

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号  
特許第7209829号  
(P7209829)

(45)発行日 令和5年1月20日(2023.1.20)

(24)登録日 令和5年1月12日(2023.1.12)

(51)国際特許分類		F I	
H 0 4 N	19/122 (2014.01)	H 0 4 N	19/122
H 0 4 N	19/157 (2014.01)	H 0 4 N	19/157
H 0 4 N	19/176 (2014.01)	H 0 4 N	19/176
H 0 4 N	19/60 (2014.01)	H 0 4 N	19/60

請求項の数 14 (全55頁)

(21)出願番号	特願2021-525812(P2021-525812)	(73)特許権者	520353802
(86)(22)出願日	令和1年12月4日(2019.12.4)		テンセント・アメリカ・エルエルシー
(65)公表番号	特表2022-507314(P2022-507314 A)		アメリカ合衆国 9 4 3 0 6 カリフォル ニア州 パロアルト パーク・ブルヴァ ード 2 7 4 7
(43)公表日	令和4年1月18日(2022.1.18)	(74)代理人	100110364
(86)国際出願番号	PCT/US2019/064402		弁理士 実広 信哉
(87)国際公開番号	WO2020/117902	(74)代理人	100150197
(87)国際公開日	令和2年6月11日(2020.6.11)		弁理士 松尾 直樹
審査請求日	令和3年5月12日(2021.5.12)	(72)発明者	シン・ジャオ
(31)優先権主張番号	62/776,354		アメリカ合衆国・カリフォルニア・9 4 3 0 6 ・パロ・アルト・パーク・ブル バード・2 7 4 7 ・テンセント・アメリ カ・エルエルシー内
(32)優先日	平成30年12月6日(2018.12.6)	(72)発明者	リャン・ジャオ
(33)優先権主張国・地域又は機関	米国(US)		
(31)優先権主張番号	62/785,681		
(32)優先日	平成30年12月27日(2018.12.27)		
	最終頁に続く		最終頁に続く

(54)【発明の名称】 単一レベルの変換分割および適応サブブロック変換を行うための方法、装置、およびプログラム

(57)【特許請求の範囲】

【請求項1】

ビデオシーケンスを符号化するためにコーディングユニット(CU)を1つまたは複数の変換ユニット(TU)に分割する方法であって、

前記CUの高さを決定するステップと、

前記CUの幅を決定するステップと、

前記CUの前記高さおよび前記CUの前記幅に基づいて、前記1つまたは複数のTUのそれぞれのTUサイズを決定するステップと、

前記CUの前記高さおよび前記CUの前記幅に基づいて、前記1つまたは複数のTUの1つまたは複数のTU位置を決定するステップと、

前記1つまたは複数のTUの前記決定されたサイズおよび前記1つまたは複数のTUの前記1つまたは複数の位置に基づいて、前記CUを前記1つまたは複数のTUに分割するステップと、を含み、

前記1つまたは複数のTUのTUに利用可能な変換タイプの順序は、前記1つまたは複数のTU位置中の前記TUの位置に基づいて決定される、方法。

【請求項2】

前記CUの前記高さは前記CUの前記幅に等しく、

前記1つまたは複数のTUの数は4つであり、

前記1つまたは複数のTUの前記それぞれの高さは、前記CUの前記高さの半分であり、

前記1つまたは複数のTUの前記それぞれの幅は、前記CUの前記幅の半分である、請求項

1に記載の方法。

【請求項 3】

前記CUの前記幅は前記CUの前記高さよりも大きく、  
前記1つまたは複数のTUの数は2つであり、  
前記1つまたは複数のTUのそれぞれの高さは、前記CUの前記高さに等しく、  
前記1つまたは複数のTUの前記それぞれの幅は、前記CUの前記幅の半分である、請求項1に記載の方法。

【請求項 4】

前記CUの前記高さは前記CUの前記幅よりも大きく、  
前記1つまたは複数のTUの数は2つであり、  
前記1つまたは複数のTUのそれぞれの高さは、前記CUの前記高さの半分であり、  
前記1つまたは複数のTUの前記それぞれの幅は、前記CUの前記幅に等しい、請求項1に記載の方法。

【請求項 5】

前記1つまたは複数のTUは、より小さなTUに分割されていない、請求項1に記載の方法。

【請求項 6】

ビデオシーケンスを符号化するためにコーディングユニット(CU)を1つまたは複数の変換ユニット(TU)に分割する方法であって、

前記CUの高さを決定するステップと、

前記CUの幅を決定するステップと、

前記CUの前記高さおよび前記CUの前記幅に基づいて、前記1つまたは複数のTUのそれぞれのTUサイズを決定するステップと、

前記CUの前記高さおよび前記CUの前記幅に基づいて、前記1つまたは複数のTUの1つまたは複数のTU位置を決定するステップと、

前記1つまたは複数のTUの前記決定されたサイズおよび前記1つまたは複数のTUの前記1つまたは複数の位置に基づいて、前記CUを前記1つまたは複数のTUに分割するステップとを含む、

前記CUのインター予測符号化を行うために複数の予測ブロックが使用されることを決定するステップと、

前記複数の予測ブロックの予測サンプル値に基づいて、前記TUサイズおよび前記1つまたは複数のTU位置の中から少なくとも1つを決定するステップと、をさらに含む、  
方法。

【請求項 7】

前記複数の予測ブロックは、第1の予測ブロックおよび第2の予測ブロックを含み、  
前記方法は、

前記第1の予測ブロックから前記第2の予測ブロックを差し引くことによって仮想残差ブロック(VRB)を計算するステップと、

前記VRBの残差エネルギー分布に基づいて、前記TUサイズおよび前記1つまたは複数のTU位置の中から前記少なくとも1つを決定するステップと、をさらに含む、

請求項6に記載の方法。

【請求項 8】

前記1つまたは複数のTU位置は、潜在的なTU位置のサブセットから選択される、請求項6に記載の方法。

【請求項 9】

前記サブセットは、シーケンス・パラメータ集合、ピクチャパラメータ集合、スライスヘッダ、またはタイルヘッダのうちの1つで通知される、請求項8に記載の方法。

【請求項 10】

ビデオシーケンスを符号化するためにコーディングユニット(CU)を1つまたは複数の変換ユニット(TU)に分割する方法であって、

10

20

30

40

50

前記CUの高さを決定するステップと、  
前記CUの幅を決定するステップと、  
前記CUの前記高さおよび前記CUの前記幅に基づいて、前記1つまたは複数のTUのそれぞれのTUサイズを決定するステップと、  
前記CUの前記高さおよび前記CUの前記幅に基づいて、前記1つまたは複数のTUの1つまたは複数のTU位置を決定するステップと、  
前記1つまたは複数のTUの前記決定されたサイズおよび前記1つまたは複数のTUの前記1つまたは複数の位置に基づいて、前記CUを前記1つまたは複数のTUに分割するステップと  
を含み、

前記CUのインター予測符号化を行うために単一の予測ブロックが使用されることを決定するステップと、

10

仮想予測ブロックを配置するステップと、

前記予測ブロックおよび前記仮想予測ブロックに基づいて、前記TUサイズおよび前記1つまたは複数のTU位置の中から少なくとも1つを決定するステップと、をさらに含む、  
 方法。

#### 【請求項 1 1】

前記仮想予測ブロックは、前記CUの動きベクトルをミラーリングし、前記予測ブロックの参照ピクチャとは異なる参照ピクチャ内に前記仮想予測ブロックを配置することによって導出される、請求項 1 0 に記載の方法。

#### 【請求項 1 2】

20

前記CUは第1の参照フレームに関連付けられ、

前記方法は、

前記第1の参照フレームとは異なる第2の参照フレームに関連付けられた隣接ブロックを決定するステップと、

前記CUの位置、前記隣接ブロックの動きベクトル、および前記第2の参照フレームに基づいて前記仮想予測ブロックを導出するステップと、をさらに含む、

請求項 1 0 に記載の方法。

#### 【請求項 1 3】

請求項 1 ~ 1 2 のうちいずれか一項に記載の方法を行うよう構成された装置。

#### 【請求項 1 4】

30

ビデオシーケンスを符号化するためにコーディングユニット (CU) を1つまたは複数の変換ユニット (TU) に分割するための装置の1つまたは複数のコンピュータに、請求項 1 ~ 1 2 のうちいずれか一項に記載の方法を実行させるためのコンピュータプログラム。

#### 【発明の詳細な説明】

#### 【技術分野】

#### 【0 0 0 1】

関連出願の相互参照

本出願は、2018年12月6日に出願された米国仮出願第62 / 776 , 354号、2018年12月27日に米国特許商標庁に出願された米国仮出願第62 / 785 , 681号、および2019年1月25日に米国特許商標庁に提出された米国出願第16 / 694 , 338号に基づく優先権を主張し、これらの開示は参照によりその全体が本明細書に組み込まれる。

40

#### 【0 0 0 2】

本開示は、高効率ビデオ符号化 (HEVC) を超える次世代ビデオ符号化技術、たとえば、多用途ビデオ符号化 (VVC) に関する。より具体的には、1つのコーディングユニット内で複数の変換を適用するための変換スキーム、すなわち変換分割と、予測ブロック情報を使用して変換サイズを決定する適応サブブロック変換が開示される。

#### 【背景技術】

#### 【0 0 0 3】

ITU - T VCEG (Q6 / 16) およびISO / IEC MPEG (JTC 1 / SC 29 / WG 11) は、2013 (バージョン1)、2014 (バージョン2)、2015 (バージョン3) および2016 (バ

50

ージョン4)において、H.265/HEVC(高効率ビデオ符号化)標準を公開した。それ以来、彼らは、HEVC標準(その拡張機能を含む)の圧縮機能を大幅に超える圧縮機能を備えた将来のビデオ符号化技術の標準化の潜在的な必要性を研究してきた。2017年10月、彼らはHEVC(CfP)を超える機能を備えたビデオ圧縮に関する共同提案募集を発表した。2018年2月15日までに、標準ダイナミックレンジ(SDR)で合計22のCfP応答、ハイ・ダイナミック・レンジ(HDR)で12のCfP応答、および360のビデオカテゴリで12のCfP応答がそれぞれ提出された。2018年4月、受領されたすべてのCfP応答は、122 MPEG/第10回JVET(Joint Video Exploration Team - Joint Video Expert Team)会議で評価された。JVETは慎重に評価し、HEVCを超える次世代ビデオ符号化、すなわちいわゆる多用途ビデオ符号化(VVC)の標準化を正式に開始した。VTMの現在のバージョン(VVCテストモデル)、すなわちVTM3。

10

**【発明の概要】****【課題を解決するための手段】****【0004】**

一実施形態によれば、少なくとも1つのプロセッサを使用してビデオシーケンスを符号化するためにコーディングユニット(CU)を1つまたは複数の変換ユニット(TU)に分割する方法は、CUの高さを決定するステップと、CUの幅を決定するステップと、CUの高さおよびCUの幅に基づいて、1つまたは複数のTUのそれぞれのTUサイズを決定するステップと、CUの高さおよびCUの幅に基づいて、1つまたは複数のTUの1つまたは複数のTU位置を決定するステップと、1つまたは複数のTUの決定されたサイズおよび1つまたは複数のTUの1つまたは複数の位置に基づいて、CUを1つまたは複数のTUに分割するステップと、を含む。

20

**【0005】**

一実施形態によれば、ビデオシーケンスを符号化するためにコーディングユニット(CU)を1つまたは複数の変換ユニット(TU)に分割するための装置は、プログラムコードを格納するよう構成された少なくとも1つのメモリと、プログラムコードを読み取り、プログラムコードによって指示するように動作するよう構成された少なくとも1つのプロセッサと、を含み、プログラムコードは、少なくとも1つのプロセッサにCUの高さを決定させるよう構成された第1の決定コードと、少なくとも1つのプロセッサにCUの幅を決定させるよう構成された第2の決定コードと、少なくとも1つのプロセッサに、CUの高さおよびCUの幅に基づいて、1つまたは複数のTUのそれぞれのTUサイズを決定させるよう構成された第3の決定コードと、少なくとも1つのプロセッサに、CUの高さおよびCUの幅に基づいて、1つまたは複数のTUの1つまたは複数のTU位置を決定させるよう構成された第4の決定コードと、少なくとも1つのプロセッサに、1つまたは複数のTUの決定されたサイズおよび1つまたは複数のTUの1つまたは複数の位置に基づいて、CUを1つまたは複数のTUに分割させるよう構成された分割コードと、を含む。

30

**【0006】**

一実施形態によれば、非一時的なコンピュータ可読媒体は1つまたは複数の命令を含む命令を格納し、この命令は、ビデオシーケンスを符号化するためにコーディングユニット(CU)を1つまたは複数の変換ユニット(TU)に分割するための装置の1つまたは複数のプロセッサによって実行されると、1つまたは複数のプロセッサに、CUの高さを決定し、CUの幅を決定し、CUの高さおよびCUの幅に基づいて、1つまたは複数のTUのそれぞれのTUサイズを決定し、CUの高さおよびCUの幅に基づいて、1つまたは複数のTUの1つまたは複数のTU位置を決定し、1つまたは複数のTUの決定されたサイズおよび1つまたは複数のTUの1つまたは複数の位置に基づいて、CUを1つまたは複数のTUに分割させる。

40

**【0007】**

開示される主題のさらなる特徴、性質、および様々な利点は、以下の詳細な説明および添付の図面からより明らかになるであろう。

**【図面の簡単な説明】****【0008】**

50

- 【図 1】一実施形態による、四分木二分木構造の例を示す図である。
- 【図 2 A】一実施形態による、マルチタイプツリー構造の例を示す図である。
- 【図 2 B】一実施形態による、マルチタイプツリー構造の例を示す図である。
- 【図 3】一実施形態による、サブブロック変換モードの例を示す図である。
- 【図 4 A】一実施形態による、正方形ブロックのサブブロック分割の例を示す図である。
- 【図 4 B】一実施形態による、正方形ブロックのサブブロック分割の例を示す図である。
- 【図 5 A】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 5 B】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 5 C】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 5 D】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。 10
- 【図 5 E】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 6 A】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 6 B】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 6 C】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 7 A】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 7 B】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 7 C】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 8 A】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 8 B】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。
- 【図 8 C】一実施形態による、長方形ブロックのサブブロック分割の例を示す図である。 20
- 【図 9 A】一実施形態による、異なる変換集合の適用の例を示す図である。
- 【図 9 B】一実施形態による、異なる変換集合の適用の例を示す図である。
- 【図 9 C】一実施形態による、異なる変換集合の適用の例を示す図である。
- 【図 9 D】一実施形態による、異なる変換集合の適用の例を示す図である。
- 【図 10】本開示の一実施形態による通信システムの簡略化されたブロック図を示す図である。
- 【図 11】ストリーミング環境におけるビデオエンコーダおよびデコーダの配置の図である。
- 【図 12】本開示の一実施形態によるビデオデコーダの機能ブロック図である。
- 【図 13】本開示の一実施形態によるビデオエンコーダの機能ブロック図である。 30
- 【図 14】本開示の一実施形態による、符号化されたビデオビットストリーム内の現在のブロックを符号化するために使用される参照サンプルを選択するための例示的なプロセスのフローチャートである。
- 【図 15】一実施形態によるコンピュータシステムの図である。
- 【発明を実施するための形態】
- 【0009】
- HEVCでは、コーディングツリーユニット（CTU）は、様々なローカル特性に適應するために、コーディングツリーと呼ばれる四分木構造を使用してコーディングユニット（CU）に分割される。ピクチャ間（時間的）またはピクチャ内（空間的）予測を使用してピクチャ領域を符号化するかどうかの決定は、CUレベルで行われる。各CUは、PU分割タイプに応じて、1つ、2つ、または4つの予測ユニット（PU）にさらに分割されうる。1つのPU内で、同じ予測プロセスが適用され、関連する情報はPUベースでデコーダに送信される。PU分割タイプに基づいて予測プロセスを適用して残差ブロックを取得した後、CUのコーディングツリーのような他の四分木構造に従って、CUは変換ユニット（TU）に分割されうる。HEVC構造の特徴は、CU、PU、およびTUを含む複数のパーティションの概念があることである。HEVCでは、CUまたはTUは、正方形のみでありうる、インター予測ブロックの場合、PUは、正方形または長方形であってもよい。HEVCでは、1つの符号化ブロックは、4つの正方形のサブブロックにさらに分割されえ、変換は各サブブロック、すなわちTUに対して行われうる。各TUは、（四分木分割を使用して）再帰的により小さなTUにさらに分割されえ、これは、残差四分木（RQT）と呼ばれる。 40
- 50

## 【 0 0 1 0 】

ピクチャ境界では、HEVCは暗黙的な四分木分割を採用しているため、サイズがピクチャ境界に収まるまでブロックは四分木分割し続ける。

## 【 0 0 1 1 】

HEVCでは、CTUは、コーディングツリーと呼ばれる四分木構造を使用してCUに分割され、様々なローカル特性に適應する。ピクチャ間（時間的）またはピクチャ内（空間的）予測を使用してピクチャ領域を符号化するかどうかの決定は、CUレベルで行われる。各CUは、PU分割タイプに応じて、1つ、2つ、または4つのPUにさらに分割されうる。1つのPU内で、同じ予測プロセスが適用され、関連する情報はPUベースでデコーダに送信される。PU分割タイプに基づいて予測プロセスを適用して残差ブロックを取得した後、CUのコーディングツリーのような他の四分木構造に従って、CUは変換ユニット（TU）に分割されうる。HEVC構造の重要な特徴の1つは、CU、PU、およびTUを含む複数の分割の概念があることである。

10

## 【 0 0 1 2 】

四分木二分木（QTBT）構造は、複数の分割タイプの概念を取り去り、すなわち、CU、PU、およびTUの分離の概念を取り去り、CU分割形状の柔軟性を高める。QTBTブロック構造では、CUは正方形または長方形のいずれかの形状を有しうる。図1に示すように、コーディングツリーユニット（CTU）は、最初に四分木構造によって分割される。四分木リーフノードは、二分木構造によってさらに分割される。二分木分割には、対称水平分割と対称垂直分割の2つの分割タイプがある。二分木リーフノードはコーディングユニット（CU）と呼ばれ、そのセグメンテーションは、さらなる分割なしに、予測および変換処理に使用される。これは、CU、PU、およびTUがQTBT符号化ブロック構造で同じブロックサイズを有することを意味する。JEMでは、CUは異なる色成分の符号化ブロック（CB）で構成される場合があり、たとえば、4：2：0彩度フォーマットのPスライスとBスライスの場合、1つのCUには1つの輝度CBと2つの彩度CBが含まれ、単一の成分のCBで構成される場合もあり、たとえば、Iスライスの場合、1つのCUに1つの輝度CBのみまたは2つの彩度CBだけが含まれる。

20

## 【 0 0 1 3 】

QTBT分割スキームには、次のパラメータが定義されている。

CTU size：四分木のルートノードサイズ、HEVCと同じ概念

MinQTSize：最小許容四分木リーフノードサイズ

MaxBTSize：最大許容二分木ルートノードサイズ

MaxBTDepth：最大許容二分木深さ

MinBTSize：最小許容二分木リーフノードサイズ

30

## 【 0 0 1 4 】

QTBT分割構造の一例では、CTUサイズは128×128輝度サンプルとして設定され、2つの対応する64×64ブロックの彩度サンプルがある。MinQTSizeは16×16として設定され、MaxBTSizeは64×64として設定され、MinBTSize（幅と高さの両方）は4×4に設定され、MaxBTDepthは4に設定される。四分木分割は最初にCTUに適用され、四分木リーフノードが生成される。四分木リーフノードは、16×16（すなわち、MinQTSize）から128×128（すなわち、CTUサイズ）までのサイズを持つことができる。リーフ四分木ノードが128×128の場合、サイズがMaxBTSize（すなわち、64×64）を超えるため、二分木によってさらに分割されることはない。それ以外の場合、リーフ四分木ノードは二分木によってさらに分割される可能性がある。したがって、四分木リーフノードは二分木のルートノードでもあり、二分木の深さは0である。二分木の深さがMaxBTDepth（すなわち、4）に達すると、それ以上の分割は考慮されない。二分木ノードの幅がMinBTSize（すなわち、4）に等しい場合、それ以上の水平分割は考慮されない。同様に、二分木ノードの高さがMinBTSizeに等しい場合、それ以上の垂直分割は考慮されない。二分木のリーフノードは、さらなる分割なしに、予測および変換処理によってさらに処理される。JEMでは、最大CTUサイズは256×256輝度サンプルである。

40

50

## 【 0 0 1 5 】

図1の左側部分は、QTBTを使用することによるブロック分割の例を示し、図1の右側部分は、対応するツリー表現を示している。実線は四分木分割を示し、点線は二分木分割を示す。二分木の各分割（すなわち、非リーフ）ノードでは、使用される分割タイプ（すなわち、水平または垂直）を示す1つのフラグが通知される。ここで、0は水平分割を示し、1は垂直分割を示す。四分木分割の場合、四分木分割は常にブロックを水平方向と垂直方向の両方に分割して、同じサイズの4つのサブブロックを生成するため、分割タイプを指定する必要はない。

## 【 0 0 1 6 】

さらに、QTBTスキームは、輝度と彩度が別々のQTBT構造を持つ柔軟性をサポートする。現在、PスライスとBスライスの場合、1つのCTU内の輝度CTBと彩度CTBは、同じQTBT構造を共有している。ただし、Iスライスの場合、輝度CTBはQTBT構造によってCUに分割され、彩度CTBは他のQTBT構造によって彩度CUに分割される。これは、IスライスのCUが輝度成分の符号化ブロックまたは2つの彩度成分の符号化ブロックで構成され、PスライスまたはBスライスのCUが3つの色成分すべての符号化ブロックで構成されることを意味する。

10

## 【 0 0 1 7 】

HEVCでは、動き補償のメモリアクセスを減らすために、小さなブロックのインター予測が制限されているため、 $4 \times 8$ および $8 \times 4$ ブロックでは二重予測がサポートされておらず、 $4 \times 4$ ブロックではインター予測がサポートされていない。JEM - 7.0で実装されているQTBTでは、これらの制限は取り除かれている。

20

## 【 0 0 1 8 】

マルチタイプツリー（MTT）構造は、QTBTよりも柔軟なツリー構造でありうる。MTTでは、図2Aおよび2Bに示すように、四分木および二分木以外に、水平および垂直の中央側トリプルツリーが導入される。具体的には、図2Aは、垂直方向の中央側トリプルツリー分割の例を示し、図2Bは、水平方向の中央側トリプルツリー分割の例を示している。

## 【 0 0 1 9 】

四分木および二分木が常にブロックの中心に沿って分割している一方で、トリプルツリー分割はブロックの中心にあるオブジェクトをキャプチャできるため、トリプルツリー分割は、四分木および二分木分割を補完しうる。さらに、提案されたトリプルツリーの分割の幅および高さは常に2の累乗であるため、追加の変換は必要ない。

30

## 【 0 0 2 0 】

2レベルのツリーの設計は、複雑さの軽減が動機となる場合がある。理論的には、ツリーの走査の複雑さは $T^D$ である。ここで、Tは分割タイプの番号を示し、Dはツリーの深さを示す。

## 【 0 0 2 1 】

HEVCでは、プライマリ変換は4ポイント、8ポイント、16ポイント、および32ポイントのDCT - 2であり、変換コアマトリックスは、8ビットの整数、すなわち8ビットの変換コアを使用して表される。以下に示すように、小さいDCT - 2の変換コアマトリックスは、大きいDCT - 2の一部である。

40

## 【 0 0 2 2 】

## 【表 1】

表 1 (Table 1) :  $4 \times 4$  変換

```
{64, 64, 64, 64}
{83, 36, -36, -83}
{64, -64, -64, 64}
{36, -83, 83, -36}
```

## 【 0 0 2 3 】

50

【表 2】

表 2 (Table 2) : 8×8 変換

```

{64, 64, 64, 64, 64, 64, 64, 64}
{89, 75, 50, 18, -18, -50, -75, -89}
{83, 36, -36, -83, -83, -36, 36, 83}
{75, -18, -89, -50, 50, 89, 18, -75}
{64, -64, -64, 64, 64, -64, -64, 64}
{50, -89, 18, 75, -75, -18, 89, -50}
{36, -83, 83, -36, -36, 83, -83, 36}
{18, -50, 75, -89, 89, -75, 50, -18}

```

10

【 0 0 2 4】

【表 3】

表 3 (Table 3) : 16×16 変換

```

{64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64}
{90 87 80 70 57 43 25 9 -9-25-43-57-70-80-87-90}
{89 75 50 18-18-50-75-89-89-75-50-18 18 50 75 89}
{87 57 9-43-80-90-70-25 25 70 90 80 43 -9-57-87}
{83 36-36-83-83-36 36 83 83 36-36-83-83-36 36 83}
{80 9-70-87-25 57 90 43-43-90-57 25 87 70 -9-80}
{75-18-89-50 50 89 18-75-75 18 89 50-50-89-18 75}
{70-43-87 9 90 25-80-57 57 80-25-90 -9 87 43-70}
{64-64-64 64 64-64-64 64 64-64-64 64 64-64-64 64}
{57-80-25 90 -9-87 43 70-70-43 87 9-90 25 80-57}
{50-89 18 75-75-18 89-50-50 89-18-75 75 18-89 50}
{43-90 57 25-87 70 9-80 80 -9-70 87-25-57 90-43}
{36-83 83-36-36 83-83 36 36-83 83-36-36 83-83 36}
{25-70 90-80 43 9-57 87-87 57 -9-43 80-90 70-25}
{18-50 75-89 89-75 50-18-18 50-75 89-89 75-50 18}
{ 9-25 43-57 70-80 87-90 90-87 80-70 57-43 25 -9}

```

20

【 0 0 2 5】

30

40

50



au、 at、 as、 ar、 aq、 ap }

{bg、 bj、 bm、 bp、 bs、 bv、 by、 cb、 ce、 ch、 ck、 - ci、 - cf、 - cc、 - bz、 - bw、 - bt、 - bq、 - bn、 - bk、 - bh、 - bf、 - bi、 - bl、 - bo、 - br、 - bu、 - bx、 - ca、 - cd、 - cg、 - cj、 cj、 cg、 cd、 ca、 bx、 bu、 br、 bo、 bl、 bi、 bf、 bh、 bk、 bn、 bq、 bt、 bw、 bz、 cc、 cf、 ci、 - ck、 - ch、 - ce、 - cb、 - by、 - bv、 - bs、 - bp、 - bm、 - bj、 - bg }

{ah、 ai、 aj、 ak、 al、 am、 an、 ao、 - ao、 - an、 - am、 - al、 - ak、 - aj、 - ai、 - ah、 - ah、 - ai、 - aj、 - ak、 - al、 - am、 - an、 - ao、 ao、 an、 am、 al、 ak、 aj、 ai、 ah、 ah、 ai、 aj、 ak、 al、 am、 an、 ao、 - ao、 - an、 - am、 - al、 - ak、 - aj、 - ai、 - ah、 - ah、 - ai、 - aj、 - ak、 - al、 - am、 - an、 - ao、 ao、 an、 am、 al、 ak、 aj、 ai、 ah }

10

{bh、 bm、 br、 bw、 cb、 cg、 - ck、 - cf、 - ca、 - bv、 - bq、 - bl、 - bg、 - bi、 - bn、 - bs、 - bx、 - cc、 - ch、 cj、 ce、 bz、 bu、 bp、 bk、 bf、 bj、 bo、 bt、 by、 cd、 ci、 - ci、 - cd、 - by、 - bt、 - bo、 - bj、 - bf、 - bk、 - bp、 - bu、 - bz、 - ce、 - cj、 ch、 cc、 bx、 bs、 bn、 bi、 bg、 bl、 bq、 bv、 ca、 cf、 ck、 - cg、 - cb、 - bw、 - br、 - bm、 - bh }

{aq、 at、 aw、 az、 bc、 - be、 - bb、 - ay、 - av、 - as、 - ap、 - ar、 - au、 - ax、 - ba、 - bd、 bd、 ba、 ax、 au、 ar、 ap、 as、 av、 ay、 bb、 be、 - bc、 - az、 - aw、 - at、 - aq、 - aq、 - at、 - aw、 - az、 - bc、 be、 bb、 ay、 av、 as、 ap、 ar、 au、 ax、 ba、 bd、 - bd、 - ba、 - ax、 - au、 - ar、 - ap、 - as、 - av、 - ay、 - bb、 - be、 bc、 az、 aw、 at、 aq }

20

{bi、 bp、 bw、 cd、 ck、 - ce、 - bx、 - bq、 - bj、 - bh、 - bo、 - bv、 - cc、 - cj、 - cf、 by、 br、 bk、 bg、 bn、 bu、 cb、 ci、 - cg、 - bz、 - bs、 - bl、 - bf、 - bm、 - bt、 - ca、 - ch、 ch、 ca、 bt、 bm、 bf、 bl、 bs、 bz、 cg、 - ci、 - cb、 - bu、 - bn、 - bg、 - bk、 - br、 - by、 - cf、 cj、 cc、 bv、 bo、 bh、 bj、 bq、 bx、 ce、 - ck、 - cd、 - bw、 - bp、 - bi }

{ad、 ae、 af、 ag、 - ag、 - af、 - ae、 - ad、 - ad、 - ae、 - af、 - ag、 ag、 af、 ae、 ad、 ad、 ae、 af、 ag、 - ag、 - af、 - ae、 - ad、 - ad、 - ae、 - af、 - ag、 ag、 af、 ae、 ad、 ad、 ae、 af、 ag、 - ag、 - af、 - ae、 - ad、 - ad、 - ae、 - af、 - ag、 ag、 af、 ae、 ad }

30

{bj、 bs、 cb、 ck、 - cc、 - bt、 - bk、 - bi、 - br、 - ca、 - cj、 cd、 bu、 bl、 bh、 bq、 bz、 ci、 - ce、 - bv、 - bm、 - bg、 - bp、 - by、 - ch、 cf、 bw、 bn、 bf、 bo、 bx、 cg、 - cg、 - bx、 - bo、 - bf、 - bn、 - bw、 - cf、 ch、 by、 bp、 bg、 bm、 bv、 ce、 - ci、 - bz、 - bq、 - bh、 - bl、 - bu、 - cd、 cj、 ca、 br、 bi、 bk、 bt、 cc、 - ck、 - cb、 - bs、 - bj }

{ar、 aw、 bb、 - bd、 - ay、 - at、 - ap、 - au、 - az、 - be、 ba、 av、 aq、 as、 ax、 bc、 - bc、 - ax、 - as、 - aq、 - av、 - ba、 be、 az、 au、 ap、 at、 ay、 bd、 - bb、 - aw、 - ar、 - ar、 - aw、 - bb、 bd、 ay、 at、 ap、 au、 az、 be、 - ba、 - av、 - aq、 - as、 - ax、 - bc、 bc、 ax、 as、 aq、 av、 ba、 - be、 - az、 - au、 - ap、 - at、 - ay、 - bd、 bb、 aw、 ar }

40

{bk、 bv、 cg、 - ce、 - bt、 - bi、 - bm、 - bx、 - ci、 cc、 br、 bg、 bo、 bz、 ck、 - ca、 - bp、 - bf、 - bq、 - cb、 cj、 by、 bn、 bh、 bs、 cd、 - ch、 - bw、 - bl、 - bj、 - bu、 - cf、 cf、 bu、 bj、 bl、 bw、 ch、 - cd、 - bs、 - bh、 - bn、 - by、 - cj、 cb、 bq、 bf、 bp、 ca、 - ck、 - bz、 - bo、 - bg、 - br、 - cc、 ci、 bx、 bm、 bi、 bt、 ce、 - cg、 - bv、 - bk }

{ai、 al、 ao、 - am、 - aj、 - ah、 - ak、 - an、 an、 ak、 ah、 aj、 am、 - ao、 - al、 - ai、 - ai、 - al、 - ao、 am、 aj、 ah、 ak、 an、 - an、 - ak、 - ah、 - aj、 - am、 ao、 al、 ai、 ai、 al、 ao、 - am、 - aj、 - ah、 - ak、 - an、 an、 ak、 ah、 aj、 am、 - ao、 - al、 - ai、 - ai、 - al、 - ao、 am、 aj、 ah、 ak、 an、 - an、 - ak、 - ah、 - aj }

50

、 - am、 ao、 al、 ai }

{ bl、 by、 - ck、 - bx、 - bk、 - bm、 - bz、 cj、 bw、 bj、 bn、 ca、 - ci、 - bv、 - bi、 - bo、 - cb、 ch、 bu、 bh、 bp、 cc、 - cg、 - bt、 - bg、 - bq、 - cd、 cf、 bs、 b f、 br、 ce、 - ce、 - br、 - bf、 - bs、 - cf、 cd、 bq、 bg、 bt、 cg、 - cc、 - bp、 - b h、 - bu、 - ch、 cb、 bo、 bi、 bv、 ci、 - ca、 - bn、 - bj、 - bw、 - cj、 bz、 bm、 bk 、 bx、 ck、 - by、 - bl }

{ as、 az、 - bd、 - aw、 - ap、 - av、 - bc、 ba、 at、 ar、 ay、 - be、 - ax、 - aq、 - au、 - bb、 bb、 au、 aq、 ax、 be、 - ay、 - ar、 - at、 - ba、 bc、 av、 ap、 aw、 bd 、 - az、 - as、 - as、 - az、 bd、 aw、 ap、 av、 bc、 - ba、 - at、 - ar、 - ay、 be、 ax 、 aq、 au、 bb、 - bb、 - au、 - aq、 - ax、 - be、 ay、 ar、 at、 ba、 - bc、 - av、 - a p、 - aw、 - bd、 az、 as }

10

{ bm、 cb、 - cf、 - bq、 - bi、 - bx、 cj、 bu、 bf、 bt、 ci、 - by、 - bj、 - bp、 - c e、 cc、 bn、 bl、 ca、 - cg、 - br、 - bh、 - bw、 ck、 bv、 bg、 bs、 ch、 - bz、 - bk、 - bo、 - cd、 cd、 bo、 bk、 bz、 - ch、 - bs、 - bg、 - bv、 - ck、 bw、 bh、 br、 cg、 - ca、 - bl、 - bn、 - cc、 ce、 bp、 bj、 by、 - ci、 - bt、 - bf、 - bu、 - cj、 bx、 bi、 bq、 cf、 - cb、 - bm }

{ ab、 ac、 - ac、 - ab、 - ab、 - ac、 ac、 ab、 ab、 ac、 - ac、 - ab、 - ab、 - ac、 ac、 ab、 ab、 ac、 - ac、 - ab、 - ab、 - ab、 - ac、 ac、 ab、 ab、 ac、 - ac、 - ab、 - ab、 - a b、 - ac、 ac、 ab、 ab、 ac、 - ac、 - ab、 - ab、 - ac、 ac、 ab、 ab、 ac、 - ac、 - ab 、 - ab、 - ac、 ac、 ab }

20

{ bn、 ce、 - ca、 - bj、 - br、 - ci、 bw、 bf、 bv、 - cj、 - bs、 - bi、 - bz、 cf、 b o、 bm、 cd、 - cb、 - bk、 - bq、 - ch、 bx、 bg、 bu、 - ck、 - bt、 - bh、 - by、 cg 、 bp、 bl、 cc、 - cc、 - bl、 - bp、 - cg、 by、 bh、 bt、 ck、 - bu、 - bg、 - bx、 ch、 bq、 bk、 cb、 - cd、 - bm、 - bo、 - cf、 bz、 bi、 bs、 cj、 - bv、 - bf、 - bw、 ci、 br 、 bj、 ca、 - ce、 - bn }

{ at、 bc、 - ay、 - ap、 - ax、 bd、 au、 as、 bb、 - az、 - aq、 - aw、 be、 av、 ar 、 ba、 - ba、 - ar、 - av、 - be、 aw、 aq、 az、 - bb、 - as、 - au、 - bd、 ax、 ap、 a y、 - bc、 - at、 - at、 - bc、 ay、 ap、 ax、 - bd、 - au、 - as、 - bb、 az、 aq、 aw、 - be、 - av、 - ar、 - ba、 ba、 ar、 av、 be、 - aw、 - aq、 - az、 bb、 as、 au、 bd、 - ax、 - ap、 - ay、 bc、 at }

30

{ bo、 ch、 - bv、 - bh、 - ca、 cc、 bj、 bt、 - cj、 - bq、 - bm、 - cf、 bx、 bf、 by 、 - ce、 - bl、 - br、 - ck、 bs、 bk、 cd、 - bz、 - bg、 - bw、 cg、 bn、 bp、 ci、 - b u、 - bi、 - cb、 cb、 bi、 bu、 - ci、 - bp、 - bn、 - cg、 bw、 bg、 bz、 - cd、 - bk、 - bs、 ck、 br、 bl、 ce、 - by、 - bf、 - bx、 cf、 bm、 bq、 cj、 - bt、 - bj、 - cc、 ca 、 bh、 bv、 - ch、 - bo }

{ aj、 ao、 - ak、 - ai、 - an、 al、 ah、 am、 - am、 - ah、 - al、 an、 ai、 ak、 - ao 、 - aj、 - aj、 - ao、 ak、 ai、 an、 - al、 - ah、 - am、 am、 ah、 al、 - an、 - ai、 - a k、 ao、 aj、 aj、 ao、 - ak、 - ai、 - an、 al、 ah、 am、 - am、 - ah、 - al、 an、 ai、 a k、 - ao、 - aj、 - aj、 - ao、 ak、 ai、 an、 - al、 - ah、 - am、 am、 ah、 al、 - an、 - ai、 - ak、 ao、 aj }

40

{ bp、 ck、 - bq、 - bo、 - cj、 br、 bn、 ci、 - bs、 - bm、 - ch、 bt、 bl、 cg、 - bu 、 - bk、 - cf、 bv、 bj、 ce、 - bw、 - bi、 - cd、 bx、 bh、 cc、 - by、 - bg、 - cb、 bz 、 bf、 ca、 - ca、 - bf、 - bz、 cb、 bg、 by、 - cc、 - bh、 - bx、 cd、 bi、 bw、 - ce、 - bj、 - bv、 cf、 bk、 bu、 - cg、 - bl、 - bt、 ch、 bm、 bs、 - ci、 - bn、 - br、 cj、 bo、 bq、 - ck、 - bp }

{ au、 - be、 - at、 - av、 bd、 as、 aw、 - bc、 - ar、 - ax、 bb、 aq、 ay、 - ba、 - ap、 - az、 az、 ap、 ba、 - ay、 - aq、 - bb、 ax、 ar、 bc、 - aw、 - as、 - bd、 av、 a t、 be、 - au、 - au、 be、 at、 av、 - bd、 - as、 - aw、 bc、 ar、 ax、 - bb、 - aq、 - ay、 ba、 ap、 az、 - az、 - ap、 - ba、 ay、 aq、 bb、 - ax、 - ar、 - bc、 aw、 as、 bd

50

{ - av, - at, - be, au }

{ bq, - ci, - bl, - bv, cd, bg, ca, - by, - bi, - cf, bt, bn, ck, - bo, - b  
s, cg, bj, bx, - cb, - bf, - cc, bw, bk, ch, - br, - bp, cj, bm, bu, - ce,  
- bh, - bz, bz, bh, ce, - bu, - bm, - cj, bp, br, - ch, - bk, - bw, cc, bf  
、 cb, - bx, - bj, - cg, bs, bo, - ck, - bn, - bt, cf, bi, by, - ca, - bg, - c  
d, bv, bl, ci, - bq }

{ ae, - ag, - ad, - af, af, ad, ag, - ae, - ae, ag, ad, af, - af, - ad, - a  
g, ae, ae, - ag, - ad, - af, af, ad, ag, - ae, - ae, ag, ad, af, - af, - ad  
、 - ag, ae, ae, - ag, - ad, - af, af, ad, ag, - ae, - ae, ag, ad, af, - af,  
- ad, - ag, ae, ae, - ag, - ad, - af, af, ad, ag, - ae, - ae, ag, ad, af, -  
af, - ad, - ag, ae }

10

{ br, - cf, - bg, - cc, bu, bo, - ci, - bj, - bz, bx, bl, ck, - bm, - bw,  
ca, bi, ch, - bp, - bt, cd, bf, ce, - bs, - bq, cg, bh, cb, - bv, - bn, cj,  
bk, by, - by, - bk, - cj, bn, bv, - cb, - bh, - cg, bq, bs, - ce, - bf, - c  
d, bt, bp, - ch, - bi, - ca, bw, bm, - ck, - bl, - bx, bz, bj, ci, - bo, - b  
u, cc, bg, cf, - br }

{ av, - bb, - ap, - bc, au, aw, - ba, - aq, - bd, at, ax, - az, - ar, - b  
e, as, ay, - ay, - as, be, ar, az, - ax, - at, bd, aq, ba, - aw, - au, bc,  
ap, bb, - av, - av, bb, ap, bc, - au, - aw, ba, aq, bd, - at, - ax, az, ar  
、 be, - as, - ay, ay, as, - be, - ar, - az, ax, at, - bd, - aq, - ba, aw, au  
、 - bc, - ap, - bb, av }

20

{ bs, - cc, - bi, - cj, bl, bz, - bv, - bp, cf, bf, cg, - bo, - bw, by, bm  
、 - ci, - bh, - cd, br, bt, - cb, - bj, - ck, bk, ca, - bu, - bq, ce, bg, ch  
、 - bn, - bx, bx, bn, - ch, - bg, - ce, bq, bu, - ca, - bk, ck, bj, cb, - b  
t, - br, cd, bh, ci, - bm, - by, bw, bo, - cg, - bf, - cf, bp, bv, - bz, -  
bl, cj, bi, cc, - bs }

{ ak, - am, - ai, ao, ah, an, - aj, - al, al, aj, - an, - ah, - ao, ai, am  
、 - ak, - ak, am, ai, - ao, - ah, - an, aj, al, - al, - aj, an, ah, ao, - ai  
、 - am, ak, ak, - am, - ai, ao, ah, an, - aj, - al, al, aj, - an, - ah, - ao  
、 ai, am, - ak, - ak, am, ai, - ao, - ah, - an, aj, al, - al, - aj, an, ah,  
ao, - ai, - am, ak }

30

{ bt, - bz, - bn, cf, bh, ck, - bi, - ce, bo, by, - bu, - bs, ca, bm, - c  
g, - bg, - cj, bj, cd, - bp, - bx, bv, br, - cb, - bl, ch, bf, ci, - bk, - cc  
、 bq, bw, - bw, - bq, cc, bk, - ci, - bf, - ch, bl, cb, - br, - bv, bx, bp  
、 - cd, - bj, cj, bg, cg, - bm, - ca, bs, bu, - by, - bo, ce, bi, - ck, - bh  
、 - cf, bn, bz, - bt }

{ aw, - ay, - au, ba, as, - bc, - aq, be, ap, bd, - ar, - bb, at, az, - a  
v, - ax, ax, av, - az, - at, bb, ar, - bd, - ap, - be, aq, bc, - as, - ba, a  
u, ay, - aw, - aw, ay, au, - ba, - as, bc, aq, - be, - ap, - bd, ar, bb,  
- at, - az, av, ax, - ax, - av, az, at, - bb, - ar, bd, ap, be, - aq, - bc,  
as, ba, - au, - ay, aw }

40

{ bu, - bw, - bs, by, bq, - ca, - bo, cc, bm, - ce, - bk, cg, bi, - ci,  
- bg, ck, bf, cj, - bh, - ch, bj, cf, - bl, - cd, bn, cb, - bp, - bz, br, bx  
、 - bt, - bv, bv, bt, - bx, - br, bz, bp, - cb, - bn, cd, bl, - cf, - bj, ch  
、 bh, - cj, - bf, - ck, bg, ci, - bi, - cg, bk, ce, - bm, - cc, bo, ca, - bq  
、 - by, bs, bw, - bu }

{ aa, - aa, - aa, aa, aa, - aa, - aa, aa, aa, - aa, - aa, aa, aa, - aa, -  
aa, aa, aa, - aa, - aa, aa, aa, - aa, - aa, aa, aa, - aa, - aa, aa, aa, - aa  
、 - aa, aa, aa, - aa, - aa, aa, aa, - aa, - aa, aa, aa, - aa, - aa, aa, aa,  
- aa, - aa, aa, aa, - aa, - aa, aa, aa, - aa, - aa, aa, aa, - aa, - aa, aa,

50

aa、 - aa、 - aa、 aa }

{bv、 - bt、 - bx、 br、 bz、 - bp、 - cb、 bn、 cd、 - bl、 - cf、 bj、 ch、 - bh、 - cj、 bf、 - ck、 - bg、 ci、 bi、 - cg、 - bk、 ce、 bm、 - cc、 - bo、 ca、 bq、 - by、 - bs、 bw、 bu、 - bu、 - bw、 bs、 by、 - bq、 - ca、 bo、 cc、 - bm、 - ce、 bk、 cg、 - bi、 - ci、 bg、 ck、 - bf、 cj、 bh、 - ch、 - bj、 cf、 bl、 - cd、 - bn、 cb、 bp、 - bz、 - br、 bx、 bt、 - bv }

{ax、 - av、 - az、 at、 bb、 - ar、 - bd、 ap、 - be、 - aq、 bc、 as、 - ba、 - au、 ay、 aw、 - aw、 - ay、 au、 ba、 - as、 - bc、 aq、 be、 - ap、 bd、 ar、 - bb、 - at、 az、 av、 - ax、 - ax、 av、 az、 - at、 - bb、 ar、 bd、 - ap、 be、 aq、 - bc、 - as、 ba、 au、 - ay、 - aw、 aw、 ay、 - au、 - ba、 as、 bc、 - aq、 - be、 ap、 - bd、 - ar、 b、 b、 at、 - az、 - av、 ax }

10

{bw、 - bq、 - cc、 bk、 ci、 - bf、 ch、 bl、 - cb、 - br、 bv、 bx、 - bp、 - cd、 bj、 - cj、 - bg、 cg、 bm、 - ca、 - bs、 bu、 by、 - bo、 - ce、 bi、 ck、 - bh、 cf、 bn、 - bz、 - bt、 bt、 bz、 - bn、 - cf、 bh、 - ck、 - bi、 ce、 bo、 - by、 - bu、 bs、 ca、 - bm、 - cg、 bg、 - cj、 - bj、 cd、 bp、 - bx、 - bv、 br、 cb、 - bl、 - ch、 bf、 - ci、 - bk、 cc、 bq、 - bw }

{al、 - aj、 - an、 ah、 - ao、 - ai、 am、 ak、 - ak、 - am、 ai、 ao、 - ah、 an、 aj、 - al、 - al、 aj、 an、 - ah、 ao、 ai、 - am、 - ak、 ak、 am、 - ai、 - ao、 ah、 - an、 - aj、 al、 al、 - aj、 - an、 ah、 - ao、 - ai、 am、 ak、 - ak、 - am、 ai、 ao、 - ah、 an、 aj、 - al、 - al、 aj、 an、 - ah、 ao、 ai、 - am、 - ak、 ak、 am、 - ai、 - ao、 ah、 - an、 - aj、 al }

20

{bx、 - bn、 - ch、 bg、 - ce、 - bq、 bu、 ca、 - bk、 - ck、 bj、 - cb、 - bt、 br、 cd、 - bh、 ci、 bm、 - by、 - bw、 bo、 cg、 - bf、 cf、 bp、 - bv、 - bz、 bl、 cj、 - bi、 cc、 bs、 - bs、 - cc、 bi、 - cj、 - bl、 bz、 bv、 - bp、 - cf、 bf、 - cg、 - bo、 bw、 by、 - bm、 - ci、 bh、 - cd、 - br、 bt、 cb、 - bj、 ck、 bk、 - ca、 - bu、 bq、 ce、 - bg、 ch、 bn、 - bx }

{ay、 - as、 - be、 ar、 - az、 - ax、 at、 bd、 - aq、 ba、 aw、 - au、 - bc、 ap、 - bb、 - av、 av、 bb、 - ap、 bc、 au、 - aw、 - ba、 aq、 - bd、 - at、 ax、 az、 - ar、 b、 e、 as、 - ay、 - ay、 as、 be、 - ar、 az、 ax、 - at、 - bd、 aq、 - ba、 - aw、 au、 bc、 - ap、 bb、 av、 - av、 - bb、 ap、 - bc、 - au、 aw、 ba、 - aq、 bd、 at、 - ax、 - a、 z、 ar、 - be、 - as、 ay }

30

{by、 - bk、 cj、 bn、 - bv、 - cb、 bh、 - cg、 - bq、 bs、 ce、 - bf、 cd、 bt、 - bp、 - ch、 bi、 - ca、 - bw、 bm、 ck、 - bl、 bx、 bz、 - bj、 ci、 bo、 - bu、 - cc、 bg、 - cf、 - br、 br、 cf、 - bg、 cc、 bu、 - bo、 - ci、 bj、 - bz、 - bx、 bl、 - ck、 - bm、 bw、 ca、 - bi、 ch、 bp、 - bt、 - cd、 bf、 - ce、 - bs、 bq、 cg、 - bh、 cb、 bv、 - bn、 - cj、 bk、 - by }

{af、 - ad、 ag、 ae、 - ae、 - ag、 ad、 - af、 - af、 ad、 - ag、 - ae、 ae、 ag、 - ad、 af、 af、 - ad、 ag、 ae、 - ae、 - ag、 ad、 - af、 - af、 ad、 - ag、 - ae、 ae、 ag、 - ad、 af、 af、 - ad、 ag、 ae、 - ae、 - ag、 ad、 - af、 - af、 ad、 - ag、 - ae、 ae、 ag、 - ad、 af }

40

{bz、 - bh、 ce、 bu、 - bm、 cj、 bp、 - br、 - ch、 bk、 - bw、 - cc、 bf、 - cb、 - bx、 bj、 - cg、 - bs、 bo、 ck、 - bn、 bt、 cf、 - bi、 by、 ca、 - bg、 cd、 bv、 - bl、 ci、 bq、 - bq、 - ci、 bl、 - bv、 - cd、 bg、 - ca、 - by、 bi、 - cf、 - bt、 bn、 - ck、 - bo、 bs、 cg、 - bj、 bx、 cb、 - bf、 cc、 bw、 - bk、 ch、 br、 - bp、 - cj、 bm、 - bu、 - ce、 bh、 - bz }

{az、 - ap、 ba、 ay、 - aq、 bb、 ax、 - ar、 bc、 aw、 - as、 bd、 av、 - at、 be、 a、 u、 - au、 - be、 at、 - av、 - bd、 as、 - aw、 - bc、 ar、 - ax、 - bb、 aq、 - ay、 - ba、 ap、 - az、 - az、 ap、 - ba、 - ay、 aq、 - bb、 - ax、 ar、 - bc、 - aw、 as、 - b、 d、 - av、 at、 - be、 - au、 au、 be、 - at、 av、 bd、 - as、 aw、 bc、 - ar、 ax、 bb、

50



- aj, ah, - ak, an }

{ cf, - bu, bj, - bl, bw, - ch, - cd, bs, - bh, bn, - by, cj, cb, - bq, bf, - bp, ca, ck, - bz, bo, - bg, br, - cc, - ci, bx, - bm, bi, - bt, ce, cg, - bv, bk, - bk, bv, - cg, - ce, bt, - bi, bm, - bx, ci, cc, - br, bg, - bo, bz, - ck, - ca, bp, - bf, bq, - cb, - cj, by, - bn, bh, - bs, cd, ch, - bw, bl, - bj, bu, - cf }

{ bc, - ax, as, - aq, av, - ba, - be, az, - au, ap, - at, ay, - bd, - bb, aw, - ar, ar, - aw, bb, bd, - ay, at, - ap, au, - az, be, ba, - av, aq, - as, ax, - bc, - bc, ax, - as, aq, - av, ba, be, - az, au, - ap, at, - ay, bd, - bb, - aw, ar, - ar, aw, - bb, - bd, ay, - at, ap, - au, az, - be, - ba, a v, - aq, as, - ax, bc }

10

{ cg, - bx, bo, - bf, bn, - bw, cf, ch, - by, bp, - bg, bm, - bv, ce, ci, - bz, bq, - bh, bl, - bu, cd, cj, - ca, br, - bi, bk, - bt, cc, ck, - cb, bs, - bj, bj, - bs, cb, - ck, - cc, bt, - bk, bi, - br, ca, - cj, - cd, bu, - bl, bh, - bq, bz, - ci, - ce, bv, - bm, bg, - bp, by, - ch, - cf, bw, - b n, bf, - bo, bx, - cg }

{ ag, - af, ae, - ad, ad, - ae, af, - ag, - ag, af, - ae, ad, - ad, ae, - af, ag, ag, - af, ae, - ad, ad, - ae, af, - ag, - ag, af, - ae, ad, - ad, ae, - af, ag, ag, - af, ae, - ad, ad, - ae, af, - ag, - ag, af, - ae, ad, - ad, ae, - af, ag, ag, - af, ae, - ad, ad, - ae, af, - ag, - ag, af, - ae, ad, - ad, ae, - af, ag }

20

{ ch, - ca, bt, - bm, bf, - bl, bs, - bz, cg, ci, - cb, bu, - bn, bg, - bk, br, - by, cf, cj, - cc, bv, - bo, bh, - bj, bq, - bx, ce, ck, - cd, bw, - bp, bi, - bi, bp, - bw, cd, - ck, - ce, bx, - bq, bj, - bh, bo, - bv, cc, - cj, - cf, by, - br, bk, - bg, bn, - bu, cb, - ci, - cg, bz, - bs, bl, - bf, bm, - bt, ca, - ch }

{ bd, - ba, ax, - au, ar, - ap, as, - av, ay, - bb, be, bc, - az, aw, - a t, aq, - aq, at, - aw, az, - bc, - be, bb, - ay, av, - as, ap, - ar, au, - a x, ba, - bd, - bd, ba, - ax, au, - ar, ap, - as, av, - ay, bb, - be, - bc, az, - aw, at, - aq, aq, - at, aw, - az, bc, be, - bb, ay, - av, as, - ap, a r, - au, ax, - ba, bd }

30

{ ci, - cd, by, - bt, bo, - bj, bf, - bk, bp, - bu, bz, - ce, cj, ch, - cc, bx, - bs, bn, - bi, bg, - bl, bq, - bv, ca, - cf, ck, cg, - cb, bw, - br, bm, - bh, bh, - bm, br, - bw, cb, - cg, - ck, cf, - ca, bv, - bq, bl, - bg, bi, - bn, bs, - bx, cc, - ch, - cj, ce, - bz, bu, - bp, bk, - bf, bj, - bo, bt, - by, cd, - ci }

{ ao, - an, am, - al, ak, - aj, ai, - ah, ah, - ai, aj, - ak, al, - am, an, - ao, - ao, an, - am, al, - ak, aj, - ai, ah, - ah, ai, - aj, ak, - al, am, - an, ao, ao, - an, am, - al, ak, - aj, ai, - ah, ah, - ai, aj, - ak, al, - am, an, - ao, - ao, an, - am, al, - ak, aj, - ai, ah, - ah, ai, - aj, ak, - al, am, - an, ao }

40

{ cj, - cg, cd, - ca, bx, - bu, br, - bo, bl, - bi, bf, - bh, bk, - bn, bq, - bt, bw, - bz, cc, - cf, ci, ck, - ch, ce, - cb, by, - bv, bs, - bp, bm, - bj, bg, - bg, bj, - bm, bp, - bs, bv, - by, cb, - ce, ch, - ck, - ci, cf, - cc, bz, - bw, bt, - bq, bn, - bk, bh, - bf, bi, - bl, bo, - br, bu, - b x, ca, - cd, cg, - cj }

{ be, - bd, bc, - bb, ba, - az, ay, - ax, aw, - av, au, - at, as, - ar, a q, - ap, ap, - aq, ar, - as, at, - au, av, - aw, ax, - ay, az, - ba, bb, - bc, bd, - be, - be, bd, - bc, bb, - ba, az, - ay, ax, - aw, av, - au, at, - as, ar, - aq, ap, - ap, aq, - ar, as, - at, au, - av, aw, - ax, ay, - az,

50

ba、 - bb、 bc、 - bd、 be }

{ ck、 - cj、 ci、 - ch、 cg、 - cf、 ce、 - cd、 cc、 - cb、 ca、 - bz、 by、 - bx、 bw、 - bv、 bu、 - bt、 bs、 - br、 bq、 - bp、 bo、 - bn、 bm、 - bl、 bk、 - bj、 bi、 - bh、 bg、 - bf、 bf、 - bg、 bh、 - bi、 bj、 - bk、 bl、 - bm、 bn、 - bo、 bp、 - bq、 br、 - bs、 bt、 - bu、 bv、 - bw、 bx、 - by、 bz、 - ca、 cb、 - cc、 cd、 - ce、 cf、 - cg、 ch、 - ci、 cj、 - ck }

}

ここで

{ aa、 ab、 ac、 ad、 ae、 af、 ag、 ah、 ai、 aj、 ak、 al、 am、 an、 ao、 ap、 aq、 ar、 as、 at、 au、 av、 aw、 ax、 ay、 az、 ba、 bb、 bc、 bd、 be、 bf、 bg、 bh、 bi、 bj、 bk、 bl、 bm、 bn、 bo、 bp、 bq、 br、 bs、 bt、 bu、 bv、 bw、 bx、 by、 bz、 ca、 cb、 c c、 cd、 ce、 cf、 cg、 ch、 ci、 cj、 ck } =

10

{ 64、 83、 36、 89、 75、 50、 18、 90、 87、 80、 70、 57、 43、 25、 9、 90、 90、 88、 85、 82、 78、 73、 67、 61、 54、 46、 38、 31、 22、 13、 4、 91、 90、 90、 90、 88、 87、 86、 84、 83、 81、 79、 77、 73、 71、 69、 65、 62、 59、 56、 52、 4 8、 44、 41、 37、 33、 28、 24、 20、 15、 11、 7、 2 } (式1)

【 0 0 2 9 】

HEVCで採用されているDCT - 2および4 × 4 DST - 7に加えて、VVCでは適応型多重変換 (AMT、拡張型多重変換 (EMT) または多重変換選択 (MTS) と呼ばれることもある ) スキームが、コード間ブロックとコード内ブロックの両方の残差符号化に使用されている。これはHEVCの現在の変換以外に、DCT / DSTファミリーからの多重選択変換を使用する。新しく導入された変換マトリックスは、DST - 7、DCT - 8である。表6 (Table 6) は、選択したDST / DCTの基底関数を示す。

20

【 0 0 3 0 】

【表 5】

表 6 (Table 6) : N 点入力に対する DCT-2、DST-7、および DCT-8 の変換基底関数

変換タイプ	基底関数 $T_i(j)$ 、 $i, j=0, 1, \dots, N-1$
DCT-2	$T_i(j) = \omega_0 \cdot \sqrt{\frac{2}{N}} \cdot \cos\left(\frac{\pi \cdot i \cdot (2j + 1)}{2N}\right)$ <p>ここで <math>\omega_0 = \begin{cases} \sqrt{\frac{2}{N}} &amp; i = 0 \\ 1 &amp; i \neq 0 \end{cases}</math></p>
DCT-8	$T_i(j) = \sqrt{\frac{4}{2N + 1}} \cdot \cos\left(\frac{\pi \cdot (2i + 1) \cdot (2j + 1)}{4N + 2}\right)$
DST-7	$T_i(j) = \sqrt{\frac{4}{2N + 1}} \cdot \sin\left(\frac{\pi \cdot (2i + 1) \cdot (j + 1)}{2N + 1}\right)$

30

40

【 0 0 3 1 】

VVCのすべての主要な変換マトリックスは、8ビット表現で使用される。AMTは、幅と高さの両方が32以下のCUに適用され、AMTを適用するかどうかは、mts\_flagと呼ばれるフラグによって制御される。mts\_flagが0に等しい場合、残差のコーディングにはDCT

50

- 2のみが適用される。mts\_flagが1に等しい場合、インデックスmts\_idxは、表7 (Table 7)に従って使用される水平および垂直変換を指定するために2つのピンを使用してさらに通知される。ここで、値1はDST - 7を使用し、値2はDCT - 8を使用することを意味する。

【 0 0 3 2 】

【表 6】

表 7 (Table 7) : mts\_idx [x] [y] [cIdx] に応じた trTypeHor および trTypeVer の指定

mts_idx [xTbY] [yTbY] [cIdx]	trTypeHor	trTypeVer
-1	0	0
0	1	1
1	2	1
2	1	2
3	2	2

10

20

【 0 0 3 3 】

DST - 7の基底ベクトルで構成されるマトリックスである変換コアは、次のように表すこともできる。

表8 (Table 8) : 4ポイントのDST - 7 :

- { a, b, c, d }
- { c, c, 0, -c }
- { d, -a, -c, b }
- { b, -d, c, -a }

ここで

{ a, b, c, d } = { 29, 55, 74, 84 } (式2)

30

【 0 0 3 4 】

表9 (Table 9) : 8ポイントのDST - 7 :

- { a, b, c, d, e, f, g, h, }
- { c, f, h, e, b, -a, -d, -g, }
- { e, g, b, -c, -h, -d, a, f, }
- { g, c, -d, -f, a, h, b, -e, }
- { h, -a, -g, b, f, -c, -e, d, }
- { f, -e, -a, g, -d, -b, h, -c, }
- { d, -h, e, -a, -c, g, -f, b, }
- { b, -d, f, -h, g, -e, c, -a, }

ここで

{ a, b, c, d, e, f, g, h } = { 17, 32, 46, 60, 71, 78, 85, 86 } (式3)

40

【 0 0 3 5 】

表10 (Table 10) : 16ポイントのDST - 7 :

- { a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, }
- { c, f, i, l, o, o, l, i, f, c, 0, -c, -f, -i, -l, -o, }
- { e, j, o, m, h, c, -b, -g, -l, -p, -k, -f, -a, d, i, n, }
- { g, n, l, e, -b, -i, -p, -j, -c, d, k, o, h, a, -f, -m, }
- { i, o, f, -c, -l, -l, -c, f, o, i, 0, -i, -o, -f, c, l, }

50

{ k, k, 0, -k, -k, 0, k, k, 0, -k, -k, 0, k, k, 0, -k, }  
 { m, g, -f, -n, -a, l, h, -e, -o, -b, k, i, -d, -p, -c, j, }  
 { o, c, -l, -f, i, i, -f, -l, c, o, 0, -o, -c, l, f, -i, }  
 { p, -a, -o, b, n, -c, -m, d, l, -e, -k, f, j, -g, -i, h, }  
 { n, -e, -i, j, d, -o, a, m, -f, -h, k, c, -p, b, l, -g, }  
 { l, -i, -c, o, -f, -f, o, -c, -i, l, 0, -l, i, c, -o, f, }  
 { j, -m, c, g, -p, f, d, -n, i, a, -k, l, -b, -h, o, -e, }  
 { h, -p, i, -a, -g, o, -j, b, f, -n, k, -c, -e, m, -l, d, }  
 { f, -l, o, -i, c, c, -i, o, -l, f, 0, -f, l, -o, i, -c, }  
 { d, -h, l, -p, m, -i, e, -a, -c, g, -k, o, -n, j, -f, b, }  
 { b, -d, f, -h, j, -l, n, -p, o, -m, k, -i, g, -e, c, -a, }

10

ここで

{ a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p } = { 9, 17, 25, 33, 41, 49, 56, 62, 66, 72, 77, 81, 83, 87, 89, 90 } (式4)

【 0 0 3 6 】

表11 (Table 11) : 32ポイントのDST - 7 :

{ a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, A, B, C, D, E, F, }

{ c, f, i, l, o, r, u, x, A, D, F, C, z, w, t, q, n, k, h, e, b, -a, -d, -g, -j, -m, -p, -s, -v, -y, -B, -E, }

20

{ e, j, o, t, y, D, D, y, t, o, j, e, 0, -e, -j, -o, -t, -y, -D, -D, -y, -t, -o, -j, -e, 0, e, j, o, t, y, D, }

{ g, n, u, B, D, w, p, i, b, -e, -l, -s, -z, -F, -y, -r, -k, -d, c, j, q, x, E, A, t, m, f, -a, -h, -o, -v, -C, }

{ i, r, A, C, t, k, b, -g, -p, -y, -E, -v, -m, -d, e, n, w, F, x, o, f, -c, -l, -u, -D, -z, -q, -h, a, j, s, B, }

{ k, v, F, u, j, -a, -l, -w, -E, -t, -i, b, m, x, D, s, h, -c, -n, -y, -C, -r, -g, d, o, z, B, q, f, -e, -p, -A, }

{ m, z, z, m, 0, -m, -z, -z, -m, 0, m, z, z, m, 0, -m, -z, -z, -m, 0, m, z, }

30

{ o, D, t, e, -j, -y, -y, -j, e, t, D, o, 0, -o, -D, -t, -e, j, y, y, j, -e, -t, -D, -o, 0, o, D, t, e, -j, -y, }

{ q, E, n, -c, -t, -B, -k, f, w, y, h, -i, -z, -v, -e, l, C, s, b, -o, -F, -p, a, r, D, m, -d, -u, -A, -j, g, x, }

{ s, A, h, -k, -D, -p, c, v, x, e, -n, -F, -m, f, y, u, b, -q, -C, -j, i, B, r, -a, -t, -z, -g, l, E, o, -d, -w, }

{ u, w, b, -s, -y, -d, q, A, f, -o, -C, -h, m, E, j, -k, -F, -l, i, D, n, -g, -B, -p, e, z, r, -c, -x, -t, a, v, }

{ w, s, -d, -A, -o, h, E, k, -l, -D, -g, p, z, c, -t, -v, a, x, r, -e, -B, -n, i, F, j, -m, -C, -f, q, y, b, -u, }

40

{ y, o, -j, -D, -e, t, t, -e, -D, -j, o, y, 0, -y, -o, j, D, e, -t, -t, e, D, j, -o, -y, 0, y, o, -j, -D, -e, t, }

{ A, k, -p, -v, e, F, f, -u, -q, j, B, a, -z, -l, o, w, -d, -E, -g, t, r, -i, -C, -b, y, m, -n, -x, c, D, h, -s, }

{ C, g, -v, -n, o, u, -h, -B, a, D, f, -w, -m, p, t, -i, -A, b, E, e, -x, -l, q, s, -j, -z, c, F, d, -y, -k, r, }

{ E, c, -B, -f, y, i, -v, -l, s, o, -p, -r, m, u, -j, -x, g, A, -d, -D, a, F, b, -C, -e, z, h, -w, -k, t, n, -q, }

{ F, -a, -E, b, D, -c, -C, d, B, -e, -A, f, z, -g, -y, h, x, -i, -w, j, v, -k, -u, l, t, -m, -s, n, r, -o, -q, p, }

50

{ D, -e, -y, j, t, -o, -o, t, j, -y, -e, D, 0, -D, e, y, -j, -t, o, o, -t, -j, y, e, -D, 0, D, -e, -y, j, t, -o, }

{ B, -i, -s, r, j, -A, -a, C, -h, -t, q, k, -z, -b, D, -g, -u, p, l, -y, -c, E, -f, -v, o, m, -x, -d, F, -e, -w, n, }

{ z, -m, -m, z, 0, -z, m, m, -z, 0, z, -m, -m, z, 0, -z, m, m, -z, 0, z, -m, }

{ x, -q, -g, E, -j, -n, A, -c, -u, t, d, -B, m, k, -D, f, r, -w, -a, y, -p, -h, F, -i, -o, z, -b, -v, s, e, -C, l, }

{ v, -u, -a, w, -t, -b, x, -s, -c, y, -r, -d, z, -q, -e, A, -p, -f, B, -o, -g, C, -n, -h, D, -m, -i, E, -l, -j, F, -k, }

10

{ t, -y, e, o, -D, j, j, -D, o, e, -y, t, 0, -t, y, -e, -o, D, -j, -j, D, -o, -e, y, -t, 0, t, -y, e, o, -D, j, }

{ r, -C, k, g, -y, v, -d, -n, F, -o, -c, u, -z, h, j, -B, s, -a, -q, D, -l, -f, x, -w, e, m, -E, p, b, -t, A, -i, }

{ p, -F, q, -a, -o, E, -r, b, n, -D, s, -c, -m, C, -t, d, l, -B, u, -e, -k, A, -v, f, j, -z, w, -g, -i, y, -x, h, }

{ n, -B, w, -i, -e, s, -F, r, -d, -j, x, -A, m, a, -o, C, -v, h, f, -t, E, -q, c, k, -y, z, -l, -b, p, -D, u, -g, }

{ l, -x, C, -q, e, g, -s, E, -v, j, b, -n, z, -A, o, -c, -i, u, -F, t, -h, -d, p, -B, y, -m, a, k, -w, D, -r, f, }

20

{ j, -t, D, -y, o, -e, -e, o, -y, D, -t, j, 0, -j, t, -D, y, -o, e, e, -o, y, -D, t, -j, 0, j, -t, D, -y, o, -e, }

{ h, -p, x, -F, y, -q, i, -a, -g, o, -w, E, -z, r, -j, b, f, -n, v, -D, A, -s, k, -c, -e, m, -u, C, -B, t, -l, d, }

{ f, -l, r, -x, D, -C, w, -q, k, -e, -a, g, -m, s, -y, E, -B, v, -p, j, -d, -b, h, -n, t, -z, F, -A, u, -o, i, -c, }

{ d, -h, l, -p, t, -x, B, -F, C, -y, u, -q, m, -i, e, -a, -c, g, -k, o, -s, w, -A, E, -D, z, -v, r, -n, j, -f, b, }

{ b, -d, f, -h, j, -l, n, -p, r, -t, v, -x, z, -B, D, -F, E, -C, A, -y, w, -u, s, -q, o, -m, k, -i, g, -e, c, -a, }

30

ここで

{ a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, A, B, C, D, E, F } = { 4, 9, 13, 17, 21, 26, 30, 34, 38, 42, 45, 50, 53, 56, 60, 63, 66, 68, 72, 74, 77, 78, 80, 82, 84, 85, 86, 88, 88, 89, 90, 90 } (式5)

【0037】

表12 (Table 12) : 4ポイントのDCT - 8 :

{ a, b, c, d, }

{ b, 0, -b, -b, }

{ c, -b, -d, a, }

{ d, -b, a, -c, }

40

ここで

{ a, b, c, d } = { 84, 74, 55, 29 } (式6)

【0038】

表13 (Table 13) : 8ポイントのDCT - 8 :

{ a, b, c, d, e, f, g, h, }

{ b, e, h, -g, -d, -a, -c, -f, }

{ c, h, -e, -a, -f, g, b, d, }

{ d, -g, -a, -h, c, e, -f, -b, }

{ e, -d, -f, c, g, -b, -h, a, }

50

{ f, - a, g, e, - b, h, d, - c, }  
 { g, - c, b, - f, - h, d, - a, e, }  
 { h, - f, d, - b, a, - c, e, - g, }

ここで

{ a, b, c, d, e, f, g, h } = { 86, 85, 78, 71, 60, 46, 32, 17 } (式7)

【 0 0 3 9 】

表14 (Table 14) : 16ポイントのDCT - 8 :

{ a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, }  
 { b, e, h, k, n, 0, - n, - k, - h, - e, - b, - b, - e, - h, - k, - n, }  
 { c, h, m, - p, - k, - f, - a, - e, - j, - o, n, i, d, b, g, l, }  
 { d, k, - p, - i, - b, - f, - m, n, g, a, h, o, - l, - e, - c, - j, }  
 { e, n, - k, - b, - h, 0, h, b, k, - n, - e, - e, - n, k, b, h, }  
 { f, 0, - f, - f, 0, f, f, 0, - f, - f, 0, f, f, 0, - f, - f, }  
 { g, - n, - a, - m, h, f, - o, - b, - l, i, e, - p, - c, - k, j, d, }  
 { h, - k, - e, n, b, 0, - b, - n, e, k, - h, - h, k, e, - n, - b, }  
 { i, - h, - j, g, k, - f, - l, e, m, - d, - n, c, o, - b, - p, a, }  
 { j, - e, - o, a, - n, - f, i, k, - d, - p, b, - m, - g, h, l, - c, }  
 { k, - b, n, h, - e, 0, e, - h, - n, b, - k, - k, b, - n, - h, e, }  
 { l, - b, i, o, - e, f, - p, - h, c, - m, - k, a, - j, - n, d, - g, }  
 { m, - e, d, - l, - n, f, - c, k, o, - g, b, - j, - p, h, - a, i, }  
 { n, - h, b, - e, k, 0, - k, e, - b, h, - n, - n, h, - b, e, - k, }  
 { o, - k, g, - c, b, - f, j, - n, - p, l, - h, d, - a, e, - i, m, }  
 { p, - n, l, - j, h, - f, d, - b, a, - c, e, - g, i, - k, m, - o, }

10

20

ここで

{ a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p } = { 90, 89, 87, 83, 81, 77, 72, 66, 62, 56, 49, 41, 33, 25, 17, 9 } (式8)

【 0 0 4 0 】

表14 (Table 14) : 32ポイントのDCT - 8 :

{ a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, A, B, C, D, E, F, }  
 { b, e, h, k, n, q, t, w, z, C, F, - E, - B, - y, - v, - s, - p, - m, - j, - g, - d, - a, - c, - f, - i, - l, - o, - r, - u, - x, - A, - D, }  
 { c, h, m, r, w, B, 0, - B, - w, - r, - m, - h, - c, - c, - h, - m, - r, - w, - B, 0, B, w, r, m, h, c, c, h, m, r, w, B, }  
 { d, k, r, y, F, - A, - t, - m, - f, - b, - i, - p, - w, - D, C, v, o, h, a, g, n, u, B, - E, - x, - q, - j, - c, - e, - l, - s, - z, }  
 { e, n, w, F, - y, - p, - g, - c, - l, - u, - D, A, r, i, a, j, s, B, - C, - t, - k, - b, - h, - q, - z, E, v, m, d, f, o, x, }  
 { f, q, B, - A, - p, - e, - g, - r, - C, z, o, d, h, s, D, - y, - n, - c, - i, - t, - E, x, m, b, j, u, F, - w, - l, - a, - k, - v, }  
 { g, t, 0, - t, - g, - g, - t, 0, t, g, g, t, 0, - t, - g, - g, - t, 0, t, g, g, t, 0, - t, - g, - g, - t, 0, t, g, g, t, 0, - t, - g, - g, - t, 0, t, g, g, t, }  
 { h, w, - B, - m, - c, - r, 0, r, c, m, B, - w, - h, - h, - w, B, m, c, r, 0, - r, - c, - m, - B, w, h, h, w, - B, - m, - c, - r, }  
 { i, z, - w, - f, - l, - C, t, c, o, F, - q, - a, - r, E, n, d, u, - B, - k, - g, - x, y, h, j, A, - v, - e, - m, - D, s, b, p, }  
 { j, C, - r, - b, - u, z, g, m, F, - o, - e, - x, w, d, p, - E, - l, - h, - A, t, a, s, - B, - i, - k, - D, q, c, v, - y, - f, - n, }  
 { k, F, - m, - i, - D, o, g, B, - q, - e, - z, s, c, x, - u, - a, - v, w, b, t, - y, - d, - r, A, f, p, - C, - h, - n, E, j, l, }

30

40

50

{ l, -E, -h, -p, A, d, t, -w, -a, -x, s, e, B, -o, -i, -F, k, m, -D, -g, -q, z, c, u, -v, -b, -y, r, f, C, -n, -j, }

{ m, -B, -c, -w, r, h, 0, -h, -r, w, c, B, -m, -m, B, c, w, -r, -h, 0, h, r, -w, -c, -B, m, m, -B, -c, -w, r, h, }

{ n, -y, -c, -D, i, s, -t, -h, E, d, x, -o, -m, z, b, C, -j, -r, u, g, -F, -e, -w, p, l, -A, -a, -B, k, q, -v, -f, }

{ o, -v, -h, C, a, D, -g, -w, n, p, -u, -i, B, b, E, -f, -x, m, q, -t, -j, A, c, F, -e, -y, l, r, -s, -k, z, d, }

{ p, -s, -m, v, j, -y, -g, B, d, -E, -a, -F, c, C, -f, -z, i, w, -l, -t, o, q, -r, -n, u, k, -x, -h, A, e, -D, -b, }

10

{ q, -p, -r, o, s, -n, -t, m, u, -l, -v, k, w, -j, -x, i, y, -h, -z, g, A, -f, -B, e, C, -d, -D, c, E, -b, -F, a, }

{ r, -m, -w, h, B, -c, 0, c, -B, -h, w, m, -r, -r, m, w, -h, -B, c, 0, -c, B, h, -w, -m, r, r, -m, -w, h, B, -c, }

{ s, -j, -B, a, -C, -i, t, r, -k, -A, b, -D, -h, u, q, -l, -z, c, -E, -g, v, p, -m, -y, d, -F, -f, w, o, -n, -x, e, }

{ t, -g, 0, g, -t, -t, g, 0, -g, t, t, -g, 0, g, -t, -t, g, 0, -g, t, t, -g, 0, g, -t, -t, g, 0, -g, t, }

{ u, -d, B, n, -k, -E, g, -r, -x, a, -y, -q, h, -F, -j, o, A, -c, v, t, -e, C, m, -l, -D, f, -s, -w, b, -z, -p, i, }

20

{ v, -a, w, u, -b, x, t, -c, y, s, -d, z, r, -e, A, q, -f, B, p, -g, C, o, -h, D, n, -i, E, m, -j, F, l, -k, }

{ w, -c, r, B, -h, m, 0, -m, h, -B, -r, c, -w, -w, c, -r, -B, h, -m, 0, m, -h, B, r, -c, w, w, -c, r, B, -h, m, }

{ x, -f, m, -E, -q, b, -t, -B, j, -i, A, u, -c, p, F, -n, e, -w, -y, g, -l, D, r, -a, s, C, -k, h, -z, -v, d, -o, }

{ y, -i, h, -x, -z, j, -g, w, A, -k, f, -v, -B, l, -e, u, C, -m, d, -t, -D, n, -c, s, E, -o, b, -r, -F, p, -a, q, }

{ z, -l, c, -q, E, u, -g, h, -v, -D, p, -b, m, -A, -y, k, -d, r, -F, -t, f, -i, w, C, -o, a, -n, B, x, -j, e, -s, }

30

{ A, -o, c, -j, v, F, -t, h, -e, q, -C, -y, m, -a, l, -x, -D, r, -f, g, -s, E, w, -k, b, -n, z, B, -p, d, -i, u, }

{ B, -r, h, -c, m, -w, 0, w, -m, c, -h, r, -B, -B, r, -h, c, -m, w, 0, -w, m, -c, h, -r, B, B, -r, h, -c, m, -w, }

{ C, -u, m, -e, d, -l, t, -B, -D, v, -n, f, -c, k, -s, A, E, -w, o, -g, b, -j, r, -z, -F, x, -p, h, -a, i, -q, y, }

{ D, -x, r, -l, f, -a, g, -m, s, -y, E, C, -w, q, -k, e, -b, h, -n, t, -z, F, B, -v, p, -j, d, -c, i, -o, u, -A, }

{ E, -A, w, -s, o, -k, g, -c, b, -f, j, -n, r, -v, z, -D, -F, B, -x, t, -p, l, -h, d, -a, e, -i, m, -q, u, -y, C, }

40

{ F, -D, B, -z, x, -v, t, -r, p, -n, l, -j, h, -f, d, -b, a, -c, e, -g, i, -k, m, -o, q, -s, u, -w, y, -A, C, -E, }

ここで

{ a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z, A, B, C, D, E, F } = { 90, 90, 89, 88, 88, 86, 85, 84, 82, 80, 78, 77, 74, 72, 68, 66, 63, 60, 56, 53, 50, 45, 42, 38, 34, 30, 26, 21, 17, 13, 9, 4 } (式9)

【0041】

VVCでは、符号化ブロックの高さと幅の両方が64以下の場合、変換サイズは常に符号化ブロックサイズと同じになる。符号化ブロックの高さまたは幅のいずれかが64より大きい

50

場合、変換またはイントラ予測を行うとき、符号化ブロックは複数のサブブロックにさらに分割され、各サブブロックの幅と高さは64以下になり、各サブブロックに1つの変換が行われる。

【0042】

JVET - J0024、JVET - K0139、およびJVET - L0358では、空間的に変化する変換スキームが提案されている。SVTでは、インター予測の残差については、符号化ブロックには残差ブロックしかないが、残差ブロックは符号化ブロックよりも小さいため、SVTの変換サイズは符号化ブロックサイズよりも小さくなる。残差ブロックまたは変換でカバーされていない領域については、残差はゼロと仮定される。

【0043】

より具体的には、JVET - L0358では、SVTはサブブロック変換 (SBT) と呼ばれる。SBTでサポートされているサブブロックタイプ (SVT -、SVT - H)、サイズ、および位置 (左半分、左四半分、右半分、右四半分、上半分、上四半分、下半分、下四半分) を図3に示す。文字「A」でラベル付けされた影付きの領域は、変換された残差ブロックであり、残りの領域は変換されておらず残差がゼロであると仮定される。

【0044】

提案されたSBTをVVC上で使用することによる提案された仕様テキストの変更を以下に示す。SBT法では、サブブロックタイプ (水平または垂直)、サイズ (半分または四半分)、および位置 (左または右、上または下) を示すために、追加のオーバーヘッドビット (cu\_sbt\_flag、cu\_sbt\_quad\_flag、cu\_sbt\_horizontal\_flag、cu\_sbt\_pos\_flag) の通知が必要であることがわかる。

【0045】

【表7】

表 15 (Table 15) : シーケンス・パラメータ集合 RBSP 構文

seq_parameter_set_rbsp ( ) {	ディスク リプタ
sps_seq_parameter_set_id	ue (v)
...	
sps_mts_intra_enabled_flag	u (1)
sps_mts_inter_enabled_flag	u (1)
sps_sbt_enable_flag	u (1)
rbsp_trailing_bits ( )	
}	

【0046】

10

20

30

40

50

## 【表 8】

表 16 (Table 16) : 一般的なスライスヘッダ構文

slice_header ( ) {	ディスク リプタ	
slice_pic_parameter_set_id	ue (v)	
slice_address	u (v)	
slice_type	ue (v)	10
if (slice_type != I) {		
log2_diff_ctu_max_bt_size	ue (v)	
if (sps_sbtmvp_enabled_flag) {		
sbtmvp_size_override_flag	u (1)	
if (sbtmvp_size_override_flag)		
log2_sbtmvp_active_size_minus2	u (3)	
}		
if (sps_temporal_mvp_enabled_flag)		
slice_temporal_mvp_enabled_flag	u (1)	
if (slice_type == B)		
mvd_l1_zero_flag	u (1)	20
if (slice_temporal_mvp_enabled_flag) {		
if (slice_type == B)		
collocated_from_l0_flag	u (1)	
}		
six_minus_max_num_merge_cand	ue (v)	
if (sps_sbt_enable_flag)		
slice_max_sbt_size_64_flag	u (1)	
}		
if (sps_alf_enabled_flag) {		
slice_alf_enabled_flag	u (1)	30
if (slice_alf_enabled_flag)		
alf_data ( )		
}		
dep_quant_enabled_flag	u (1)	
if (!dep_quant_enabled_flag)		
sign_data_hiding_enabled_flag	u (1)	
byte_alignment ( )		
}		

10

20

30

40

【 0 0 4 7 】

50

## 【表 9】

表 17 (Table 17) : 符号化ユニット構文

coding_unit (x0、y0、cbWidth、cbHeight、treeType) {	ディスクリプタ	
...		
if (CuPredMode [x0] [y0] !=MODE_INTRA && cu_skip_flag [x0] [y0] ==0)		
cu_cbf	ae (v)	10
if (cu_cbf) {		
if (CuPredMode [x0] [y0] !=MODE_INTRA && sps_sbt_enable_flag)		
if (cbWidth ≤ maxSbtSize && cbHeight ≤ maxSbtSize) {		
allowSbtVerHalf=cbWidth ≥ 8		
allowSbtVerQuad=cbWidth ≥ 16		
allowSbtHorHalf=cbHeight ≥ 8		
allowSbtHorQuad=cbHeight ≥ 16		
if (allowSbtVerHalf    allowSbtHorHalf    allowSbtVerQuad    allowSbtHorQuad)		
cu_sbt_flag [x0] [y0]	ae (v)	20
}		
if (cu_sbt_flag [x0] [y0]) {		
if ( (allowSbtVerHalf    allowSbtHorHalf) && (allowSbtVerQuad    allowSbtHorQuad) )		
cu_sbt_quad_flag [x0] [y0]	ae (v)	
if ( (cu_sbt_quad_flag [x0] [y0] && allowSbtVerQuad && allowSbtHorQuad)    (!cu_sbt_quad_flag [x0] [y0] && allowSbtVerHalf && allowSbtHorHalf) )		
cu_sbt_horizontal_flag [x0] [y0]	ae (v)	
cu_sbt_pos_flag [x0] [y0]	ae (v)	30
}		
}		
transform_tree (x0、y0、cbWidth、cbHeight、treeType)		
}		
}		

## 【 0 0 4 8 】

10

20

30

40

50

## 【表 1 0】

表 18 (Table 18) : 変換ツリー構文

transform__tree (x0, y0, tbWidth, tbHeight, treeType) {	ディスクリプタ
if (tbWidth>MaxTbSizeY    tbHeight>MaxTbSizeY) {	
trafoWidth= (tbWidth>MaxTbSizeY) ? (tbWidth/2) : tbWidth	
trafoHeight= (tbHeight>MaxTbSizeY) ? (tbHeight/2) : tbHeight	
transform__tree (x0, y0, trafoWidth, trafoHeight)	
if (tbWidth>MaxTbSizeY)	10
transform__tree (x0+trafoWidth, y0, trafoWidth, trafoHeight, treeType)	
if (tbHeight>MaxTbSizeY)	
transform__tree (x0, y0+trafoHeight, trafoWidth, trafoHeight, treeType)	
if (tbWidth>MaxTbSizeY && tbHeight>MaxTbSizeY)	
transform__tree (x0+trafoWidth, y0+trafoHeight, trafoWidth, trafoHeight, treeType)	
} else if (cu__sbt__flag [x0] [y0] )	
factorTb0=cu__sbt__quad__flag [x0] [y0] ? 1 : 2	
factorTb0=cu__sbt__pos__flag [x0] [y0] ? (4-factorTb0) : factorTb0	
noResiTb0=cu__sbt__pos__flag [x0] [y0] ? 1 : 0	20
if ( ! cu__sbt__horizontal__flag [x0] [y0] ) {	
trafoWidth=tbWidth*factorTb0/4	
transform__tree (x0, y0, trafoWidth, tbHeight, treeType, noResiTb0)	
transform__tree (x0 + trafoWidth, y0, tbWidth - trafoWidth, tbHeight, treeType, ! noResiTb0)	
}	
else {	
trafoHeight=tbHeight*factorTb0/4	
transform__tree (x0, y0, tbWidth, trafoHeight, treeType, noResiTb0)	
transform__tree (x0, y0+trafoHeight, tbWidth, tbHeight - trafoHeight, treeType, ! noResiTb0)	30
}	
} else {	
transform__unit (x0, y0, tbWidth, tbHeight, treeType, 0)	
}	
}	

## 【 0 0 4 9】

40

50

【表 1 1】

表 19 (Table 19) : 変換ユニット構文

transform_unit (x0, y0, tbWidth, tbHeight, treeType, noResi) {	ディスクリプタ
if ( (treeType == SINGLE_TREE     treeType == DUAL_TREE_LUMA) && !noResi)	
tu_cbf_luma [x0] [y0]	ae (v)
if ( (treeType == SINGLE_TREE     treeType == DUAL_TREE_CHROMA) && !noResi) {	
tu_cbf_cb [x0] [y0]	ae (v)
tu_cbf_cr [x0] [y0]	ae (v)
}	
if ( ( ( (CuPredMode [x0] [y0] == MODE_INTRA) && sps_mts_intra_enabled_flag)     ( (CuPredMode [x0] [y0] == MODE_INTER) && sps_mts_inter_enabled_flag) ) && tu_cbf_luma [x0] [y0] && treeType != DUAL_TREE_CHROMA && (tbWidth ≤ 32) && (tbHeight ≤ 32) && !cu_sbt_flag [x0] [y0] )	
cu_mts_flag [x0] [y0]	ae (v)
if (tu_cbf_luma [x0] [y0] )	
residual_coding (x0, y0, log2 (tbWidth) , log2 (tbHeight) , 0)	
if (tu_cbf_cb [x0] [y0] )	
residual_coding (x0, y0, log2 (tbWidth/2) , log2 (tbHeight/2) , 1)	
if (tu_cbf_cr [x0] [y0] )	
residual_coding (x0, y0, log2 (tbWidth/2) , log2 (tbHeight/2) , 2)	
}	

## 【 0 0 5 0 】

シーケンス・パラメータ集合のRBSPセマンティクスの例によると、0に等しいsps\_sbt\_enabled\_flagは、インター予測されたCUのサブブロック変換が無効であることを指定する。1に等しいsps\_sbt\_enabled\_flagは、インター予測されたCUのサブブロック変換が有効になっていることを指定する。

## 【 0 0 5 1 】

一般的なスライスヘッダセマンティクスの例によると、0に等しいslice\_sbt\_max\_size\_64\_flagは、サブブロック変換を許可するための最大CU幅と高さを32と指定する。1に等しいslice\_sbt\_max\_size\_64\_flagは、サブブロック変換を許可するための最大CU幅と高さを64と指定する。例を以下に示す。

$$\text{maxSbtSize} = \text{slice\_sbt\_max\_size\_64\_flag} ? 64 : 32 \text{ (式10)}$$

## 【 0 0 5 2 】

コーディングユニットのセマンティクスの例によると、1に等しいcu\_sbt\_flag [x0] [y0] は、現在のコーディングユニットに対してサブブロック変換が使用されることを指定する。0に等しいcu\_sbt\_flag [x0] [y0] は、現在のコーディングユニットに対してサブブロック変換が使用されないことを指定する。cu\_sbt\_flag [x0] [y0] が存在しない場合、その値は0に等しいと推測される。サブブロック変換を使用する場合、コーディングユニットは2つの変換ユニットへとタイリング(tiling)され、一方の変換ユニットは残差を有し、もう一方は残差を有さない。

## 【 0 0 5 3 】

一実施形態によれば、1に等しいcu\_sbt\_quad\_flag [x0] [y0] は、現在のコーディングユニットについて、サブブロック変換が現在のコーディングユニットの1/4サイズ

の変換ユニットを含むことを指定する。0に等しい`cu_sbt_quad_flag[x0][y0]`は、現在のコーディングユニットについて、サブブロック変換が現在のコーディングユニットの1/2サイズの変換ユニットを含むことを指定する。`cu_sbt_quad_flag[x0][y0]`が存在しない場合、その値は0に等しいと推測される。

【0054】

一実施形態によれば、1に等しい`cu_sbt_horizontal_flag[x0][y0]`は、現在のコーディングユニットが水平分割によって2つの変換ユニットへとタイリングされることを指定する。0に等しい`cu_sbt_horizontal_flag[x0][y0]`は、現在のコーディングユニットが垂直分割によって2つの変換ユニットへとタイリングされることを指定する。

【0055】

一実施形態によれば、`cu_sbt_horizontal_flag[x0][y0]`が存在しない場合、その値は以下のように導出される。

- `cu_sbt_quad_flag[x0][y0]`が1に等しい場合、`cu_sbt_horizontal_flag[x0][y0]`は`allowSbtHoriQuad`に等しくなるように設定される。

- それ以外の場合 (`cu_sbt_quad_flag[x0][y0]`が0に等しい)、`cu_sbt_horizontal_flag[x0][y0]`は`allowSbtHoriHalf`に等しくなるように設定される。

【0056】

一実施形態によれば、1に等しい`cu_sbt_pos_flag[x0][y0]`は、現在のコーディングユニットの第1の変換ユニットの`tu_cbf_luma`、`tu_cbf_cb`および`tu_cbf_cr`がビットストリームに存在しないことを指定する。0に等しい`cu_sbt_pos_flag[x0][y0]`は、現在のコーディングユニットの第2の変換ユニットの`tu_cbf_luma`、`tu_cbf_cb`、および`tu_cbf_cr`がビットストリームに存在しないことを指定する。

【0057】

スケーリングされた変換係数の変換プロセスの例を以下に示す。このプロセスへの入力は次のとおりである。

- 現在のピクチャの左上の輝度サンプルに対する現在の輝度変換ブロックの左上のサンプルを指定する輝度位置 (`xTbY`、`yTbY`)、

- 現在の変換ブロックの幅を指定する変数`nTbW`、

- 現在の変換ブロックの高さを指定する変数`nTbH`、

- 現在のブロックの色成分を指定する変数`cldx`、

- スケーリングされた変換係数の  $(nTbW) \times (nTbH)$  配列 `d[x][y]` ( $x=0 \dots nTbW-1$ 、 $y=0 \dots nTbH-1$ )。

【0058】

このプロセスの出力は、残差サンプルの  $(nTbW) \times (nTbH)$  の配列 `r[x][y]` である ( $x=0 \dots nTbW-1$ 、 $y=0 \dots nTbH-1$ )。

【0059】

`cu_sbt_flag[xTbY][yTbY]`が1に等しい場合、水平変換カーネルを指定する変数`trTypeHor`と垂直変換カーネルを指定する変数`trTypeVer`は、`cu_sbt_horizontal_flag[xTbY][yTbY]`と`cu_sbt_pos_flag[xTbY][yTbY]`に応じて表8 Xで導出される。

【0060】

それ以外の場合 (`cu_sbt_flag[xTbY][yTbY]`は0に等しい)、水平変換カーネルを指定する変数`trTypeHor`と垂直変換カーネルを指定する変数`trTypeVer`は、`mts_idx[xTbY][yTbY]`と`CuPredMode[xTbY][yTbY]`に応じて表8 9で導出される。

【0061】

残差サンプルの  $(nTbW) \times (nTbH)$  配列 `r`は、次のように導出される。

1. スケーリングされた変換係数 `d[x][y]` の各 (垂直) 列 ( $x=0 \dots nTbW-1$ 、 $y=0 \dots nTbH-1$ ) は、Error! Reference source not found節で指定されている1次元変換プロセスを呼び出すことにより、`e[x][y]` ( $x=0 \dots nTbW-1$ 、 $y=0 \dots nTbH-1$ ) に変換される。ここで各列について $x=0 \dots nTbW-1$  (変換ブロック`nTbH`の高

10

20

30

40

50

さを持つ)であり、リストd[x][y]でy=0..nTbH-1であり、変換タイプ変数trTypeは入力としてtrTypeVerに等しく設定され、出力はリストe[x][y](y=0..nTbH-1)である。

2. 中間サンプル値g[x][y](x=0..nTbW-1, y=0..nTbH-1)は次のように導出される。

$$g[x][y] = \text{Clip3}(\text{CoeffMin}, \text{CoeffMax}, (e[x][y] + 256) >> 9) (0 - 1)$$

3. 結果の配列g[x][y](x=0..nTbW-1, y=0..nTbH-1)の各(水平)行は、Error! Reference source not found節で指定されている1次元変換プロセスを呼び出すことにより、r[x][y](x=0..nTbW-1, y=0..nTbH-1)に変換される。ここで各行についてy=0..nTbH-1(変換ブロックnTbWの幅を持つ)であり、リストg[x][y]でx=0..nTbW-1であり、変換タイプ変数trTypeは入力としてtrTypeHorに等しく設定され、出力はリストr[x][y](x=0..nTbW-1)である。

【0062】

【表12】

表 20 (Table 10): mts\_idx[x][y] および CuPredMode[x][y] に応じた trTypeHor および trTypeVer の指定

	cu_sbt_horizontal_flag [xTbY][yTbY] == 0		cu_sbt_horizontal_flag [xTbY][yTbY] == 1	
	cu_sbt_pos_flag [xTbY][yTbY] == 0	cu_sbt_pos_flag [xTbY][yTbY] == 1	cu_sbt_pos_flag [xTbY][yTbY] == 0	cu_sbt_pos_flag [xTbY][yTbY] == 1
trTypeHor	2	1	nTbW > 32 ? 0 : 1	nTbW > 32 ? 0 : 1
trTypeVer	nTbH > 32 ? 0 : 1	nTbH > 32 ? 0 : 1	2	1

【0063】

【表13】

表 21 (Table 21): mts\_idx[x][y] および CuPredMode[x][y] に応じた trTypeHor および trTypeVer の指定

mts_idx [xTbY][yTbY]	CuPredMode [xTbY][yTbY] == MODE_INTRA		CuPredMode [xTbY][yTbY] == MODE_INTER	
	trTypeHor	trTypeVer	trTypeHor	trTypeVer
-1 (推定)	0	0	0	0
0 (00)	1	1	2	2
1 (01)	2	1	1	2
2 (10)	1	2	2	1
3 (11)	2	2	1	1

【0064】

図10は、本開示の実施形態による通信システム(1000)の簡略化されたブロック図を示す。通信システム(1000)は、ネットワーク(1050)を介して相互接続された少なく

とも2つの端末(1010~1020)を含み得る。データの一方方向送信の場合、第1の端末(1010)は、ネットワーク(1050)を介して他の端末(1020)に送信するために、ローカル位置でビデオデータをコード化することができる。第2の端末(1020)は、ネットワーク(1050)から他の端末のコード化されたビデオデータを受信し、コード化されたデータを復号し、復号されたビデオデータを表示することができる。一方方向のデータ送信は、メディア・サービング・アプリケーションなどでは一般的であってもよい。

【0065】

図10は、たとえば、ビデオ会議中に発生しうるコード化されたビデオの双方向送信をサポートするために提供される第2の対の端末(1030、1040)を示している。データの双方向送信の場合、各端末(1030、1040)は、ネットワーク(1050)を介して他の端末に送信するために、ローカル位置でキャプチャされたビデオデータをコード化することができる。各端末(1030、1040)はまた、他の端末によって送信されたコード化されたビデオデータを受信し、コード化されたデータを復号し、復号されたビデオデータをローカル表示装置に表示することができる。

10

【0066】

図10では、端末(1010~1040)は、サーバ、パーソナルコンピュータ、およびスマートフォンとして示され得るが、本開示の原理は、そのように限定されない。本開示の実施形態は、ラップトップコンピュータ、タブレットコンピュータ、メディアプレーヤ、および/または専用のビデオ会議機器での適用を見出す。ネットワーク(1050)は、たとえばワイヤラインおよび/または無線通信ネットワークを含む、端末(1010~1040)間で、コード化されたビデオデータを伝達する任意の数のネットワークを表す。通信ネットワーク(1050)は、回路交換および/またはパケット交換チャンネルでデータを交換することができる。代表的なネットワークには、通信ネットワーク、ローカル・エリア・ネットワーク、ワイド・エリア・ネットワーク、インターネットなどがある。本議論の目的のためには、ネットワーク(1050)のアーキテクチャおよびトポロジは、以下に本明細書で説明されない限り、本開示の動作にとって重要ではない場合がある。

20

【0067】

図11は、開示された主題の適用の例として、ストリーミング環境、たとえばストリーミングシステム300におけるビデオエンコーダおよびビデオデコーダの配置を示している。開示された主題は、たとえば、ビデオ会議、デジタルTV、そしてCD、DVD、メモリスティックなどを含むデジタル媒体への圧縮ビデオの格納などを含む他のビデオ対応アプリケーションに等しく適用可能であり得る。

30

【0068】

ストリーミングシステムは、たとえば非圧縮ビデオサンプルストリーム(1102)を作成する、たとえばデジタルカメラなどのビデオソース(1101)を含むことができるキャプチャサブシステム(1113)を含みうる。そのサンプルストリーム(1102)は、符号化されたビデオビットストリームと比較したときに大量のデータを強調するために太線で示され、カメラ(1101)に結合されたエンコーダ(1103)によって処理することができる。エンコーダ(1103)は、以下により詳細に説明されるような、開示された主題の態様を可能にするまたは実装するためのハードウェア、ソフトウェア、またはそれらの組み合わせを含むことができる。符号化されたビデオビットストリーム(1104)は、サンプルストリームと比較してデータ量が少ないことを強調するために細い線で示され、将来の使用のためにストリーミングサーバ(1105)に格納されうる。1つまたは複数のストリーミングクライアント(1106、1108)は、ストリーミングサーバ(1105)にアクセスして、符号化されたビデオビットストリーム(1104)のコピー(1107、1109)を取得することができる。クライアント(1106)は、符号化されたビデオビットストリーム(1107)の入ってくるコピーを復号し、ディスプレイ(1112)または他のレンダリング装置(図示せず)上でレンダリングされうる出ていくビデオサンプルストリーム(1111)を作成するビデオデコーダ(1110)を含むことができる。いくつかのストリーミングシステムでは、ビデオビットストリーム(1104、1107、1109)は、特定のビデオ符号化/圧

40

50

縮規格に従って符号化されうる。これらの標準の例には、ITU - T勧告H. 265が含まれる。一例では、開発中のビデオ符号化標準は、非公式に多用途映像符号化方式（VVC）として知られている。開示する主題は、VVCの文脈で使用され得る。

【0069】

図12は、本発明の一実施形態によるビデオデコーダ（1110）の機能ブロック図であり得る。

【0070】

受信器（1210）は、デコーダ（1110）によって復号される1つまたは複数のコーデックビデオシーケンスを受信することができ、同じまたは他の実施形態では、一度に1つの符号化されたビデオシーケンスであり、各符号化されたビデオシーケンスの復号は、他の符号化されたビデオシーケンスから独立している。符号化されたビデオシーケンスは、符号化されたビデオデータを格納する記憶装置へのハードウェア/ソフトウェアリンクであり得るチャンネル（1212）から受信されうる。受信器（1210）は、符号化されたビデオデータを他のデータ、たとえば、符号化されたオーディオデータおよび/または補助データストリームとともに受信することができ、それらはそれぞれの使用エンティティ（図示せず）に転送され得る。受信器（1210）は、符号化されたビデオシーケンスを他のデータから分離することができる。ネットワークジッタに対抗するために、バッファメモリ（1215）は、受信器（1210）とエントロピーデコーダ/パーサ（1220）（以下、「パーサ」）との間に結合され得る。受信器（1210）が十分な帯域幅および制御可能性を有する格納/転送装置から、または同期ネットワークからデータを受信しているとき、バッファ（1215）は必要とされないか、または小さいものであってもよい。インターネットなどのベストエフォートパケットネットワークで使用するためには、バッファ（1215）が必要になる場合があり、比較的大きくすることができ、有利に適応サイズにすることができる。

【0071】

ビデオデコーダ（1110）は、エントロピー符号化されたビデオシーケンスからシンボル（1221）を再構築するためのパーサ（1220）を含み得る。これらのシンボルのカテゴリには、デコーダ（1110）の動作を管理するために使用される情報、および場合によっては、ディスプレイ（1112）などのレンダリング装置を制御するための情報が含まれ、レンダリング装置は、図12に示されているように、デコーダの一体部分ではないが、デコーダに結合されうる。レンダリング装置の制御情報は、補足拡張情報（SEIメッセージ）またはビデオユーザビリティ情報（VUI）パラメータ集合フラグメント（図示せず）の形であり得る。パーサ（1220）は、受信された符号化されたビデオシーケンスを解析/エントロピー復号しうる。符号化されたビデオシーケンスの符号化は、ビデオ符号化技術または標準に従うことができ、可変長符号化、ハフマン符号化、文脈依存性のある、またはない算術符号化などを含む当業者によく知られている原理に従いうる。パーサ（1220）は、符号化されたビデオシーケンスから、グループに対応する少なくとも1つのパラメータに基づいて、ビデオデコーダにおけるピクセルのサブグループのうち少なくとも1つのサブグループパラメータの集合を抽出することができる。サブグループには、Group of Pictures（GOP）、ピクチャ、タイル、スライス、マクロブロック、コーディングユニット（CU）、ブロック、変換ユニット（TU）、予測ユニット（PU）などを含めることができる。エントロピーデコーダ/パーサはまた、符号化されたビデオシーケンスから、変換係数、量子化パラメータ（QP）値、動きベクトルなどのような情報を抽出することができる。

【0072】

パーサ（1220）は、シンボル（1221）を作成するために、バッファ（1215）から受け取られたビデオシーケンスに対してエントロピー復号/解析動作を行うことができる。パーサ（1220）は、符号化されたデータを受信し、特定のシンボル（1221）を選択的に復号することができる。さらに、パーサ（1220）は、特定のシンボル（1221）を動き補償予測ユニット（1253）、スケーラ/逆変換ユニット（1251）、イントラ予測ユニット

10

20

30

40

50

(1252)、またはループ・フィルタ(1256)に送るべきか決定することができる。

【0073】

シンボル(1221)の再構築には、符号化されたビデオ画像またはその一部(インターおよびイントラ画像、インターおよびイントラブロックなど)のタイプ、およびその他の要因に応じて、複数の異なるユニットが含まれ得る。含まれるユニットおよびその方法は、パーサ(1220)によって符号化されたビデオシーケンスから解析されたサブグループ制御情報によって制御されうる。パーサ(1220)と以下の複数のユニットとの間のそのようなサブグループ制御情報の流れは、明確性のため、図示しない。

【0074】

既に述べた機能ブロックのほかに、デコーダ(1110)は、以下に説明するように、概念的にいくつかの機能ユニットに細分することができる。商業的な制約の下で動作する実際の実装では、これらのユニットの多くは互いに密接に相互作用し、少なくとも部分的には互いに統合されうる。しかしながら、開示された主題を説明するために、以下の機能ユニットへの概念的な細分が適切である。

【0075】

最初のユニットは、スケーラ/逆変換ユニット(1251)である。スケーラ/逆変換ユニット(1251)は、量子化された変換係数、ならびに、使用する変換、ブロックサイズ、量子化係数、量子化スケーリングマトリックスなどを含む制御情報を、パーサ(1220)からシンボル(1221)として受信する。これはアグリゲータ(1255)に入力できるサンプル値を含むブロックを出力できる。

【0076】

場合によっては、スケーラ/逆変換ユニット(1251)の出力サンプルは、イントラ符号化されたブロックに関係することがあり、すなわち、それは以前に再構築されたピクチャからの予測情報を使用していないが、現在のピクチャの以前に再構築された部分からの予測情報を使用できるブロックである。そのような予測情報は、イントラピクチャ予測ユニット(1252)によって提供することができる。場合によっては、イントラピクチャ予測ユニット(1252)は、現在の(部分的に再構築された)ピクチャ(1258)からフェッチされた周囲の既に再構築された情報を使用して、再構築中のブロックと同じサイズおよび形状のブロックを生成する。アグリゲータ(1255)は、場合によっては、サンプルごとに、イントラ予測ユニット(1252)が生成した予測情報を、スケーラ/逆変換ユニット(1251)によって提供された出力サンプル情報に追加する。

【0077】

他の場合では、スケーラ/逆変換ユニット(1251)の出力サンプルは、インター符号化された、場合によっては動き補償されたブロックに関係しうる。そのような場合、動き補償予測ユニット(1253)は、参照ピクチャメモリ(1257)にアクセスして、予測に使用されるサンプルをフェッチすることができる。フェッチされたサンプルをブロックに関連するシンボル(1221)に従って動き補償した後、これらのサンプルは、出力サンプル情報を生成するために、アグリゲータ(1255)によってスケーラ/逆変換ユニットの出力に追加されうる(この場合、残差サンプルまたは残差信号と呼ばれる)。動き補償ユニットが予測サンプルをフェッチする参照ピクチャメモリフォーム内のアドレスは、動きベクトルによって制御することができ、動き補償ユニットは、たとえば、X、Y、および参照ピクチャ構成要素を有し得るシンボル(1221)の形で利用できる。動き補償はまた、サブサンプルの正確な動きベクトルが使用されているときに参照ピクチャメモリからフェッチされたサンプル値の補間や、動きベクトル予測メカニズムなどを含むことができる。

【0078】

アグリゲータ(1255)の出力サンプルは、ループ・フィルタ・ユニット(1256)における様々なループフィルタリングを受けうる。ビデオ圧縮技術は、符号化されたビデオビットストリームに含まれるパラメータによって制御され、パーサ(1220)からのシンボル(1221)としてループ・フィルタ・ユニット(1256)で利用できるループ内フィルタ技術を含みうるが、符号化されたピクチャまたは符号化されたビデオシーケンスの以前の

10

20

30

40

50

(復号順の)部分の復号中に得られたメタ情報に応答することも、以前に再構築されループフィルタリングされたサンプル値に応答することもできる。

【0079】

ループ・フィルタ・ユニット(1256)の出力は、レンダリング装置(1112)に出力することができ、かつ将来のインターピクチャ予測で使用するために参照ピクチャメモリ(1258)に格納できるサンプルストリームとすることができる。

【0080】

ある符号化されたピクチャは、完全に再構築されると、将来の予測のための参照ピクチャとして使用されうる。符号化されたピクチャが完全に再構築され、符号化されたピクチャが(たとえば、パーサ(1220)によって)参照ピクチャとして識別されると、現在の参照ピクチャ(1258)は参照ピクチャバッファ(1257)の一部になることができ、次の符号化されたピクチャの再構築を開始する前に、新しい現在のピクチャメモリが再割り当てされることができる。

10

【0081】

ビデオデコーダ(1110)は、ITU-T勧告H.265などの標準に文書化され得る所定のビデオ圧縮技術に従って復号動作を行うことができる。符号化されたビデオシーケンスは、ビデオ圧縮技術文書または標準、特にその中のプロファイル文書で指定されているように、ビデオ圧縮技術または標準の構文に準拠しているという意味で、使用されているビデオ圧縮技術または標準によって指定された構文に準拠していてもよい。また、符号化されたビデオシーケンスの複雑さが、ビデオ圧縮技術または標準のレベルで定義されているレベル内にあることも、コンプライアンスに必要である。場合によっては、レベルによって、最大ピクチャサイズ、最大フレームレート、最大再構築サンプルレート(たとえば、メガサンプル/秒で測定)、最大参照ピクチャサイズなどが制限される。レベルによって設定された制限は、場合によっては、仮想参照デコーダ(HRD)の仕様と、符号化されたビデオシーケンスで通知されるHRDバッファ管理のメタデータによってさらに制限されることがある。

20

【0082】

一実施形態では、受信器(1210)は、符号化されたビデオとともに追加の(冗長な)データを受信してもよい。追加のデータは、符号化されたビデオシーケンスの一部として含まれてもよい。追加のデータは、データを適切に復号し、および/または元のビデオデータをより正確に再構築するためにビデオデコーダ(1110)によって使用されてもよい。追加のデータは、たとえば、時間的、空間的、または信号対ノイズ比(SNR)エンハンスメントレイヤ、冗長スライス、冗長ピクチャ、前方誤り訂正コードなどの形式にすることができる。

30

【0083】

図13は、本開示の一実施形態によるビデオエンコーダ(1103)の機能ブロック図であってもよい。

【0084】

エンコーダ(1103)は、エンコーダ(1103)によって符号化されるビデオ画像をキャプチャすることができるビデオソース(1101)(エンコーダの一部ではない)からビデオサンプルを受信することができる。

40

【0085】

ビデオソース(1101)は、エンコーダ(1103)によって符号化されるソース・ビデオシーケンスを、任意の適切なビット深度(たとえば、8ビット、10ビット、12ビット、...)、任意の色空間(たとえば、BT.601 Y CrCb、RGB、...)および任意の適切なサンプリング構造(たとえば、Y CrCb 4:2:0、Y CrCb 4:4:4)であり得るデジタル・ビデオ・サンプル・ストリームの形態で提供することができる。メディアサービングシステムでは、ビデオソース(1101)は、以前に準備されたビデオを格納するストレージ装置であり得る。ビデオ会議システムでは、ビデオソース(1103)は、ローカル画像情報をビデオシーケンスとしてキャプチャするカメラであり得る。ビデオデータは、順

50

番に見たときに動きを与える複数の個別のピクチャとして提供されてもよい。ピクチャ自体は、ピクセルの空間配列として編成することができ、各ピクセルは、使用中のサンプリング構造、色空間などに応じて、1つまたは複数のサンプルを含むことができる。当業者は、ピクセルとサンプルとの関係を容易に理解することができる。以下の説明では、サンプルに重点を置いている。

**【0086】**

一実施形態によれば、エンコーダ(1103)は、リアルタイムまたはアプリケーションによって要求される任意の他の時間制約の下で、ソース・ビデオシーケンスのピクチャを符号化し、符号化されたビデオシーケンス(1343)に圧縮し得る。適切な符号化速度を実施することは、コントローラ(1350)の1つの機能である。コントローラは、以下に説明するように他の機能ユニットを制御し、これらのユニットに機能的に結合される。分かりやすくするために、結合は図示していない。コントローラによって設定されるパラメータには、レート制御関連のパラメータ(ピクチャスキップ、量子化、レート歪み最適化手法のラムダ値など)、ピクチャサイズ、Group Of Pictures (GOP)レイアウト、最大動きベクトル探索範囲、などが含まれ得る。当業者は、特定のシステム設計用に最適化されたビデオエンコーダ(1103)に関係し得るので、コントローラ(1350)の他の機能を容易に識別することができる。

10

**【0087】**

いくつかのビデオエンコーダは、当業者が「コーディング・ループ」として容易に認識するもので動作する。非常に単純化された説明として、コーディング・ループは、エンコーダ(1330)(以下、「ソースコード」)の符号化部分(符号化される入力ピクチャと、参照ピクチャとに基づいてシンボルを作成することを担当)と、シンボルを再構築して、(シンボルとコード化されたビデオビットストリームとの間の圧縮は、開示される主題で検討されるビデオ圧縮技術では可逆であるため)(リモート)デコーダも作成するサンプルデータを作成する、エンコーダ(1103)に埋め込まれた(ローカル)デコーダ(1333)から構成され得る。その再構築されたサンプルストリームは、参照ピクチャメモリ(1334)に入力される。シンボルストリームの復号は、デコーダの場所(ローカルまたはリモート)に関係なくビットパーフェクトな結果をもたらすため、参照ピクチャバッファコンテンツのコンテンツもローカルエンコーダとリモートエンコーダ間でビットパーフェクトである。言い換えると、エンコーダの予測部分は、参照ピクチャのサンプルを、デコーダが復号中に予測を使用するときに「見る」のとまったく同じサンプル値として「見る」。参照ピクチャの同期性のこの基本原理(および、たとえばチャネルエラーのために同期性を維持できない場合に生じるドリフト)は、当業者によく知られている。

20

30

**【0088】**

「ローカル」デコーダ(1333)の動作は、図12に関連して既に詳細に説明されている、「リモート」デコーダ(1110)の動作と同じであり得る。しかしながら、図13も簡単に参照すると、シンボルが利用可能であり、エン트로ピーコード(1345)およびパーサ(1220)による符号化されたビデオシーケンスへのシンボルの符号化/復号は可逆であり得るため、デコーダ(1110)のエン트로ピー復号部分、ならびにチャネル(1212)、受信器(1210)、バッファ(1215)およびパーサ(1220)は、ローカルデコーダ(1333)において完全に実装されない場合がある。

40

**【0089】**

この時点で注目できることは、デコーダに存在する解析/エン트로ピー復号以外のいかなるデコーダ技術も、対応するエンコーダに実質的に同一の機能形式で必ず存在する必要があることである。エンコーダ技術の説明は、包括的に説明されたデコーダ技術の逆であるため、省略されうる。いくつかの領域においてのみ、より詳細な説明が必要であり、以下に与えられる。

**【0090】**

その動作の一部として、ソースコード(1330)は、「参照フレーム」として指定されたビデオシーケンスからの1つまたは複数の以前に符号化されたフレームを参照して入力

50

フレームを予測的に符号化する動き補償予測符号化を行いうる。このようにして、コーディングエンジン(1332)は、入力フレームのピクセルブロックと、入力フレームに対する予測参照として選択され得る参照フレームのピクセルブロックとの間の差異をコーディングする。

【0091】

ローカル・ビデオ・デコーダ(1333)は、ソースコーダ(1330)によって作成されたシンボルに基づいて、参照フレームとして指定され得るフレームの符号化されたビデオデータを復号し得る。符号化エンジン(1332)の動作は、非可逆プロセスであることが有利であり得る。符号化されたビデオデータがビデオデコーダ(図13には図示せず)で復号され得るとき、再構築されたビデオシーケンスは、通常、いくつかのエラーを伴うソース・ビデオシーケンスのレプリカであり得る。ローカル・ビデオ・デコーダ(1333)は、参照フレームに対してビデオデコーダによって実行され得る復号プロセスを複製し、再構築された参照フレームを参照ピクチャキャッシュ(1334)に格納させ得る。このようにして、エンコーダ(1103)は、遠端ビデオデコーダによって取得される(送信エラーがない)再構築参照フレームとして、共通のコンテンツを有する再構築参照フレームのコピーをローカルに格納し得る。

10

【0092】

予測器(1335)は、符号化エンジン(1332)の予測検索を実行することができる。すなわち、符号化される新しいフレームについて、予測器(1335)は、(候補参照ピクセルブロックとして)サンプルデータまたは参照ピクチャの動きベクトル、ブロック形状などの特定のメタデータについて参照ピクチャメモリ(1334)を検索することができ、これらは新しいピクチャの適切な予測参照として機能することができる。予測器(1335)は、適切な予測参照を見つけるために、サンプルブロックのピクセルブロック(sample-by-pixel block)ごとに動作することができる。場合によっては、予測器(1335)によって得られた検索結果によって決定されるように、入力画像は、参照ピクチャメモリ(1334)に格納された複数の参照ピクチャから引き出された予測参照を有し得る。

20

【0093】

コントローラ(1350)は、たとえば、ビデオデータを符号化するために使用されるパラメータおよびサブグループパラメータの設定を含む、ビデオコーダ(1330)の符号化動作を管理することができる。

30

【0094】

前述のすべての機能ユニットの出力は、エントロピーコーダ(1345)でエントロピー符号化を受けることができる。エントロピーコーダは、たとえばハフマン符号化、可変長符号化、算術符号化などの当業者に知られている技術に従ってシンボルを可逆圧縮することにより、様々な機能ユニットによって生成されたシンボルを符号化されたビデオシーケンスに変換する。

【0095】

送信器(1340)は、エントロピーコーダ(1345)によって作成された符号化されたビデオシーケンスをバッファリングして、符号化されたビデオデータを格納するストレージ装置へのハードウェア/ソフトウェアリンクであり得る通信チャネル(1360)を介した送信に備えることができる。送信器(1340)は、ビデオコーダ(1330)からの符号化されたビデオデータを、送信される他のデータ、たとえば符号化オーディオデータおよび/または補助データストリーム(ソースは図示せず)とマージすることができる。

40

【0096】

コントローラ(1350)は、エンコーダ(1103)の動作を管理することができる。符号化中、コントローラ(1350)は、各符号化されたピクチャに、或る符号化されたピクチャタイプを割り当てることができる。これは、それぞれのピクチャに適用され得る符号化手法に影響を及ぼし得る。たとえば、多くの場合、ピクチャは以下のフレームタイプのいずれかを割り当てられ得る。

【0097】

50

イントラピクチャ (Iピクチャ) は、シーケンス内の他のフレームを予測のソースとして使用せずに符号化および復号できるものであり得る。一部のビデオコーデックでは、たとえば Independent Decoder Refreshピクチャなど、様々なタイプのイントラピクチャを使用できる。当業者は、Iピクチャのそれらの変形およびそれらのそれぞれの用途および特徴を知っている。

【0098】

予測ピクチャ (Pピクチャ) は、各ブロックのサンプル値を予測するために最大で1つの動きベクトルおよび参照インデックスを使用するイントラ予測またはインター予測を使用して符号化および復号され得るものであり得る。

【0099】

双方向予測ピクチャ (Bピクチャ) は、各ブロックのサンプル値を予測するために最大で2つの動きベクトルおよび参照インデックスを使用するイントラ予測またはインター予測を使用して符号化および復号され得るものであり得る。同様に、複数の予測ピクチャは、単一のブロックの再構築に3つ以上の参照ピクチャおよび関連付けられたメタデータを使用できる。

【0100】

ソースピクチャは、通常、空間的に複数のサンプルブロック (たとえば、それぞれ  $4 \times 4$ 、 $8 \times 8$ 、 $4 \times 8$ 、または  $16 \times 16$  サンプルのブロック) に分割され、ブロックごとに符号化される。ブロックは、ブロックのそれぞれのピクチャに適用される符号化割り当てによって決定された、他の (既に符号化された) ブロックを参照して予測的に符号化され得る。たとえば、Iピクチャのブロックは非予測的に符号化されてもよいし、または同じピクチャの既に符号化されたブロック (空間予測またはイントラ予測) を参照して予測的に符号化されてもよい。Pピクチャのピクセルブロックは、空間的予測を介して、または1つの以前に符号化された参照ピクチャを参照して時間的予測を介して、非予測的に符号化され得る。Bピクチャのブロックは、空間的予測を介して、または1つまたは2つの以前に符号化された参照ピクチャを参照して時間的予測を介して、非予測的に符号化され得る。

【0101】

ビデオコーダ (1103) は、所定のビデオコーディング技術またはITU-T勧告H.265などの標準に従ってコーディング動作を行うことができる。その動作において、ビデオコーダ (1103) は、入力ビデオシーケンスにおける時間的および空間的冗長性を活用する予測符号化動作を含む、様々な圧縮動作を実行することができる。したがって、コーディングされたビデオデータは、ビデオコーディング技術または使用されている標準で指定された構文に準拠することができる。

【0102】

一実施形態では、送信器 (1340) は、符号化されたビデオとともに追加のデータを送信することができる。ビデオコーダ (1330) は、そのようなデータを、符号化されたビデオシーケンスの一部として含み得る。追加のデータには、時間的 / 空間的 / SNRエンハンスメントレイヤ、冗長なピクチャやスライスなどの冗長データの他の形式、付加拡張情報 (SEI) メッセージ、ビジュアルユーザビリティ情報 (VUI) パラメータ集合フラグメントなどが含まれる。

【0103】

VVCでは、幅または高さが64より大きいCUを除いて、CUサイズは常にTUサイズと同じである。これは、HEVCのRQT設計よりも柔軟性が低く、よりよいコーディングパフォーマンスのために改善されうる。

【0104】

再帰的な変換分割は高い複雑さをもたらす、VVCには四分木分割を好まない非正方形の符号化ブロックがあるため、VVC上でRQTを直接利用することは最適ではない可能性がある。

【0105】

SVTは、サブブロック変換のタイプ、位置、およびサイズを示すためにいくつかのオー

10

20

30

40

50

バーヘッドビットを必要とする。これは、特にインター符号化にはコストがかかる。このような通知を回避でき、サブブロック変換のタイプ、位置、およびサイズを暗黙的に導出するために使用される方法があれば、符号化ゲインとエンコーダの複雑さの点でより効率的でありうる。

**【0106】**

SVTの制限の1つは、非常に限られたサブブロック位置（左または右、上または下など）しかサポートできない一方で、サポート位置を追加すると、必然的にエンコーダの複雑さと通知コストが増加することである。サブブロックの位置を暗黙的に導出できるならば、サポートされるサブブロックの位置は、エンコーダまたはデコーダの複雑さを増加することなく、より柔軟になりうる。

10

**【0107】**

本開示の実施形態は、別個に使用することも、任意の順序で組み合わせて使用することもできる。本開示を通じて、DST - 7はDST - 4に置き換えることができ、DCT - 8はDCT - 4に置き換えることができる。

**【0108】**

一実施形態によれば、ブロック形状適応型変換分割が適用されうる。この実施形態では、サブブロック変換の数およびサブブロック変換のサイズは、符号化ブロック（CU）のサイズに依存する。

**【0109】**

一実施形態では、符号化ブロックが正方形ブロックである場合、符号化ブロックは、図4Aに示すような1回の単一の変換、または図4Bに示すような4回の4分の1サイズの変換のいずれかを使用することができる。選択はフラグによって通知され得る。

20

**【0110】**

他の実施形態では、符号化ブロックの幅が高さよりも大きい場合、符号化ブロックは、図5Aに示すような単一の変換、または複数の等しいより小さなサイズの変換のいずれかを使用することができる。選択はフラグによって通知され得る。

**【0111】**

一例では、符号化ブロックは、図5Bに示すように2つの半分サイズのサブブロックに分割され、変換は各サブブロックに行われ、半分サイズ変換のそれぞれは、符号化ブロックの半分の幅を有するが、符号化ブロックと同じ高さを有する。

30

**【0112】**

一例では、符号化ブロックは、図5Cに示すように2つの半分サイズのサブブロックに分割され、変換は各サブブロックに行われ、半分サイズ変換のそれぞれは、符号化ブロックの半分の高さを有するが、符号化ブロックと同じ幅を有する。

**【0113】**

一例では、符号化ブロックは、図5Dに示すように2つの4分の1サイズのサブブロックに分割され、変換は各サブブロックに行われ、4分の1サイズ変換のそれぞれは、符号化ブロックの4分の1の幅を有するが、符号化ブロックと同じ高さを有する。

**【0114】**

一例では、符号化ブロックは、図5Eに示すように2つの4分の1サイズのサブブロックに分割され、変換は各サブブロックに行われ、4分の1サイズ変換のそれぞれは、符号化ブロックの4分の1の高さを有するが、符号化ブロックと同じ幅を有する。

40

**【0115】**

他の実施形態では、符号化ブロックの幅が高さよりも大きい場合、符号化ブロックは、図6Aに示すような単一の変換、または3回のサブブロック変換のいずれかを適用することができる。選択はフラグによって通知され得る。

**【0116】**

図6Bに示す一例では、半分サイズ変換のそれぞれは、符号化ブロックの半分の幅を有するが、符号化ブロックと同じ高さを有する。

**【0117】**

50

図6Cに示す一例では、半分サイズ変換のそれぞれは、符号化ブロックの半分の高さを有するが、符号化ブロックと同じ幅を有する。

【0118】

他の実施形態では、符号化ブロックの高さが幅よりも大きい場合、符号化ブロックは、図7Aに示すような単一の変換、または複数の等しいより小さなサイズのいずれかを適用することができる。選択はフラグによって通知され得る。

【0119】

図7Bに示す一例では、符号化ブロックは、2つの半分サイズのサブブロックに分割され、変換は各サブブロックに行われ、半分サイズ変換のそれぞれは、符号化ブロックの半分の高さを有するが、符号化ブロックと同じ幅を有する。

10

【0120】

図7Cに示す一例では、符号化ブロックは、2つの半分サイズのサブブロックに分割され、変換は各サブブロックに行われ、半分サイズ変換のそれぞれは、符号化ブロックの半分の幅を有するが、符号化ブロックと同じ高さを有する。

【0121】

一例では、符号化ブロックは、2つの4分の1サイズのサブブロックに分割され、変換は各サブブロックに行われ、4分の1サイズ変換のそれぞれは、符号化ブロックの4分の1の高さを有するが、符号化ブロックと同じ幅を有する。

【0122】

一例では、符号化ブロックは、2つの4分の1サイズのサブブロックに分割され、変換は各サブブロックに行われ、4分の1サイズ変換のそれぞれは、符号化ブロックの4分の1の幅を有するが、符号化ブロックと同じ高さを有する。

20

【0123】

他の実施形態では、符号化ブロックの高さが幅よりも大きい場合、符号化ブロックは、図8Aに示すような単一の変換、または3回のサブブロック変換のいずれかを適用することができる。選択はフラグによって通知され得る。

【0124】

図8Bに示す一例では、サブブロック変換のそれぞれは、符号化ブロックと同じ幅を有する。

【0125】

図8Cに示す一例では、サブブロック変換のそれぞれは、符号化ブロックと同じ高さを有する。

30

【0126】

各TUについて、CU内のTUの位置に応じて、異なる変換集合がMTSに適用される場合がある。

【0127】

一実施形態では、各TUについて、CU内のTUの位置に応じて、1つの変換集合で利用可能な変換タイプの異なる順序をMTSに適用することができる。

【0128】

一例では、TUが図9Aおよび9Bに示すようにCUの左半分にある場合、MTSにおける水平変換選択のための変換集合内の変換タイプの順序は{DCT - 8、DST - 7}、あるいは、変換集合の順序は{DST - 7、DCT - 8}である。

40

【0129】

一例では、TUが図9Cおよび9Dに示すようにCUの右半分にある場合、MTSにおける水平変換選択のための変換集合内の変換タイプの順序は{DST - 7、DCT - 8}、あるいは、変換集合の順序は{DCT - 8、DST - 7}である。

【0130】

一例では、TUがCUの上半分にある場合、MTSにおける垂直変換選択のための変換集合内の変換タイプの順序は{DCT - 8、DST - 7}、あるいは、変換集合内の変換タイプの順序は{DST - 7、DCT - 8}である。

50

## 【 0 1 3 1 】

一例では、TUがCUの下半分にある場合、MTSにおける垂直変換選択のための変換集合内の変換タイプの順序は { DST - 7、DCT - 8 }、あるいは、変換集合内の変換タイプの順序は { DCT - 8、DST - 7 } である。

## 【 0 1 3 2 】

一例では、TUの左境界がCUの左境界と位置合わせされ、TUの右境界もCUの右境界と位置合わせされる場合、MTSにおける水平変換選択のための変換集合内の変換タイプの順序は、 { DCT - 8、DST - 7 }、あるいは、変換集合内の変換タイプの順序は { DST - 7、DCT - 8 } である。

## 【 0 1 3 3 】

一例では、TUの上部境界がCUの上部境界と位置合わせされ、TUの下部境界もCUの下部境界と位置合わせされる場合（たとえば、1 . c . i に示されている図）、MTSにおける垂直変換選択のための変換集合内の変換タイプの順序は、 { DCT - 8、DST - 7 }、あるいは、変換集合内の変換タイプの順序は { DST - 7、DCT - 8 } である。

## 【 0 1 3 4 】

一例では、TUの左境界がCUの左境界と位置合わせされず、TUの右境界もCUの右境界と位置合わせされない場合、MTSにおける水平変換選択のための変換集合内の変換タイプの順序は、 { DCT - 8、DST - 7 }、あるいは、変換集合内の変換タイプの順序は { DST - 7、DCT - 8 } である。

## 【 0 1 3 5 】

一例では、TUの上部境界がCUの上部境界と位置合わせされず、TUの下部境界もCUの下部境界と位置合わせされない場合、MTSにおける垂直変換選択のための変換集合内の変換タイプの順序は、 { DCT - 8、DST - 7 }、あるいは、変換集合内の変換タイプの順序は { DST - 7、DCT - 8 } である。

## 【 0 1 3 6 】

一実施形態では、各TUについて、CU内のTUの位置に応じて、1つの変換集合で利用可能な異なる数の変換タイプをMTSに適用することができる。

## 【 0 1 3 7 】

一例では、TUがCUの左半分または右半分にある場合、水平変換の変換集合は、2つの変換タイプ、DST - 7またはDCT - 8を有するか、あるいは、変換集合は、1つのデフォルト変換DCT - 2（またはDST - 7、またはDCT - 8）のみを有する。

## 【 0 1 3 8 】

一例では、TUがCUの上半分または下半分にある場合、垂直変換の変換集合は、2つの変換タイプ、DST - 7またはDCT - 8を有するか、あるいは、変換集合は、1つのデフォルト変換DCT - 2（またはDST - 7、またはDCT - 8）のみを有する。

## 【 0 1 3 9 】

他の実施形態では、上記の方法は、CU間にのみ適用され、CU内には適用されない。

## 【 0 1 4 0 】

一実施形態によれば、変換分割は、1つのレベルのみに制限され得る。これは、各符号化ブロックについて、符号化ブロックを複数のより小さなTUに分割した後、各TUをさらに分割することができないことを意味する。したがって、TU分割は再帰的ではない。

## 【 0 1 4 1 】

符号化ブロックについて、符号化ブロック全体に非ゼロのcb係数とcr係数がそれぞれあるかどうかを示すために、cbf\_flag\_cb\_depth0フラグとcbf\_flag\_cr\_depth0フラグが通知される場合がある。

## 【 0 1 4 2 】

符号化ブロックが複数のTUにさらに分割される場合、符号化ブロック全体に少なくとも1つの非ゼロ係数があることを示すデフォルト値でcbf\_flag\_cb\_depth0が通知されると、各TUについてcbf\_flag\_cb\_depth1フラグが通知される場合がある。

## 【 0 1 4 3 】

10

20

30

40

50

符号化ブロックが複数のTUにさらに分割される場合、ブロック全体に少なくとも1つの非ゼロ係数があることを示すデフォルト値で`cbf_flag_cr_depth0`が通知されると、各TUに対して`cbf_flag_cr_depth1`フラグが通知される場合がある。

【0144】

符号化ブロックが複数のTUにさらに分割される場合、最後に符号化されたTUについては、`cbf_flag_cb_depth1`（または`cbf_flag_cr_depth1`）フラグは通知されないが、少なくとも1つの非ゼロ`cb`（または`cr`）係数を示すデフォルト値で`cbf_flag_cb_depth0`（または`cbf_flag_cr_depth0`）が通知されると暗黙的に導出され、同じ符号化ブロック内の先行するすべてのTUの`cbf_flag_cb_depth1`（`cbf_flag_cr_depth1`）フラグは、非ゼロ係数がないことを示すデフォルト値（0など）で符号化される。

10

【0145】

暗黙的に導出された`cbf_flag_cb_depth1`（または`cbf_flag_cr_depth1`）フラグ値は、少なくとも1つの非ゼロ係数があることを示す値である場合がある。

【0146】

符号化ブロックについて、符号化ブロック全体に非ゼロの彩度係数があるかどうかを示すために、`cbf_flag_chroma_depth0`フラグが通知される場合がある。

【0147】

符号化ブロックが複数のTUにさらに分割される場合、少なくとも1つの非ゼロ係数があることを示すデフォルト値で`cbf_flag_chroma_depth0`が通知されると、各TUに対して`cbf_flag_chroma_depth1`フラグが通知される場合がある。

20

【0148】

各TUについて、`cbf_flag_chroma_depth1`フラグが少なくとも1つの非ゼロ彩度係数を示すデフォルト値（たとえば1）で通知される場合、`cbf_flag_cb_depth1`フラグと`cbf_flag_cr_depth1`フラグがさらに通知されて、少なくとも1つの非ゼロ`cb`および`cr`係数がそれぞれあるかどうかを示すことができる。

【0149】

各TUについて、`cbf_flag_chroma_depth1`フラグが少なくとも1つの非ゼロ彩度係数を示すデフォルト値（たとえば1）で通知され、`cbf_flag_cb_depth1`フラグが非ゼロ`cb`係数がないことを示すデフォルト値（たとえば0）でさらに通知される場合、`cbf_flag_cr_depth1`フラグは通知されない場合があるが、少なくとも1つの非ゼロ`cr`係数を示す値として暗黙的に導出される。

30

【0150】

符号化ブロックが複数のTUにさらに分割される場合、最後に符号化されたTUについては、`cbf_flag_chroma_depth1`フラグは通知されないが、少なくとも1つの非ゼロ係数を示すデフォルト値で`cbf_flag_chroma_depth0`が通知されると暗黙的に導出され、同じ符号化ブロック内の先行するすべてのTUの`cbf_flag_chroma_depth1`フラグは、非ゼロ係数がないことを示すデフォルト値（0など）で符号化される。

【0151】

暗黙的に導出された`cbf_flag_chroma_depth1`（または`cbf_flag_chroma_depth1`）フラグ値は、少なくとも1つの非ゼロ係数があることを示す値である場合がある。

40

【0152】

`root_cbf`は、符号化ブロック全体に非ゼロの輝度または彩度係数があるかどうかを示すために、符号化ブロック用に符号化できる。

【0153】

符号化ブロックが複数のTUにさらに分割される場合、最後に符号化されたTUについては、`cbf_flag_luma_depth1`フラグは通知されないが、同じ符号化ブロック内の先行するすべてのTUが、非ゼロの輝度または彩度係数がなく、現在のTUには非ゼロの彩度係数がないことを示すデフォルト値（たとえば、0）で符号化されている場合は、暗黙的に導出される。

【0154】

50

暗黙的に導出されたcbf\_flag\_luma\_depth1フラグ値は、少なくとも1つの非ゼロ輝度係数があることを示す値である。

【0155】

本開示において、残差ブロックのエネルギーの大部分をカバーするサブブロックは、このサブブロックによってカバーされるサンプルの測定値が、符号化ブロック全体によってカバーされるサンプルの測定値の所与のパーセンテージよりも大きいことを意味し得る。パーセンテージは、事前定義されたしきい値、たとえば、90%または80%にすることができる。測定値には、絶対値の合計、または二乗値の合計が含まれるが、これらに限定されない。

【0156】

インター符号化の場合、複数の予測ブロックを使用して現在のブロックの予測を生成する場合、たとえば、二重予測、またはインターイントラ予測では、これらの予測ブロックの予測サンプル値を使用して、現在のブロックの変換サイズおよび/または位置を決定できる。

【0157】

一実施形態では、仮想残差ブロック (VRB) は、一方の予測ブロックを他方から差し引くことによって計算され、次に、このVRBの残差エネルギー分布に従って、現在のブロックについて変換ブロックのサイズおよび位置が決定される。

【0158】

一実施形態では、サブブロック変換サイズは半分サイズのみとすることができ、残差ブロックのエネルギーの大部分をカバーするサブブロック位置は、サブブロック変換位置として決定される。

【0159】

他の実施形態では、サブブロック変換サイズは、たとえば、幅が4分の1、または高さが4分の1、または幅と高さの両方が半分である4分の1のサイズ、および残差ブロックのエネルギーの大部分をカバーするサブブロック位置は、サブブロック変換位置として決定される。

【0160】

他の実施形態では、サブブロック変換サイズは、半分サイズまたは4分の1サイズのいずれかとすることができ、サイズ選択は、現在のブロックについて明示的に通知され、次に、選択されたサブブロック変換サイズが与えられると、残差ブロックのエネルギーの大部分をカバーするサブブロックのブロック位置は、サブブロックの変換位置として決定される。

【0161】

他の実施形態では、サブブロック変換サイズおよび位置の両方が、このVRB上で異なるサブブロックサイズおよび位置を試みることによって決定され、特定のサブブロック変換サイズおよび位置が残差エネルギーの大部分をカバーする場合、それは、サブブロック変換のサイズと位置として決定される。

【0162】

異なるサブブロックサイズに対して、異なる残差パーセンテージしきい値を使用することができる。たとえば、半分サイズのサブブロック変換サイズの残差パーセンテージしきい値はK0であり、4分の1サイズのサブブロック変換サイズの残差パーセンテージしきい値はK1であり、K1はK0よりも大きくなる。

【0163】

一実施形態では、最適なサブブロック変換位置は、候補サブブロック変換サイズごとに個別に決定され、次に、サンプルあたりの残差エネルギー値が最小になる候補サブブロック変換サイズが、サブブロック変換サイズとして選択される。

【0164】

他の実施形態では、サブブロック位置のサブセットのみが許可される。たとえば、現在のブロックの左上のx方向とy方向のオフセットは、値の倍数である必要がある。値は、事

10

20

30

40

50

前定義されているか、SPS、PPS、スライスヘッダ、タイルヘッダなどのビットストリームで通知され得る。サブブロック変換でカバーされていないサンプルの場合、ゼロと仮定されうるため、係数は通知されない。

【0165】

インター符号化の場合、1つの予測ブロックのみを使用して現在のブロックの予測を生成する場合（たとえば、単一予測）、第2の仮想予測ブロックが配置され、予測ブロックとこの第2の仮想予測ブロックを使用して、現在のブロックの変換サイズおよび/または位置を決定できる。

【0166】

一実施形態では、第2の仮想予測ブロックは、現在のブロックの動きベクトルをミラーリングし、異なる参照ピクチャ内に参照ブロックを配置することによって導出される。

10

【0167】

一実施形態では、第2の仮想予測ブロックは、現在のブロックの異なる参照フレームを使用する隣接ブロックを探索することによって導出され、次に、第2の仮想予測ブロックは、現在のブロックの位置、この隣接ブロックの動きベクトル、および関連づけられる参照フレームを使用することによって導出される。

【0168】

他の実施形態では、第2の仮想予測ブロックは、平面内モードによって導出される。

【0169】

他の実施形態では、予測ブロックおよび第2の仮想予測ブロックを用いて、仮想残差ブロック（VRB）は、一方の予測ブロックを他方から差し引くことによって計算することができ、次に、このVRBの残差エネルギー分布に従って、現在のブロックについて変換ブロックのサイズおよび位置を決定することができる。次に、上記の技術を適用することができる。

20

【0170】

インター符号化の場合、フルサイズの変換サイズを使用するか、それよりも小さい変換サイズを使用するかは、各符号化ブロックについてフラグを使用して通知できる。

【0171】

一実施形態では、結果として生じるサブブロック変換サイズがサポートされていない場合、たとえば、2点変換を使用する必要がある場合、このサブブロックサイズを使用することはできない。

30

【0172】

一実施形態では、現在のブロックが異なるサンプルに対して異なる動きベクトル、たとえば、サブブロック動きを有する場合、サブブロックサイズを使用することができないので、サブブロック変換を使用することができない。

【0173】

一実施形態では、現在のブロックが異なるサンプルに対して異なる動きベクトル、たとえば、サブブロック動きを有する場合、仮想残差ブロックも複数のサブ仮想残差ブロックとして分割され、残差サンプルは、関連する動きベクトルと予測ブロックを使用して各サブ予測ブロックに対して別々に導出される。

40

【0174】

サブブロック変換のサイズと位置は、輝度サンプルによって導出できるが、彩度サンプルで共有される。一実施形態では、サブブロック変換のサイズと位置は、輝度および彩度サンプルの両方によって導出することができ、現在の輝度および彩度ブロックの両方に使用することができる。一実施形態では、サブブロック変換のサイズと位置は、輝度サンプルに対してのみ導出および使用され得る。

【0175】

仮想残差ブロックを導出するために輝度サンプルと彩度サンプルの両方が使用される場合、各サンプルの残差エネルギーは、輝度サンプルと彩度サンプルの両方の残差エネルギーの加重和によって測定され得る。重み付けは、事前定義されているか、SPS、PPS、ス

50

ライスヘッダ、タイルヘッダなどのビットストリームで通知され得る。

【0176】

上記の手法を適用できるかどうかは、VPS、PPS、SPS、スライス、タイル、タイルグループ、CTUヘッダなどの高レベルの構文要素で通知される。

【0177】

図14は、ビデオシーケンスを符号化するためにコーディングユニット(CU)を1つまたは複数の変換ユニット(TU)に分割するための例示的なプロセス1400のフローチャートである。いくつかの実装形態では、図14の1つまたは複数のプロセスブロックは、デコーダ1110によって実行され得る。いくつかの実装形態では、図14の1つまたは複数のプロセスブロックは、エンコーダ1103などの、デコーダ1110とは別のまたはそれを含む別の装置または装置のグループによって実行され得る。

10

【0178】

図14に示すように、プロセス1400は、CUの高さを決定するステップを含み得る(ブロック1410)。

【0179】

図14にさらに示すように、プロセス1400は、CUの幅を決定するステップを含み得る(ブロック1420)。

【0180】

図14にさらに示すように、プロセス1400は、CUの高さおよびCUの幅に基づいて、1つまたは複数のTUのそれぞれのTUサイズを決定するステップを含み得る(ブロック1430)

20

【0181】

図14にさらに示すように、プロセス1400は、CUの高さおよびCUの幅に基づいて、1つまたは複数のTUの1つまたは複数のTU位置を決定するステップを含み得る(ブロック1440)。

【0182】

図14にさらに示すように、プロセス1400は、1つまたは複数のTUの決定されたサイズおよび1つまたは複数のTUの1つまたは複数の位置に基づいて、CUを1つまたは複数のTUに分割するステップを含み得る(ブロック1450)。

【0183】

一実施形態によれば、CUの高さはCUの幅に等しくてもよく、1つまたは複数のTUの数は4つであってもよく、1つまたは複数のTUのそれぞれの高さは、CUの高さの半分であってもよく、1つまたは複数のTUのそれぞれの幅は、CUの幅の半分であってもよい。

30

【0184】

一実施形態によれば、CUの幅はCUの高さよりも大きくてもよく、1つまたは複数のTUの数は2つであってもよく、1つまたは複数のTUのそれぞれの高さは、CUの高さに等しくてもよく、1つまたは複数のTUのそれぞれの幅は、CUの幅の半分であってもよい。

【0185】

一実施形態によれば、CUの高さはCUの幅よりも大きくてもよく、1つまたは複数のTUの数は2つであってもよく、1つまたは複数のTUのそれぞれの高さは、CUの高さの半分であってもよく、1つまたは複数のTUのそれぞれの幅は、CUの幅に等しくてもよい。

40

【0186】

一実施形態によれば、1つまたは複数のTUのTUに利用可能な変換タイプの順序は、1つまたは複数のTU位置中のTUの位置に基づいて決定することができる。

【0187】

一実施形態によれば、1つまたは複数のTUは、より小さなTUに分割されていない。

【0188】

一実施形態によれば、プロセス1400は、CUのインター予測符号化を行うために複数の予測ブロックが使用されることを決定するステップ、および複数の予測ブロックの予測サンプル値に基づいて、TUサイズおよび1つまたは複数のTU位置の中から少なくとも1つを

50

決定するステップを含み得る。

【0189】

一実施形態によれば、複数の予測ブロックは、第1の予測ブロックおよび第2の予測ブロックを含んでもよく、プロセス1400は、第1の予測ブロックから第2の予測ブロックを差し引くことによって仮想残差ブロック（VRB）を計算するステップ、およびVRBの残差エネルギー分布に基づいて、TUサイズおよび1つまたは複数のTU位置の中から少なくとも1つを決定するステップを含み得る。

【0190】

一実施形態によれば、1つまたは複数のTU位置は、潜在的なTU位置のサブセットから選択され得る。

【0191】

一実施形態によれば、サブセットは、シーケンス・パラメータ集合、ピクチャパラメータ集合、スライスヘッダ、またはタイルヘッダのうちの1つで通知することができる。

【0192】

一実施形態によれば、プロセス1400は、CUのインター予測符号化を行うために単一の予測ブロックが使用されることを決定するステップ、仮想予測ブロックを配置するステップ、および予測ブロックおよび仮想予測ブロックに基づいて、TUサイズおよび1つまたは複数のTU位置の中から少なくとも1つを決定するステップを含み得る。

【0193】

一実施形態によれば、仮想予測ブロックは、CUの動きベクトルをミラーリングし、予測ブロックの参照ピクチャとは異なる参照ピクチャ内に仮想予測ブロックを配置することによって導出することができる。

【0194】

一実施形態によれば、CUは、第1の参照フレームに関連付けることができ、プロセス1400は、第1の参照フレームとは異なる第2の参照フレームに関連付けられた隣接ブロックを決定するステップ、CUの位置、隣接ブロックの動きベクトル、および第2の参照フレームに基づいて仮想予測ブロックを導出するステップをさらに含み得る。

【0195】

図14は、プロセス1400の例示的なブロックを示すが、いくつかの実装形態では、プロセス1400は、図14に示すものに対して追加のブロック、より少ないブロック、異なるブロック、または異なる配置のブロックを含み得る。さらに、または代わりに、プロセス1400の2つ以上のブロックを並行して実施することができる。

【0196】

さらに、提案された方法は、処理回路（たとえば、1つまたは複数のプロセッサまたは1つまたは複数の集積回路）によって実装され得る。一例では、1つまたは複数のプロセッサは、非一時的なコンピュータ可読媒体に格納されているプログラムを実行して、提案された方法の1つまたは複数を実行する。

【0197】

上記の手法は、コンピュータ可読命令を使用してコンピュータソフトウェアとして実装でき、1つまたは複数のコンピュータ可読媒体に物理的に格納できる。たとえば、図15は、開示された主題の特定の実装形態を実装するのに適したコンピュータシステム（1500）を示す。

【0198】

コンピュータソフトウェアは、任意の適切な機械コードまたはコンピュータ言語を使用してコーディングされえ、アセンブリ、コンパイル、リンク、または同様のメカニズムの対象となり、コンピュータ中央処理装置（CPU）、グラフィック処理装置（GPU）などによる直接、または解釈、マイクロコードの実行などを通じて実行されうる命令を含むコードを作成する。

【0199】

命令は、たとえば、パーソナルコンピュータ、タブレットコンピュータ、サーバ、スマ

10

20

30

40

50

ートフォン、ゲーム装置、モノのインターネット装置などを含む、様々なタイプのコンピュータまたはその構成要素上で実行されうる。

【0200】

コンピュータシステム(1500)について図15に示される構成要素は、本質的に例示であり、本開示の実施形態を実装するコンピュータソフトウェアの使用または機能の範囲に関していかなる制限も示唆することを意図しない。また、構成要素の構成は、コンピュータシステム1500の例示的な実施形態に示されている構成要素のいずれか1つまたはそれらの組み合わせに関する依存性または要件を有するものとして解釈されるべきではない。

【0201】

コンピュータシステム1500は、特定のヒューマンインタフェース入力装置を含み得る。そのようなヒューマンインタフェース入力装置は、たとえば、触覚入力(キーストローク、スワイプ、データグローブの動きなど)、オーディオ入力(音声、拍手など)、視覚入力(ジェスチャーなど)、嗅覚入力(図示せず)など、1人または複数のユーザによる入力に応答し得る。ヒューマン・インタフェース・装置を使用して、音声(会話、音楽、環境音など)、画像(スキャンした画像、静止画像カメラから取得される写真画像など)、ビデオ(2次元ビデオ、立体ビデオを含む3次元ビデオなど)など、人間による意識的な入力に必ずしも直接関係しない特定のメディアをキャプチャすることもできる。

10

【0202】

入力ヒューマン・インタフェース・装置は、キーボード(1501)、マウス(1502)、トラックパッド(1503)、タッチスクリーン(1510)、データグローブ(1204)、ジョイスティック(1505)、マイク(1506)、スキャナ(1507)、カメラ(1508)などを含み得る(それぞれ1つのみ図示)。

20

【0203】

コンピュータシステム(1500)はまた、特定のヒューマンインタフェース出力装置を含み得る。そのようなヒューマンインタフェース出力装置は、たとえば、触覚出力、音、光、および嗅覚/味覚を通じて、1人または複数のユーザの感覚を刺激している可能性がある。そのようなヒューマンインタフェース出力装置は、触覚出力装置(たとえば、タッチスクリーン1510、データグローブ1204、またはジョイスティック1505による触覚フィードバックだが、入力装置として機能しない触覚フィードバック装置も存在する場合がある)、オーディオ出力装置(スピーカ1509、ヘッドホン(図示せず)など)、視覚出力装置(カソード光線管(CRT)スクリーン、液晶ディスプレイ(LCD)スクリーン、プラズマスクリーン、有機発光ダイオード(OLED)スクリーンを含むスクリーン1510)などであり、それぞれタッチスクリーン入力機能があってもなくてもよく、それぞれ触覚フィードバック機能があってもなくてもよく、ステレオグラフィック出力、仮想現実の眼鏡(図示せず)、ホログラフィックディスプレイおよびスモークタンク(図示せず)などの手段を介して2次元の視覚的出力または3次元以上の出力を出力できるものもある)、およびプリンタ(図示せず)を含み得る。

30

【0204】

コンピュータシステム(1500)はまた、人間がアクセス可能なストレージ装置およびそれらの関連付けられる媒体、たとえば、CD/DVDなどの媒体(1521)とともにCD/DVD ROM/RW(1520)を含む光学媒体、サムドライブ(1522)、取り外し可能なハードドライブまたはソリッド・ステート・ドライブ(1523)、テープおよびフロッピーディスク(図示せず)などのレガシー磁気媒体、セキュリティドングル(図示せず)などの専用のROM/ASIC/PLDベースの装置などを含み得る。

40

【0205】

当業者はまた、ここで開示される主題に関連して使用される「コンピュータ可読媒体」という用語は、送信媒体、搬送波、または他の一時的な信号を包含しないことを理解するべきである。

【0206】

コンピュータシステム(1500)は、1つまたは複数の通信ネットワーク(1555)への

50

インタフェースも含むことができる。ネットワーク(1555)は、たとえば、無線、有線、光であり得る。さらに、ネットワーク(1555)は、ローカル、広域、大都市圏、車両および産業、リアルタイム、遅延耐性などであり得る。ネットワーク(1555)の例は、イーサネット、無線LAN、GSM、3G、4G、5G、LTEなどを含むセルラーネットワークなどのローカル・エリア・ネットワーク、ケーブルテレビ、衛星テレビ、地上波放送テレビを含むTV有線または無線ワイド・エリア・デジタル・ネットワーク、車両用、CANBusを含む産業用などを含む。特定のネットワーク(1555)は一般に、特定の汎用データポートまたは周辺バス(1549)(たとえば、コンピュータシステム(1500)のUSBポートなど)に接続された外部ネットワークインタフェース・アダプタ(1554)を必要とし、他のものは、一般に、以下に説明するようにシステムバスに接続することにより、コンピュータシステム(1500)のコアに統合される(たとえば、PCコンピュータシステムへのイーサネットインタフェースまたはスマートフォン・コンピュータ・システムへのセルラー・ネットワーク・インタフェース)。これらのネットワーク(1555)のいずれかを使用して、コンピュータシステム(1500)は他のエンティティと通信できる。このような通信は、単方向、受信のみ(たとえば、放送TV)、単方向送信のみ(たとえば、CANbusから特定のCANbus装置)、または双方向、たとえば、ローカルエリアまたはワイド・エリア・デジタル・ネットワークを使用した他のコンピュータシステムへの通信である。上記のように、特定のプロトコルとプロトコルスタックは、これらのネットワーク(1555)とネットワークインタフェース(1554)のそれぞれで使用できる。

#### 【0207】

前述のヒューマン・インタフェース・装置、ヒューマンアクセス可能ストレージ装置、およびネットワークインタフェースは、コンピュータシステム(1500)のコア(1540)に接続することができる。

#### 【0208】

コア1540には、1つまたは複数の中央処理装置(CPU)1541、グラフィック処理装置(GPU)1542、フィールド・プログラマブル・ゲート領域(FPGA)1543形式の特殊なプログラマブル処理装置、特定のタスク用のハードウェアアクセラレータ1544などを含めることができる。これらの装置は、読み取り専用メモリ(ROM)1545、ランダムアクセスメモリ(RAM)1546、ユーザがアクセスできない内部ハードドライブ、ソリッド・ステート・ドライブ(SSD)などの内部大容量記憶装置およびそのようなもの(1547)とともに、システムバス(1248)を介して接続することができる。一部のコンピュータシステムでは、システムバス1248に1つまたは複数の物理プラグの形でアクセスして、追加のCPU、GPUなどによる拡張を可能にすることができる。周辺機器は、コアのシステムバス1248に直接、または周辺バス1549を介して接続することができる。周辺バスのアーキテクチャには、周辺成分相互接続(PCI)、USBなどが含まれる。

#### 【0209】

CPU1541、GPU1542、FPGA1543、およびアクセラレータ1544は、組み合わせて前述のコンピュータコードを構成できる特定の命令を実行できる。そのコンピュータコードは、ROM1545またはRAM1546に格納できる。移行データはRAM1546にも格納できるが、永続データは、たとえば内部大容量記憶装置1547に格納できる。1つまたは複数のCPU1541、GPU1542、大容量記憶装置1547、ROM1545、RAM1546などと密接に関連付けることができるキャッシュメモリを使用することにより、任意のメモリ装置への高速格納および取り出しが可能になる。

#### 【0210】

コンピュータ可読媒体は、様々なコンピュータ実装動作を実行するためのコンピュータコードを内部に有することができる。媒体およびコンピュータコードは、本開示の目的のために特別に設計および構築されたものであり得るか、またはそれらは、コンピュータソフトウェア技術の当業者に周知であり利用可能な種類のものであり得る。

#### 【0211】

限定ではなく一例として、アーキテクチャ1500、特にコア1540を有するコンピュータ

10

20

30

40

50

システムは、1つまたは複数の有形のコンピュータ可読媒体に組み込まれたソフトウェアを実行するプロセッサ（CPU、GPU、FPGA、アクセラレータなどを含む）の結果として機能を提供することができる。そのようなコンピュータ可読媒体は、上で紹介したユーザがアクセス可能な大容量記憶装置、およびコア内部大容量記憶装置1547やROM1545などの非一時的な性質を持つコア1540の特定の記憶装置に関連付けられた媒体であり得る。本開示の様々な実施形態を実装するソフトウェアは、そのような装置に格納され、コア1540によって実行され得る。コンピュータ可読媒体は、特定のニーズに従って、1つまたは複数のメモリ装置またはチップを含むことができる。ソフトウェアは、コア1540および特にその中のプロセッサ（CPU、GPU、FPGAなどを含む）に、RAM1546に格納されたデータ構造の定義やソフトウェアで定義されたプロセスに従ってそのようなデータ構造を変更することを含む、ここで説明する特定のプロセスまたは特定のプロセスの特定の部分を実行させることができる。加えて、または代替として、コンピュータシステムは、ここで説明する特定のプロセスまたは特定のプロセスの特定の部分を実行するために、ソフトウェアの代わりにまたはソフトウェアと一緒に動作することができる回路（たとえばアクセラレータ1544）にハードワイヤードまたは他の方法で組み込まれたロジックの結果として機能を提供できる。ソフトウェアへの言及はロジックを含むことができ、必要に応じてその逆も可能である。コンピュータ可読媒体への言及は、実行のためのソフトウェア、実行のためのロジックを具体化する回路、またはその両方を必要に応じて格納する回路（集積回路（IC）など）を包含することができる。本開示は、ハードウェアとソフトウェアの任意の適切な組み合わせを包含する。

10

20

#### 【0212】

本開示はいくつかの例示的な実施形態を説明してきたが、本開示の範囲内にある変更、置換、および様々な代替均等物が存在する。したがって、当業者は、本明細書では明示的に示されていないか、または記載されていないが、本開示の原理を具現化し、したがってその精神および範囲内にある多数のシステムおよび方法を考案できることが理解されよう。

#### 【0213】

頭字語：

HEVC：High Efficiency Video Coding 高効率ビデオ符号化

HDR：high dynamic range ハイ・ダイナミック・レンジ

SDR：standard dynamic range 標準ダイナミックレンジ

VVC：Versatile Video Coding 多用途ビデオ符号化

JVET：Joint Video Exploration Team 共同ビデオ探索チーム

CU：Coding Unit コーディングユニット

PU：Prediction Unit 予測ユニット

MTS：Multiple Transform Selection 多重変換選択

EMT：Enhanced Multiple Transform 拡張型多重変換

AMT：Adaptive Multiple Transform 適応型多重変換

SVT：Spatially Varying Transform 空間的に変化する変換

SBT：Sub-block Transform サブブロック変換

VPS：Video Parameter Set ビデオパラメータ集合

PPS：Picture Parameter Set ピクチャパラメータ集合

SPS：Sequence Parameter Set シーケンス・パラメータ集合

#### 【符号の説明】

#### 【0214】

300 ストリーミングシステム

1000 通信システム

1010, 1020, 1030, 1040 端末

1050 ネットワーク

1101 ビデオソース

1102 サンプルストリーム

30

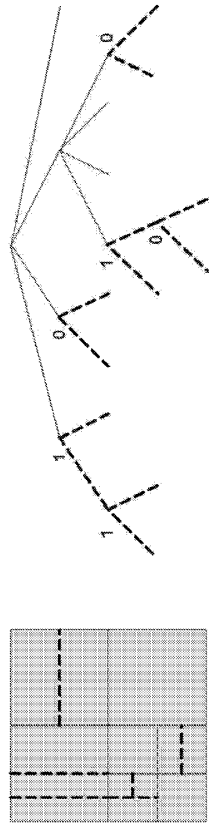
40

50

1103	エンコーダ	
1104	符号化されたビデオビットストリーム	
1105	ストリーミングサーバ	
1106, 1108	ストリーミングクライアント	
1107, 1109	符号化されたビデオビットストリームのコピー	
1110	デコーダ	
1111	ビデオサンプルストリーム	
1112	ディスプレイ	
1113	キャプチャサブシステム	
1204	データグループ	10
1210	受信器	
1212	チャンネル	
1215	バッファメモリ	
1220	パーサ	
1221	シンボル	
1248	システムバス	
1251	スケーラ / 逆変換ユニット	
1252	イントラ予測ユニット	
1253	動き補償予測ユニット	
1255	アグリゲータ	20
1256	ループ・フィルタ・ユニット	
1257	参照ピクチャバッファ	
1258	現在の参照ピクチャメモリ	
1400	プロセス	
1500	コンピュータシステム	
1505	ジョイスティック	
1509	スピーカ	
1510	タッチスクリーン	
1540	コア	
1541	CPU	30
1542	GPU	
1543	FPGA	
1544	ハードウェアアクセラレータ	
1545	ROM	
1546	RAM	
1547	大容量記憶装置	
1549	周辺バス	
1554	ネットワークインタフェース	
1555	ネットワーク	40

【 図面 】  
【 図 1 】

FIG. 1



【 図 2 B 】

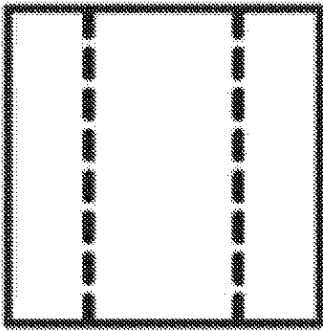


FIG. 2B

【 図 2 A 】

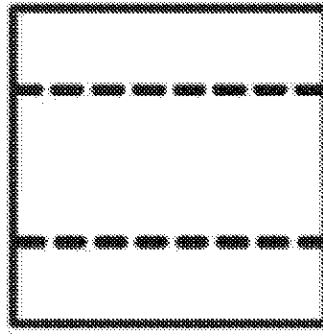
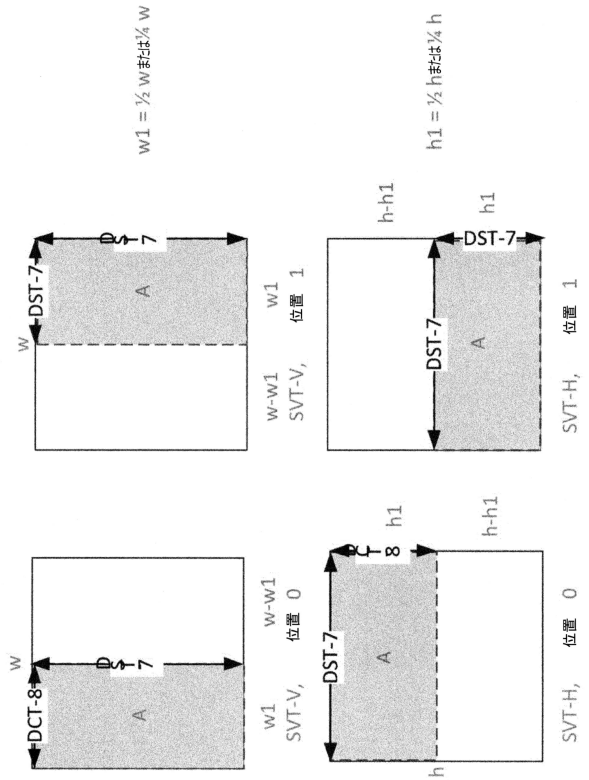


FIG. 2A

【 図 3 】



10

20

30

40

50

【 4 A 】

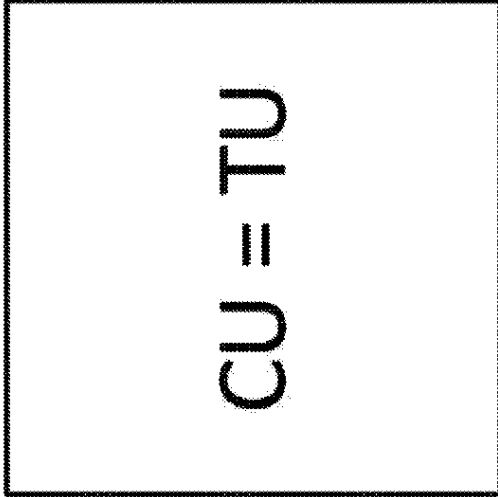


FIG. 4A

【 4 B 】

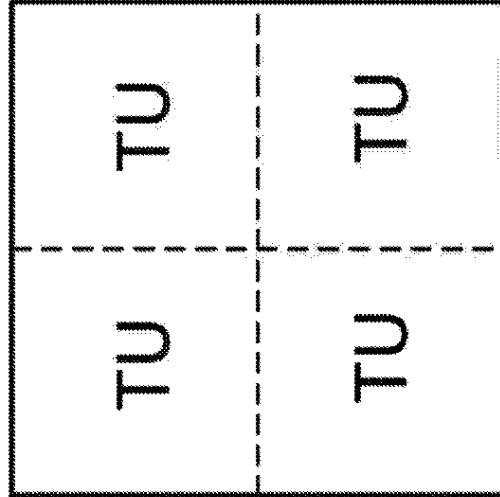


FIG. 4B

10

【 5 A 】



FIG. 5A

【 5 B 】

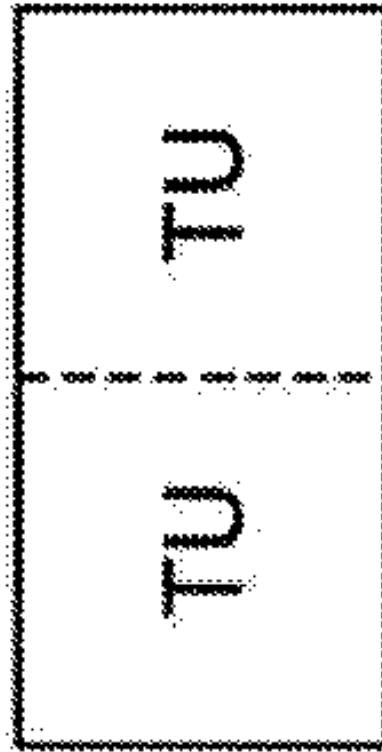


FIG. 5B

20

30

40

50

【 5 C 】

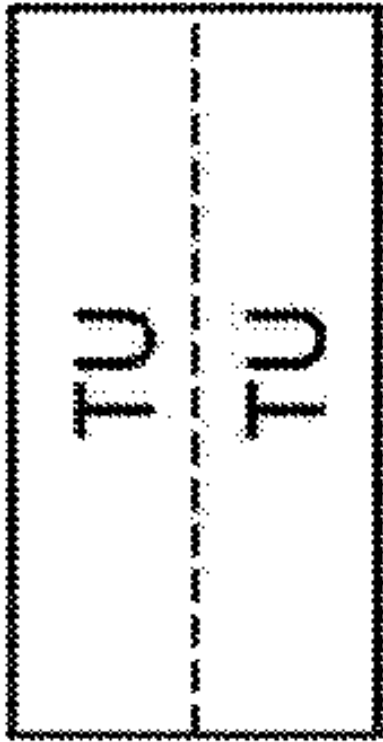


FIG. 5C

【 5 D 】

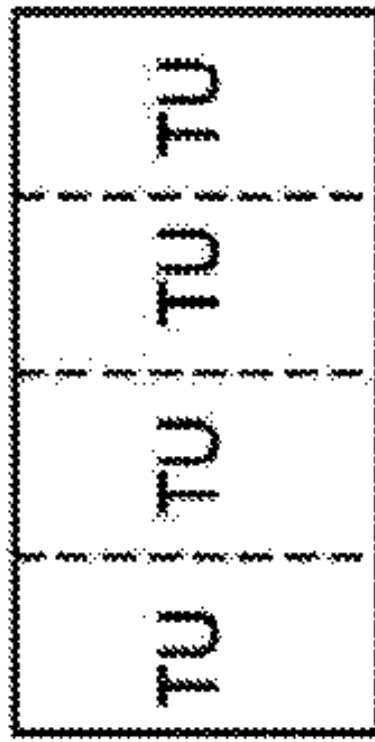


FIG. 5D

【 5 E 】

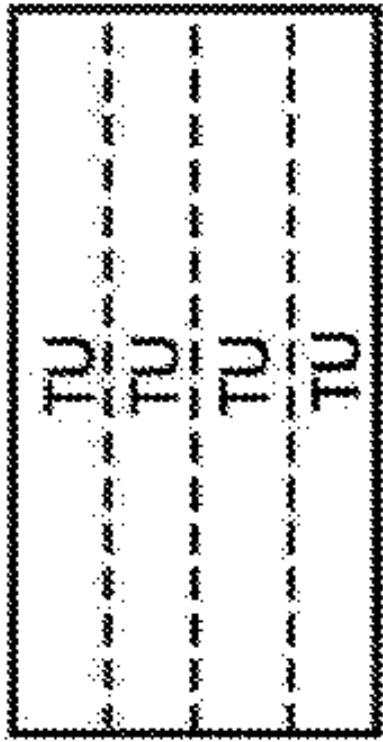


FIG. 5E

【 6 A 】



FIG. 6A

10

20

30

40

50

【 6 B 】

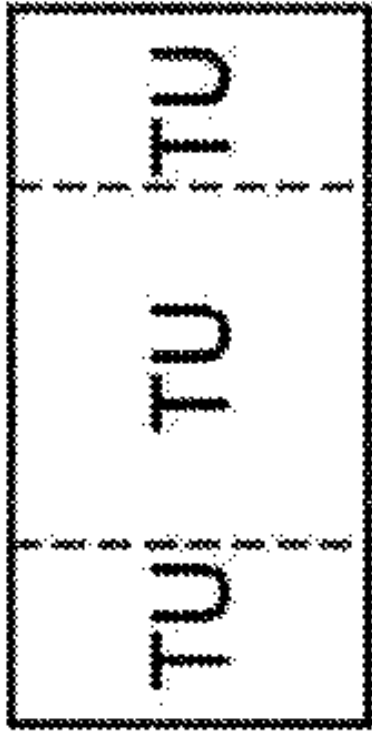


FIG. 6B

【 6 C 】

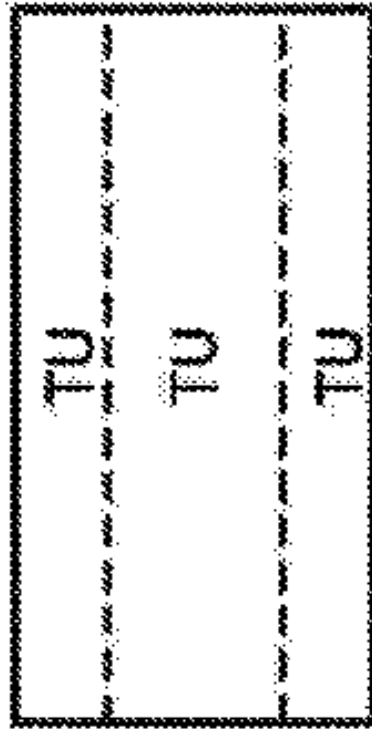


FIG. 6C

【 7 A 】



FIG. 7A

【 7 B 】



FIG. 7B

10

20

30

40

50

【 7 C 】

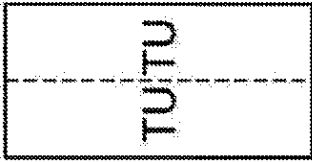


FIG. 7C

【 8 A 】



FIG. 8A

【 8 B 】

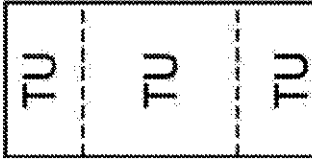


FIG. 8B

【 8 C 】

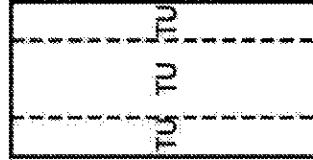


FIG. 8C

10

【 9 A 】

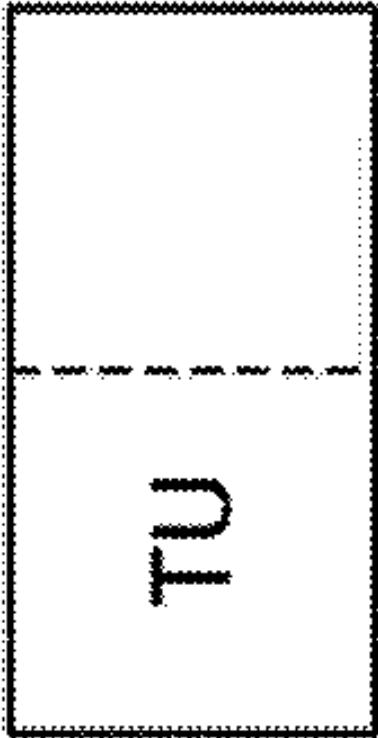


FIG. 9A

【 9 B 】

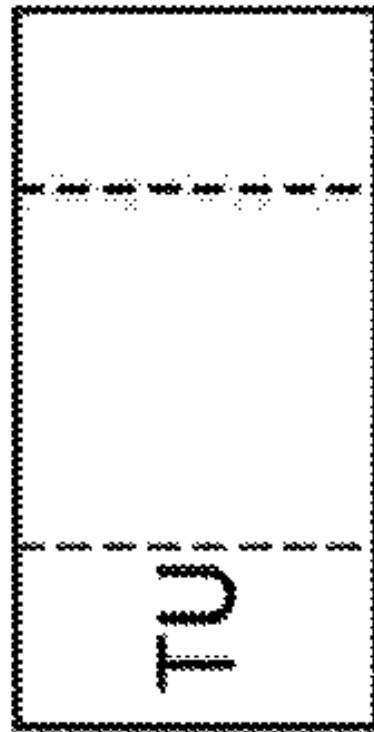


FIG. 9B

20

30

40

50

【図 9 C】

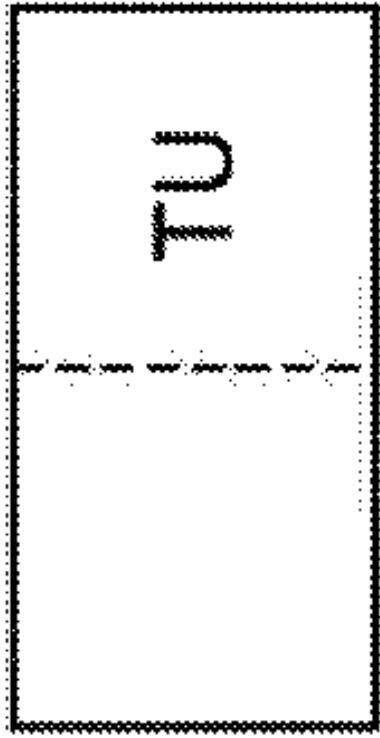


FIG. 9C

【図 9 D】

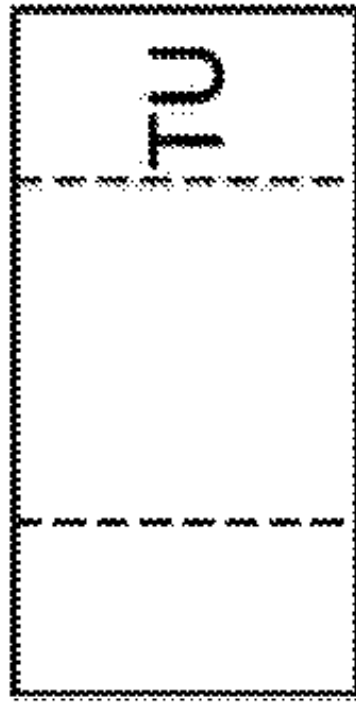
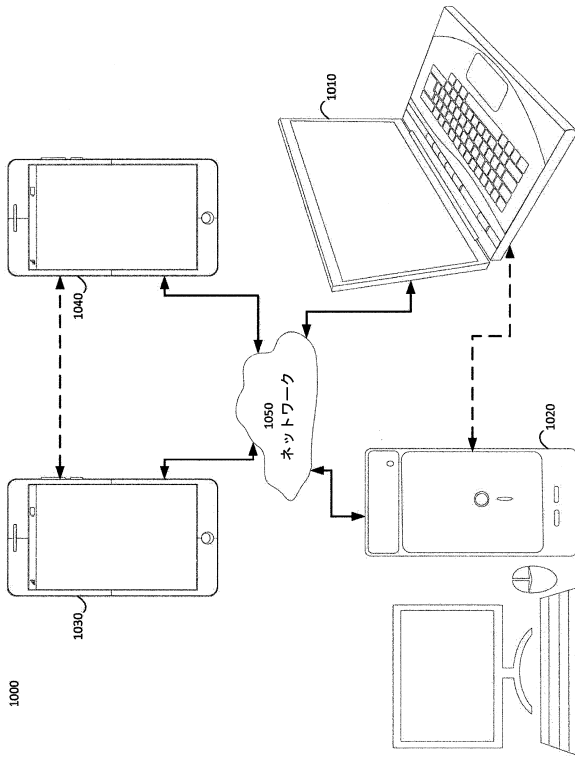


FIG. 9D

【図 10】



1000

【図 11】

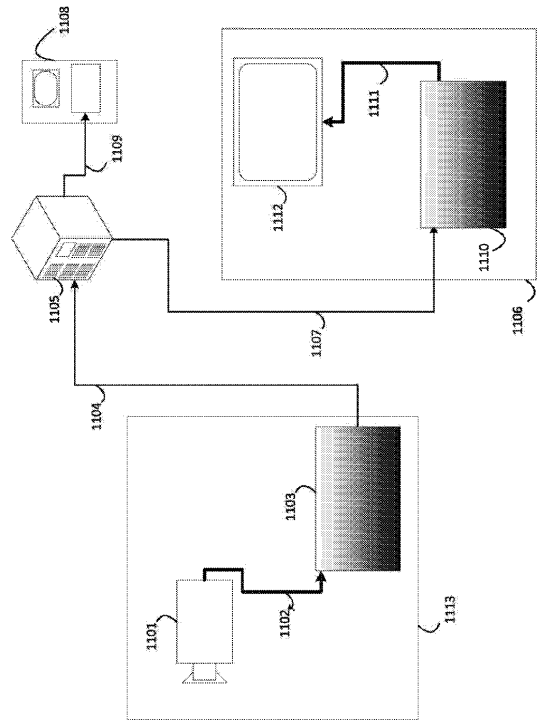


FIG. 11

10

20

30

40

50



## フロントページの続き

(33)優先権主張国・地域又は機関

米国(US)

(31)優先権主張番号 16/694,338

(32)優先日 令和1年11月25日(2019.11.25)

(33)優先権主張国・地域又は機関

米国(US)

アメリカ合衆国・カリフォルニア・94306・パロ・アルト・パーク・ブールバード・2747  
・テンセント・アメリカ・エルエルシー内

(72)発明者 シアン・リ

アメリカ合衆国・カリフォルニア・94306・パロ・アルト・パーク・ブールバード・2747  
・テンセント・アメリカ・エルエルシー内

(72)発明者 シャン・リュウ

アメリカ合衆国・カリフォルニア・94306・パロ・アルト・パーク・ブールバード・2747  
・テンセント・アメリカ・エルエルシー内

審査官 久保 光宏

(56)参考文献 米国特許出願公開第2018/0131962(US, A1)

韓国公開特許第10-2017-0132036(KR, A)

米国特許出願公開第2019/0289301(US, A1)

大久保 榮 監修, 「インプレス標準教科書シリーズ H.265/HEVC教科書」, 初版, 日本, 株式会社インプレスジャパン, 2013年10月21日, 第108~114頁, ISBN: 978-4-8443-3468-2.

(58)調査した分野 (Int.Cl., DB名)

H04N19/00-19/98

CSDB(日本国特許庁)

IEEEExplore(IEEE)