造を指してもよい。追加的に、いくつかの態様において、本明細書で説明される機能は、符号化及び復号のために構成される専用ハードウェア及び／又はソフトウェア・モジュール内に提供されてもよく、又は組み合わされたコーデックに組み込まれてもよい。また、この技術は、1つ以上の回路又は論理素子で完全に実装することができる。

30

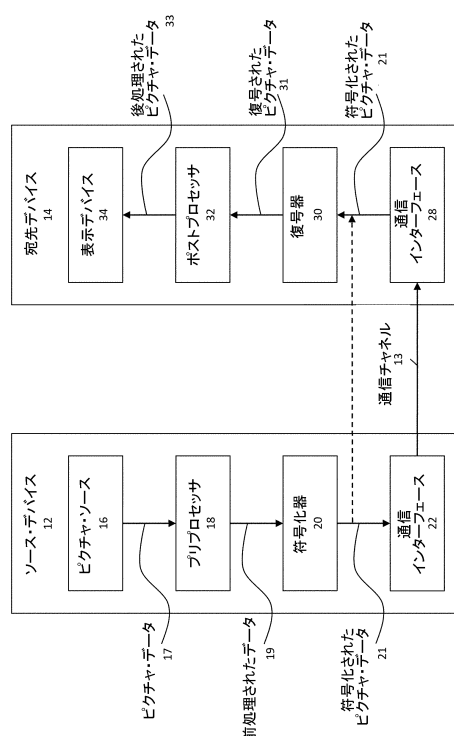
【0391】

本開示の技術は、無線ハンドセット、集積回路(IC)、又はICのセット(例えば、チップセット)を含む、広範な種類のデバイス又は装置で実装されてもよい。本開示では、開示された技術を実行するように構成されているデバイスの機能的態様を強調するために、種々のコンポーネント、モジュール、又はユニットが説明されるが、必ずしも異なるハードウェアユニットによる実現を必要としない。むしろ、上述のように、種々のユニットは、コーデック・ハードウェア・ユニット内で組み合わされてもよく、又は、適切なソフトウェア及び／又はファームウェアと共に、上述の1つ以上のプロセッサを含む、相互動作可能なハードウェアユニットの集合によって提供されてもよい。

40

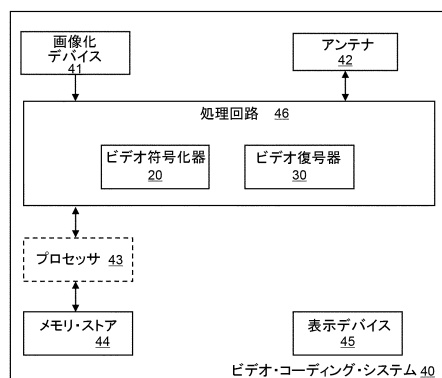
【凶面】

【図 1 A】



10

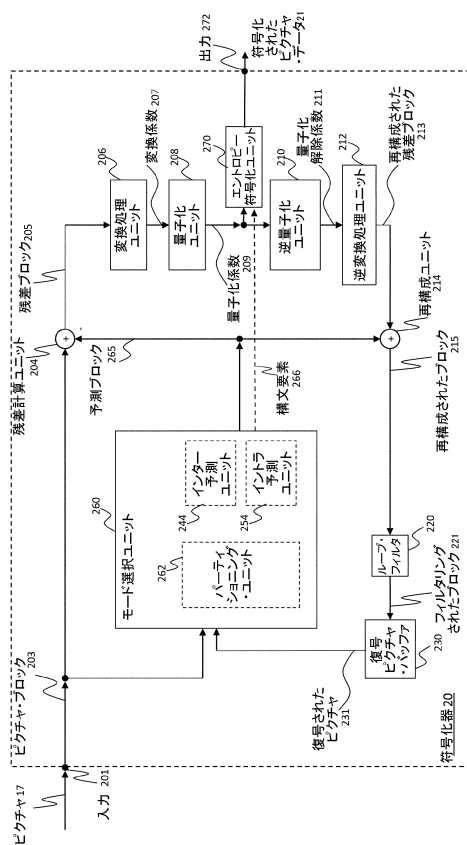
【図 1 B】



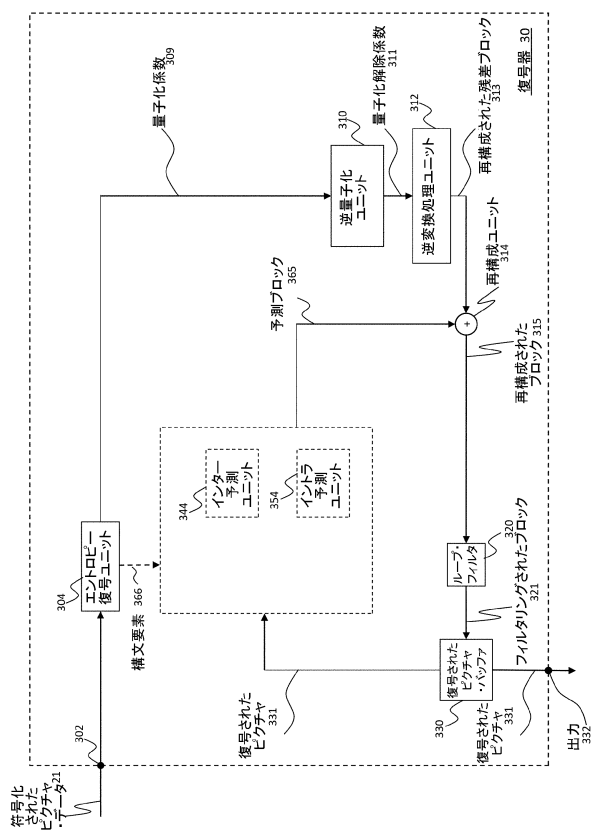
10

20

【図 2】



【図 3】

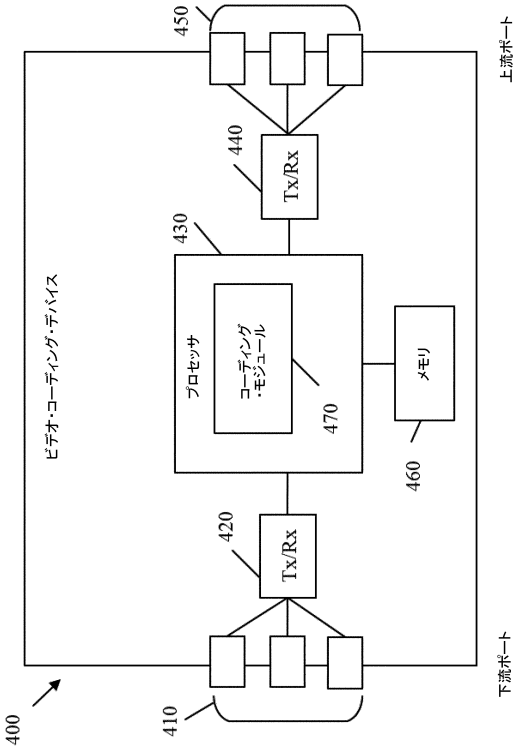


30

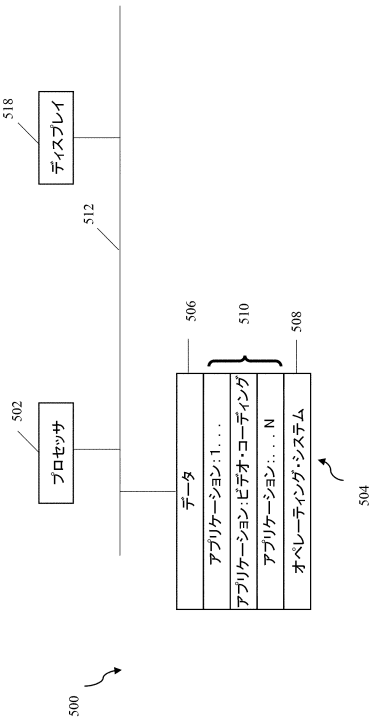
40

50

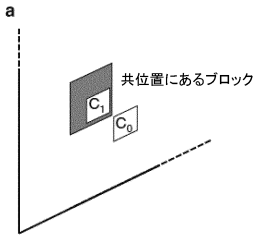
【図 4】



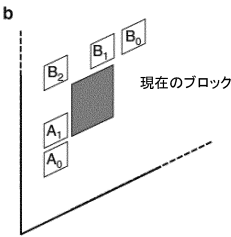
【図 5】



【図 6 a】



【図 6 b】



10

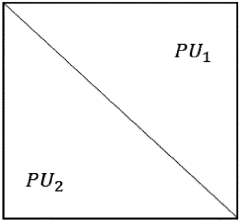
20

30

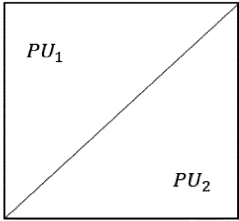
40

50

【図 7】

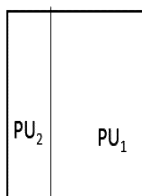
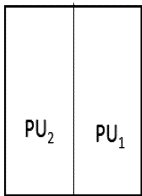
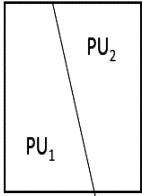


左上のコーナーから
右下のコーナーへの分裂

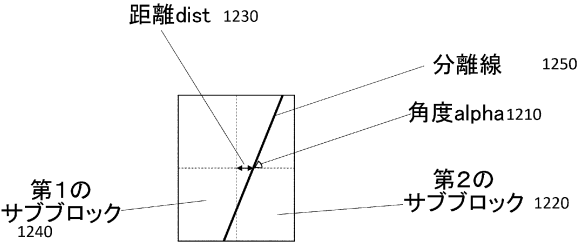


右上のコーナーから
左下のコーナーへの分裂

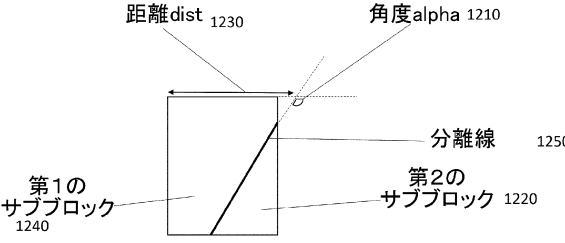
【図 8】



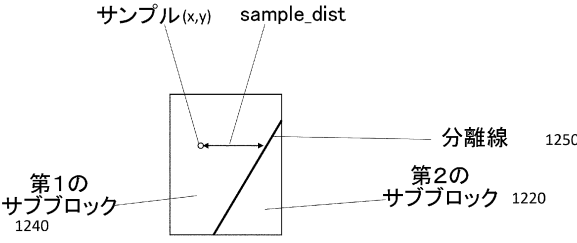
【図 9】



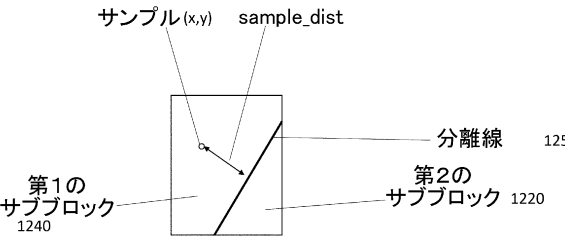
【図 10】



【図 11】



【図 12】



10

20

30

40

50

【図 1 3】

whRatio	alphaN	stepD
0	0	0
0	1	18
0	2	31
0	3	40
0	4	42
0	5	40
0	6	31
0	7	18
0	8	0
1	0	0
1	1	38
1	2	71
1	3	97
1	4	115
1	5	125
1	6	126
1	7	118
1	8	102
2	0	0
2	1	78
2	2	149
2	3	210
2	4	260
2	5	295
2	6	315
2	7	319
2	8	307

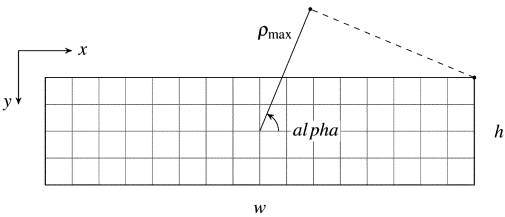
whRatio	alphaN	stepD
3	0	0
3	1	158
3	2	306
3	3	438
3	4	549
3	5	636
3	6	694
3	7	721
3	8	717
4	0	0
4	1	318
4	2	619
4	3	893
4	4	1129
4	5	1317
4	6	1450
4	7	1524
4	8	1536

【図 1 4】

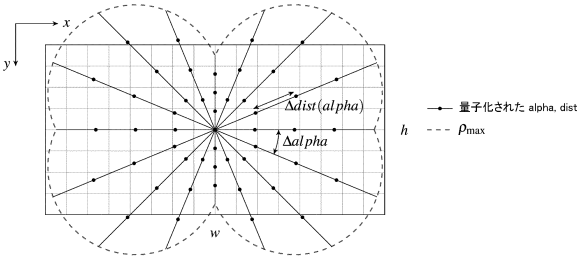
idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
η[idx]	64	63	59	53	45	36	24	12	0	-12	-24	-36	-45	-53	-59	-63
idx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
η[idx]	-64	-63	-59	-53	-45	-36	-24	-12	0	12	24	36	45	53	59	63

10

【図 1 5】



【図 1 6】



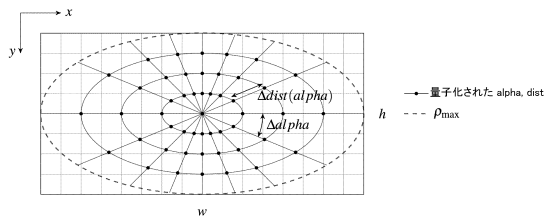
20

30

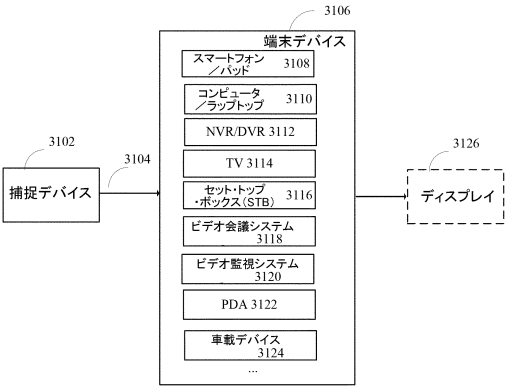
40

50

【図 17】

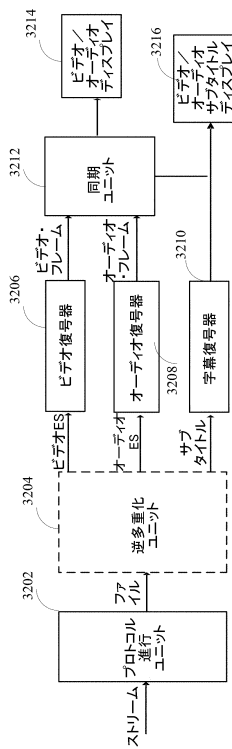


【図 18】

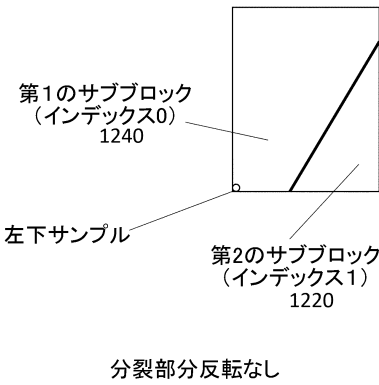


10

【図 19】



【図 20】



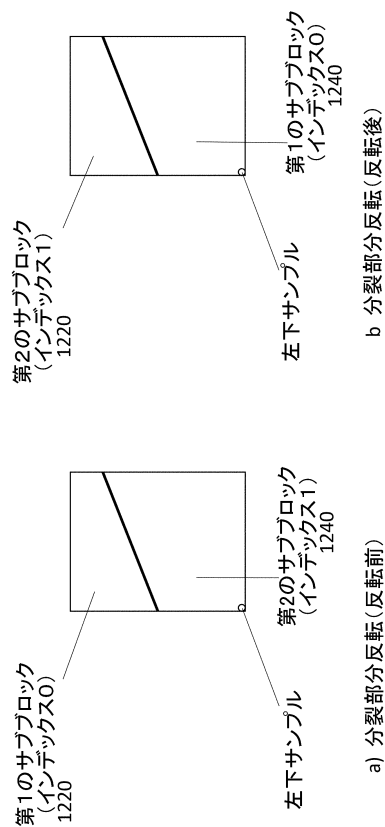
20

30

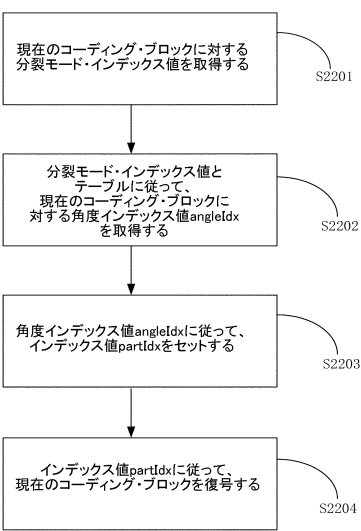
40

50

【図 2 1】



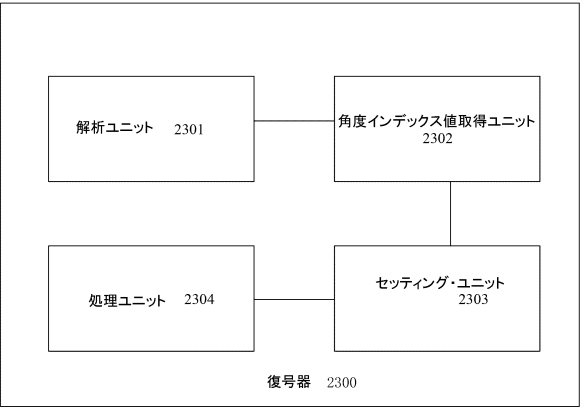
【図 2 2】



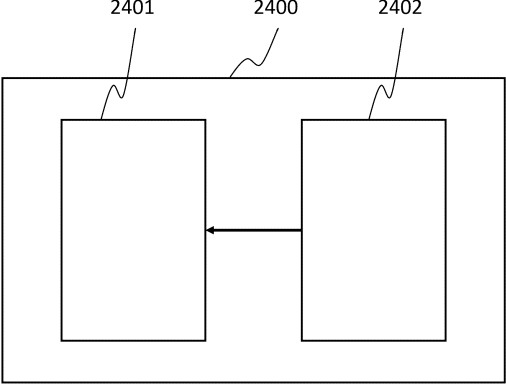
10

20

【図 2 3】



【図 2 4】



30

40

50

フロントページの続き

- (74)代理人 100070150
弁理士 伊東 忠彦
- (74)代理人 100135079
弁理士 宮崎 修
- (72)発明者 ガオ, ハン
ドイツ連邦共和国, ミュンヘン 80992, リースシュトラッセ 25, ホアウェイ・テクノロジーズ・デュッセルドルフ・ゲーエムベーハー内
- (72)発明者 エセンリク, セミ
ドイツ連邦共和国, ミュンヘン 80992, リースシュトラッセ 25, ホアウェイ・テクノロジーズ・デュッセルドルフ・ゲーエムベーハー内
- (72)発明者 アルシナ, エレナ アレクサンドロブナ
ドイツ連邦共和国, ミュンヘン 80992, リースシュトラッセ 25, ホアウェイ・テクノロジーズ・デュッセルドルフ・ゲーエムベーハー内
- (72)発明者 コトラ, アナンド メハー
ドイツ連邦共和国, ミュンヘン 80992, リースシュトラッセ 25, ホアウェイ・テクノロジーズ・デュッセルドルフ・ゲーエムベーハー内
- (72)発明者 ワーン, ビヤオ
ドイツ連邦共和国, ミュンヘン 80992, リースシュトラッセ 25, ホアウェイ・テクノロジーズ・デュッセルドルフ・ゲーエムベーハー内
- 審査官 田部井 和彦
- (56)参考文献 特表2012-531808 (JP, A)
Benjamin Bross et al., Versatile Video Coding (Draft 6) [online], JVET-O2001-vE, インターネット <URL: https://jvet-experts.org/doc_end_user/documents/16_Geneva/wg11/JVET-P0068-v1.zip>, 2019年09月17日, pp.3-5,19,35,71-72,146-148,200-203,227-228,269-273,1-3, JVET-P0068_GEO_CE4-1.1_WD(on_top_of_JVET-O2001-vE).docx
Han Gao et al., CE4-Related: Geometric Merge Mode (GEO) Simplifications [online], JVET-P0107-v2, インターネット <URL: https://jvet-experts.org/doc_end_user/documents/16_Geneva/wg11/JVET-P0107-v2.zip>, 2019年10月02日
Benjamin Bross et al., Versatile Video Coding (Draft 6) [online], JVET-O2001-vE (JVET-P0885), インターネット <URL: https://jvet-experts.org/doc_end_user/documents/16_Geneva/wg11/JVET-P0885-v1.zip>, 2019年10月03日, pp.269-276, JVET_P0885_JontP0107_0246_0304(onTopOf_CE4-1.14).docx
- (58)調査した分野 (Int.Cl., DB名)
H04N 19/119
H04N 19/14
H04N 19/176
H04N 19/70