

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号  
特許第7337164号  
(P7337164)

(45)発行日 令和5年9月1日(2023.9.1)

(24)登録日 令和5年8月24日(2023.8.24)

(51)国際特許分類	F I
H 0 4 N 19/13 (2014.01)	H 0 4 N 19/13
H 0 4 N 19/18 (2014.01)	H 0 4 N 19/18
H 0 4 N 19/61 (2014.01)	H 0 4 N 19/61
H 0 4 N 19/157(2014.01)	H 0 4 N 19/157

請求項の数 12 (全76頁)

(21)出願番号	特願2021-531755(P2021-531755)	(73)特許権者	520353802 テンセント・アメリカ・エルエルシー 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
(86)(22)出願日	令和2年2月12日(2020.2.12)	(74)代理人	100107766 弁理士 伊東 忠重
(65)公表番号	特表2022-515029(P2022-515029 A)	(74)代理人	100070150 弁理士 伊東 忠彦
(43)公表日	令和4年2月17日(2022.2.17)	(74)代理人	100135079 弁理士 宮崎 修
(86)国際出願番号	PCT/US2020/017860	(72)発明者	ジャオ, シン 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・ブルバ ード 2 7 4 7 テンセント アメリカ エル 最終頁に続く
(87)国際公開番号	WO2020/167905		
(87)国際公開日	令和2年8月20日(2020.8.20)		
審査請求日	令和3年6月3日(2021.6.3)		
(31)優先権主張番号	62/804,666		
(32)優先日	平成31年2月12日(2019.2.12)		
(33)優先権主張国・地域又は機関	米国(US)		
(31)優先権主張番号	16/787,628		
(32)優先日	令和2年2月11日(2020.2.11)		
	最終頁に続く		

(54)【発明の名称】 ビデオ復号及び符号化の方法、装置並びにコンピュータプログラム

(57)【特許請求の範囲】

【請求項1】

デコーダが実行するビデオ復号の方法であって、  
符号化ビデオビットストリームから変換ブロック(TB)の符号化情報を復号するステップ  
であって、前記符号化情報は、二次変換が適用される前記TBの領域を示し、前記領域は、  
前記二次変換によって計算された変換係数を有する第1の部分領域と、第2の部分領域とを  
含む、ステップと、  
前記TBにおける変換係数について、前記変換係数を決定するために使用される隣接変換  
係数が前記第2の部分領域に位置するか否かを決定するステップと、  
前記隣接変換係数が前記第2の部分領域に位置すると決定された場合、前記隣接変換係  
数のデフォルト値に従って前記変換係数のコンテキスト値を決定するステップと、  
前記TBにおけるサンプルについて、前記変換係数の前記コンテキスト値を用いてエント  
ロピー復号することによって前記変換係数を決定し、前記変換係数に基づいて前記サンプ  
ルを復元するステップと  
を含む方法。

【請求項2】

前記TBにおける前記変換係数は、第1の係数グループ(CG)内の複数の変換係数のうち1  
つであり、前記第1のCGについての第1のCGフラグは、前記変換係数のうち少なくとも1  
つが非ゼロ変換係数であるか否かを示し、  
変換係数を含む第2のCGは、前にエントロピー復号されており、前記第1のCGの隣接す

るCGであり、

当該方法は、

前記第2のCGの位置を決定するステップと、

前記第2のCGが前記第2の部分領域に位置すると決定された場合、前記第2のCGについての第2のCGフラグのデフォルト値に基づいて前記第1のCGフラグのコンテキスト値を決定し、前記第1のCGフラグの前記コンテキスト値を用いてエントロピー復号することによって前記第1のCGフラグを決定するステップと

を更に含む、請求項1に記載の方法。

【請求項3】

前記TBにおける前記変換係数は、第1のCG内の複数の変換係数のうち1つであり、前記第1のCGについての第1のCGフラグは、前記変換係数のうち少なくとも1つが非ゼロ変換係数であるか否かを示し、

10

第2のCGは、第1の変換係数と第2の変換係数とを含み、前記第2のCGは、前にエントロピー復号されており、前記第1のCGの隣接するCGであり、

当該方法は、

前記第2のCGの位置を決定するステップと、

前記第2の変換係数を含む前記第2のCGの一部が前記第2の部分領域に位置し、前記第1の変換係数が非ゼロ変換係数である場合、前記非ゼロ変換係数である前記第1の変換係数に基づいて前記第1のCGフラグのコンテキスト値を決定し、前記第1のCGフラグの前記コンテキスト値を用いてエントロピー復号することによって前記第1のCGフラグを決定するステップと

20

を更に含む、請求項1に記載の方法。

【請求項4】

前記変換係数が前記第2の部分領域に位置するか否かを決定するステップと、

前記変換係数が前記第2の部分領域に位置すると決定された場合、前記変換係数が伝達されず、ゼロであると決定するステップと、

前記変換係数が前記第2の部分領域に位置しないと決定された場合、前記隣接変換係数が前記第2の部分領域に位置するか否かの決定を実行するステップと

を更に含む、請求項1乃至3のうちいずれか1項に記載の方法。

【請求項5】

前記決定された変換係数は、シンタックスエレメントを用いて表現されており、前記シンタックスエレメントは、前記変換係数が非ゼロ変換係数であるか否か、前記変換係数のパリティ、前記変換係数が2よりも大きいか否か、及び前記変換係数が4よりも大きいか否かのうち1つを示す、請求項1乃至4のうちいずれか1項に記載の方法。

30

【請求項6】

ビデオ復号の装置であって、

符号化ビデオビットストリームから変換ブロック(TB)の符号化情報を復号するステップであって、前記符号化情報は、二次変換が適用される前記TBの領域を示し、前記領域は、前記二次変換によって計算された変換係数を有する第1の部分領域と、第2の部分領域とを含む、ステップと、

40

前記TBにおける変換係数について、前記変換係数を決定するために使用される隣接変換係数が前記第2の部分領域に位置するか否かを決定するステップと、

前記隣接変換係数が前記第2の部分領域に位置すると決定された場合、前記隣接変換係数のデフォルト値に従って前記変換係数のコンテキスト値を決定するステップと、

前記TBにおけるサンプルについて、前記変換係数の前記コンテキスト値を用いてエントロピー復号することによって前記変換係数を決定し、前記変換係数に基づいて前記サンプルを復元するステップと

を実行するように構成された処理回路を含む装置。

【請求項7】

前記TBにおける前記変換係数は、第1の係数グループ(CG)内の複数の変換係数のうち1

50

つであり、前記第1のCGについての第1のCGフラグは、前記変換係数のうち少なくとも1つが非ゼロ変換係数であるか否かを示し、

変換係数を含む第2のCGは、前にエントロピー復号されており、前記第1のCGの隣接するCGであり、

前記処理回路は、

前記第2のCGの位置を決定するステップと、

前記第2のCGが前記第2の部分領域に位置すると決定された場合、前記第2のCGについての第2のCGフラグのデフォルト値に基づいて前記第1のCGフラグのコンテキスト値を決定し、前記第1のCGフラグの前記コンテキスト値を用いてエントロピー復号することによって前記第1のCGフラグを決定するステップと

10

を実行するように更に構成される、請求項6に記載の装置。

【請求項8】

前記TBにおける前記変換係数は、第1のCG内の複数の変換係数のうち1つであり、前記第1のCGについての第1のCGフラグは、前記変換係数のうち少なくとも1つが非ゼロ変換係数であるか否かを示し、

第2のCGは、第1の変換係数と第2の変換係数とを含み、前記第2のCGは、前にエントロピー復号されており、前記第1のCGの隣接するCGであり、

前記処理回路は、

前記第2のCGの位置を決定するステップと、

前記第2の変換係数を含む前記第2のCGの一部が前記第2の部分領域に位置し、前記第1の変換係数が非ゼロ変換係数である場合、前記非ゼロ変換係数である前記第1の変換係数に基づいて前記第1のCGフラグのコンテキスト値を決定し、前記第1のCGフラグの前記コンテキスト値を用いてエントロピー復号することによって前記第1のCGフラグを決定するステップと

20

を実行するように更に構成される、請求項6に記載の装置。

【請求項9】

前記処理回路は、

前記変換係数が前記第2の部分領域に位置するか否かを決定するステップと、

前記変換係数が前記第2の部分領域に位置すると決定された場合、前記変換係数が伝達されず、ゼロであると決定するステップと、

30

前記変換係数が前記第2の部分領域に位置しないと決定された場合、前記隣接変換係数が前記第2の部分領域に位置するか否かの決定を実行するステップと

を実行するように更に構成される、請求項6乃至8のうちいずれか1項に記載の装置。

【請求項10】

前記処理回路は、

前記変換係数のシンタックスエレメントを決定するように更に構成され、前記シンタックスエレメントは、前記変換係数が非ゼロ変換係数であるか否か、前記変換係数のパリティ、前記変換係数が2よりも大きいか否か、及び前記変換係数が4よりも大きいか否かのうち1つを示す、請求項6乃至9のうちいずれか1項に記載の装置。

【請求項11】

40

プロセッサに請求項1乃至5のうちいずれか1項に記載の方法を実行させるコンピュータプログラム。

【請求項12】

エンコーダが実行するビデオ符号化の方法であって、

ビデオビットストリームを符号化し、符号化ビデオビットストリームを生成するステップと、

前記符号化ビデオビットストリームから変換ブロック(TB)の符号化情報を復号するステップであって、前記符号化情報は、二次変換が適用される前記TBの領域を示し、前記領域は、前記二次変換によって計算された変換係数を有する第1の部分領域と、第2の部分領域とを含む、ステップと、

50

前記TBにおける変換係数について、前記変換係数を決定するために使用される隣接変換係数が前記第2の部分領域に位置するか否かを決定するステップと、

前記隣接変換係数が前記第2の部分領域に位置すると決定された場合、前記隣接変換係数のデフォルト値に従って前記変換係数のコンテキスト値を決定するステップと、

前記TBにおけるサンプルについて、前記変換係数の前記コンテキスト値を用いてエンコード復号することによって前記変換係数を決定し、前記変換係数に基づいて前記サンプルを復元するステップと

を含む方法。

【発明の詳細な説明】

【技術分野】

10

【0001】

[関連出願への相互参照]

本願は、2020年2月11日に出願された米国特許出願第16/787,628号「Method and apparatus for video coding」の優先権の利益を主張する。当該米国特許出願は、2019年2月12日に出願された米国仮出願第62/804,666号「Transform Coefficient Zero-Out」の優先権の利益を主張する。これらの前の出願の全開示が全て参照によって本明細書に組み込まれる。

【0002】

[技術分野]

本開示は、概してビデオ符号化に関連する実施形態を記載する。

20

【背景技術】

【0003】

本明細書において提供される背景技術の説明は、本開示の背景を一般的に提示するためのものである。本願の発明者の研究は、当該研究がこの背景技術の段落に記載されている範囲において、また、出願時に従来技術として特に適することのない説明の側面も、本開示に対する従来技術として明示的にも暗示的にも認められるものではない。

【0004】

ビデオ符号化及び復号は、動き補償によるインターピクチャ予測を使用して実行できる。非圧縮デジタルビデオは、一連のピクチャを含むことができ、各ピクチャは、例えば、1920×1080の輝度サンプル及び関連する色差サンプルの空間次元を有する。一連のピクチャは、例えば、毎秒60ピクチャ又は60Hzの固定又は可変のピクチャレート(フレームレートとしても非公式に知られている)を有することができる。非圧縮ビデオは、かなりのビットレート要件を有する。例えば、サンプル当たり8ビットの1080p60 4:2:0ビデオ(60Hzのフレームレートの1920×1080の輝度サンプル解像度)は、1.5Gbit/sに近い帯域幅を必要とする。1時間のこのようなビデオは、600Gバイトを超える記憶空間を必要とする。

30

【0005】

ビデオ符号化及び復号の1つの目的は、圧縮を通じて入力ビデオ信号の冗長性を低減できることである。圧縮は、場合によっては2桁以上も上記の帯域幅又は記憶空間の要件を低減するのに役立つことができる。可逆圧縮及び不可逆圧縮の双方並びにこれらの組み合わせを使用することができる。可逆圧縮とは、元の信号の正確なコピーが圧縮された元の信号から復元できる技術を示す。不可逆圧縮を使用する場合、復元された信号は、元の信号と同一ではない可能性があるが、元の信号と復元された信号との間の歪みは、復元された信号を目的のアプリケーションにとって有用にするほど十分に小さい。ビデオの場合、不可逆圧縮が広く使用されている。許容される歪みの量はアプリケーションに依存する。例えば、特定の消費者のストリーミングアプリケーションのユーザは、テレビ配信アプリケーションのユーザよりも高い歪みを許容する可能性がある。達成可能な圧縮比は、より高い許容可能な歪み/許容される歪みがより高い圧縮比をもたらすことができるということを反映できる。

40

【0006】

50

ビデオエンコーダ及びデコーダは、例えば、動き補償、変換、量子化及びエントロピー符号化を含むいくつかの広いカテゴリからの技術を利用することができる。

【0007】

ビデオコーデック技術は、イントラ符号化として知られる技術を含むことができる。イントラ符号化では、サンプル値は、前に復元された参照ピクチャからのサンプル又は他のデータを参照せずに表される。いくつかのビデオコーデックでは、ピクチャは空間的にサンプルのブロックに細分される。サンプルの全てのブロックがイントラモードで符号化される場合、そのピクチャはイントラピクチャとすることができる。イントラピクチャと、独立デコーダリフレッシュピクチャのようなそれらの派生物は、デコーダ状態をリセットするために使用でき、したがって、符号化ビデオビットストリーム及びビデオセッションにおける最初のピクチャとして或いは静止画像として使用できる。イントラブロックのサンプルは変換を受けさせることができ、変換係数はエントロピー符号化の前に量子化できる。イントラ予測は、変換前ドメインにおけるサンプル値を最小化する技術とすることができる。場合によっては、変換後のDC値が小さく、AC係数が小さいほど、エントロピー符号化後のブロックを表すために所与の量子化ステップサイズにおいて必要とされるビットが少なくなる。

10

【0008】

例えば、MPEG-2世代の符号化技術から知られているような従来のイントラ符号化は、イントラ予測を使用しない。しかし、いくつかのより新しいビデオ圧縮技術は、例えば、空間的に隣接しており復号順で前のデータのブロックを符号化/復号する間に取得された周囲のサンプルデータ及び/又はメタデータから試みる技術を含む。このような技術は、以下では「イントラ予測(intra prediction)」技術と呼ばれる。少なくともいくつかの場合、イントラ予測は復元中のカレントピクチャからの参照データのみを使用し、参照ピクチャからの参照データを使用しない点に留意すべきである。

20

【0009】

多くの形式のイントラ予測が存在し得る。所与のビデオ符号化技術においてこのような技術のうち1つ以上が使用できる場合、使用される技術は、イントラ予測モードで符号化できる。或る場合、モードは、サブモード及び/又はパラメータを有することができ、これらは個別に符号化されてもよく、或いは、モードコードワードに含まれてもよい。所与のモード/サブモード/パラメータの組み合わせに使用するコードワードは、イントラ予測を通じた符号化効率利得に影響を与える可能性があり、同様に、コードワードをビットストリームに変換するために使用されるエントロピー符号化技術にも影響を与える可能性がある。

30

【0010】

特定のイントラ予測モードがH.264で導入され、H.265で改良されて、JEM(joint exploration model)、VVC(versatile video coding)及びBMS(benchmark set)のようなより新しい符号化技術で更に改良されている。予測ブロックは、既に利用可能なサンプルに属する隣接するサンプル値を使用して形成できる。隣接サンプルのサンプル値は、方向に従って予測ブロックにコピーされる。使用中の方向への参照は、ビットストリームにおいて符号化でき、或いは、それ自体が予測されてもよい。

40

【0011】

図1Aを参照すると、右下に示されているのは、H.265の33個の可能な予測子方向(35個のイントラモードのうち33個の角度モードに対応する)から知られている9個の予測子方向のサブセットである。矢印が収束する点(101)は、予測されるサンプルを表す。矢印は、サンプルが予測される方向を表す。例えば、矢印(102)は、サンプル(101)が、水平から45度の角度の右上に対する1つ又は複数のサンプルから予測されることを示す。同様に、矢印(103)は、サンプル(101)が、水平から22.5度の角度でサンプル(101)の左下に対する1つ又は複数のサンプルから予測されることを示す。

【0012】

依然として図1Aを参照すると、左上には、4×4のサンプルの正方形ブロック(104)が

50

示されている(破線の太線で示されている)。正方形ブロック(104)は、16個のサンプルを含み、各サンプルは「S」と、Y次元におけるその位置(例えば、行インデックス)と、X次元におけるその位置(例えば、列インデックス)とでそれぞれラベル付けされる。例えば、サンプルS21は、Y次元における第2のサンプル(上から)及びX次元における第1のサンプル(左から)である。同様に、サンプルS44は、Y次元及びX次元の双方においてブロック(104)内の第4のサンプルである。ブロックのサイズが4×4のサンプルであるので、S44は右下にある。さらに、同様の番号付け方式に従った参照サンプルが示されている。参照サンプルは、Rと、ブロック(104)に対するそのY位置(例えば、行インデックス)及びX位置(列インデックス)とでラベル付けされる。H.264及びH.265の双方において、予測サンプルは復元中のブロックに隣接しており、したがって、負の値が使用される必要はない。

10

## 【0013】

イントラピクチャ予測は、伝達された予測方向に応じて、隣接サンプルから参照サンプル値をコピーすることによって機能できる。例えば、符号化ビデオビットストリームが、このブロックについて、矢印(102)と一致する予測方向を示す信号伝達を含むと仮定する。すなわち、サンプルは、水平から45度の角度で右上に対する1つ又は複数の予測サンプルから予測されると仮定する。この場合、サンプルS41、S32、S23及びS14は、同じ参照サンプルR05から予測される。次いで、サンプルS44は、参照サンプルR08から予測される。

## 【0014】

或る場合、特に方向が45度で均一に割り切れない場合、参照サンプルを計算するために、複数の参照サンプルの値が、例えば補間によって組み合わせられてもよい。

20

## 【0015】

ビデオ符号化技術の発達に伴い、可能な方向の数が増加している。H.264(2003年)では、9個の異なる方向が表現可能であった。これは、H.265(2013年)で33個に増加し、開示の時点でのJEM/VVC/BMSでは、最大で65個の方向をサポートできる。最も可能性の高い方向を特定するために実験が行われており、エントロピー符号化における或る技術は、より可能性の低い方向に対して特定のペナルティを受け入れて、少数のビットでこれらの可能性の高い方向を表すために使用されている。さらに、場合によっては、方向自体が、隣接する既に復号されたブロックで使用される隣接方向から予測できる。

## 【0016】

図1Bは、時間と共に増加する予測方向の数を示す、JEMに従った65個のイントラ予測方向を示す概略図(180)を示す。

30

## 【0017】

方向を表す符号化ビデオビットストリームにおけるイントラ予測方向ビットのマッピングは、ビデオ符号化技術によって異なる可能性があり、例えば、予測方向の簡単な直接マッピングから、イントラ予測モード、コードワード、最確モードを含む複雑な適応方式、及び同様の技術まで及び可能性がある。しかし、全ての場合で、ビデオコンテンツにおいて、特定の他の方向よりも統計的に生じにくい特定の方向が存在し得る。ビデオ圧縮の目標は冗長性の低減であるので、良好に機能するビデオ符号化技術において、これらのより可能性の低い方向は、より可能性の高い方向よりもより多くのビット数によって表される。

40

## 【0018】

ビデオ符号化及び復号は、動き補償を用いたインターピクチャ予測を使用して実行できる。動き補償は不可逆圧縮技術であり、前に復元されたピクチャ又はその一部(参照ピクチャ)からのサンプルデータのブロックが、動きベクトル(以下、MVという)によって示される方向に空間的にシフトされた後に、新たに復元されるピクチャ又はその一部の予測に使用されるという技術に関連付けることができる。場合によっては、参照ピクチャは現在復元中のピクチャと同じものにすることができる。MVは、X及びYの2次元を有してもよく、或いは、3次元を有してもよく、第3の次元は、使用中の参照ピクチャを示す(後者は、間接的に、時間次元とすることができる)。

## 【0019】

50

いくつかのビデオ圧縮技術では、サンプルデータの特定の領域に適用可能なMVは、他のMVから予測でき、例えば、復元中の領域に空間的に隣接しており、復号順でそのMVに先行するサンプルデータの他の領域に関連するMVから予測できる。これにより、MVを符号化するために必要なデータ量をかなり低減でき、それによって冗長性を除去し、圧縮を増加させることができる。例えば、カメラから導出された入力ビデオ信号(ナチュラルビデオとして知られている)を符号化する場合、単一のMVが適用可能な領域よりも大きい領域が同様の方向に移動し、したがって、場合によっては隣接領域のMVから導出された同様の動きベクトルを使用して予測できるという統計的な可能性が存在するので、MV予測は効果的に機能し得る。その結果、所与の領域に対して検出されたMVは、周囲のMVから予測されるMVと同様又は同一であることになり、そのMVは、エントロピー符号化の後に、

10

【0020】

H.265/HEVC(ITU-T Rec. H.265, 「High Efficiency Video Coding」, December 2016)には、様々なMV予測メカニズムが記載されている。H.265が提供する多くのMV予測メカニズムの中で、本明細書において「空間マージ(spatial merge)」と呼ばれる技術が存在する。

【0021】

20

図2を参照すると、カレントブロック(201)は、動き探索処理中に、空間的にシフトされた同じサイズの前のブロックから予測可能であることがエンコーダによって検出されたサンプルを含む。MVを直接符号化する代わりに、MVは、1つ以上の参照ピクチャに関連するメタデータから導出でき、例えば、A0、A1及びB0、B1、B2(それぞれ202~206)と示される5個の周囲のサンプルのうちいずれか1つに関連するMVを使用して、(復号順で)最近の参照ピクチャから導出できる。H.265では、MV予測は、隣接ブロックが使用しているものと同じ参照ピクチャからの予測子を使用できる。

【発明の概要】

【0022】

30

本開示の態様は、ビデオ符号化/復号のための方法及び装置を提供する。いくつかの例では、ビデオ復号の装置は、処理回路を含む。処理回路は、符号化ビデオビットストリームから変換ブロック(TB, transform block)の符号化情報を復号できる。符号化情報は、二次変換が適用されるTBの領域を示すことができる。当該領域は、二次変換によって計算された変換係数を有する第1の部分領域と、第2の部分領域とを含むことができる。処理回路は、TBにおける変換係数について、変換係数を決定するために使用される隣接変換係数が第2の部分領域に位置するか否かを決定することができる。隣接変換係数が第2の部分領域に位置すると決定された場合、処理回路は、隣接変換係数のデフォルト値に従って変換係数を決定できる。処理回路は、TBにおけるサンプルについて、変換係数に基づいて復元できる。

【0023】

40

一実施形態では、TBにおける変換係数は、第1の係数グループ(CG, coefficient group)内の複数の変換係数のうち1つである。第1のCGについての第1のCGフラグは、変換係数のうち少なくとも1つが非ゼロ変換係数であるか否かを示すことができる。変換係数を含む第2のCGは、前にエントロピー復号されており、第1のCGの隣接するCGである。処理回路は、第2のCGの位置を決定できる。第2のCGが第2の部分領域に位置すると決定された場合、処理回路は、第2のCGについての第2のCGフラグのデフォルト値に基づいて第1のCGフラグを決定できる。

【0024】

一実施形態では、TBにおける変換係数は、第1のCG内の複数の変換係数のうち1つである。第1のCGについての第1のCGフラグは、変換係数のうち少なくとも1つが非ゼロ変換

50

係数であるか否かを示すことができる。第2のCGは、第1の変換係数と第2の変換係数とを含む。第2のCGは、前にエントロピー復号されており、第1のCGの隣接するCGである。処理回路は、第2のCGの位置を決定できる。第2の変換係数を含む第2のCGの一部が第2の部分領域に位置し、第1の変換係数が非ゼロ変換係数である場合、処理回路は、第2のCGについての第2のCGフラグに基づいて第1のCGフラグを決定できる。

【0025】

一実施形態では、処理回路は、変換係数が第2の部分領域に位置するか否かを決定できる。変換係数が第2の部分領域に位置すると決定された場合、処理回路は、変換係数が伝達されず、ゼロであると決定できる。変換係数が第2の部分領域に位置しないと決定された場合、処理回路は、隣接変換係数が第2の部分領域に位置するか否かの決定を実行できる。

10

【0026】

一実施形態では、処理回路は、変換係数のシンタックスエレメントを決定できる。シンタックスエレメントは、変換係数が非ゼロ変換係数であるか否か、変換係数のパリティ、変換係数が2よりも大きいか否か、及び変換係数が4よりも大きいか否かのうち1つを示すことができる。

【0027】

本開示の態様は、ビデオ復号の方法及び装置を提供する。いくつかの例では、ビデオ復号の装置は、処理回路を含む。処理回路は、符号化ビデオビットストリームから変換ブロック(TB)の符号化情報を復号できる。処理回路は、符号化情報に基づいて、二次変換がTBの第1の領域に対して実行されるか否かを決定できる。第1の領域は、二次変換によって計算された変換係数を有する第1の部分領域と、第2の部分領域とを含むことができる。二次変換が実行されると決定された場合、処理回路は、TBにおける第2の領域内の変換係数がゼロであると決定でき、第2の領域は第1の領域の外側である。

20

【0028】

一例では、TBにおける複数の変換係数を含む係数ユニットのサイズ及び位置は、第1の領域に基づいて決定され、係数ユニットの外側の変換係数はゼロである。

【0029】

一例では、第1の領域はTBにおける左上の8×8の領域であり、係数ユニットは第1の領域であり、第2の領域は左上の8×8の領域に隣接する。

30

【0030】

一例では、第1の部分領域はTBにおける左上の4×4の領域であり、係数ユニットは第1の領域内の第1の部分領域であり、第2の領域と第2の部分領域とを含む結合領域内の変換係数はゼロである。

【0031】

一例では、第1の領域はTBにおける左上の4×4の領域であり、係数ユニットは第1の領域であり、第2の領域は左上の4×4の領域に隣接する。

【0032】

また、本開示の態様は、ビデオ復号のためにコンピュータによって実行されると、コンピュータにビデオ復号の方法を実行させる命令を記憶した非一時的なコンピュータ読み取り可能媒体を提供する。

40

【図面の簡単な説明】

【0033】

開示の対象物の更なる特徴、性質及び様々な利点は、以下の詳細な説明及び添付の図面からより明らかになる。

【図1A】イントラ予測モードの例示的なサブセットの概略図である。

【図1B】例示的なイントラ予測方向の図である。

【図2】一例におけるカレントブロック及びその周囲の空間マージ候補の概略図である。

【図3】一実施形態による通信システム(300)の簡略化したブロック図の概略図である。

【図4】一実施形態による通信システム(400)の簡略化したブロック図の概略図である。

50

- 【図5】一実施形態によるデコーダの簡略化したブロック図の概略図である。  
 【図6】一実施形態によるエンコーダの簡略化したブロック図の概略図である。  
 【図7】他の実施形態によるエンコーダのブロック図を示す。  
 【図8】他の実施形態によるデコーダのブロック図を示す。  
 【図9】変換ユニットのシンタックスの例を示す。  
 【図10A】残差符号化のシンタックスの例を示す。  
 【図10B】残差符号化のシンタックスの例を示す。  
 【図11】サブブロック走査プロセス(1100)の例を示す。  
 【図12】現在の係数(1220)についてコンテキスト選択に使用されるローカルテンプレート(1230)の例を示す。

10

- 【図13】残差符号化のシンタックスの例を示す。  
 【図14A】サブブロック変換(SBT, sub-block transform)の例を示す。  
 【図14B】サブブロック変換(SBT, sub-block transform)の例を示す。  
 【図14C】サブブロック変換(SBT, sub-block transform)の例を示す。  
 【図14D】サブブロック変換(SBT, sub-block transform)の例を示す。  
 【図15A】SBTが使用される場合のビデオ符号化標準の仕様テキストへの変更を示す。  
 【図15B】SBTが使用される場合のビデオ符号化標準の仕様テキストへの変更を示す。  
 【図15C】SBTが使用される場合のビデオ符号化標準の仕様テキストへの変更を示す。  
 【図15D】SBTが使用される場合のビデオ符号化標準の仕様テキストへの変更を示す。  
 【図15E】SBTが使用される場合のビデオ符号化標準の仕様テキストへの変更を示す。  
 【図15F】SBTが使用される場合のビデオ符号化標準の仕様テキストへの変更を示す。

20

- 【図16A】本開示の実施形態による係数ユニットの例を示す。  
 【図16B】本開示の実施形態による係数ユニットの例を示す。  
 【図16C】本開示の実施形態による係数ユニットの例を示す。  
 【図16D】本開示の実施形態による係数ユニットの例を示す。  
 【図17】例示的な係数ブロック(1710)を示す。  
 【図18】ゼロアウト方法を含む例示的な二次変換を示す。  
 【図19】ゼロアウト方法を含む例示的な二次変換を示す。  
 【図20】本開示の一実施形態によるプロセス(2000)の概略を示すフローチャートを示す。

30

- 【図21】本開示の一実施形態によるプロセス(2100)の概略を示すフローチャートを示す。  
 【図22】一実施形態によるコンピュータシステムの概略図である。

【発明を実施するための形態】

【0034】

図3は、本開示の一実施形態による通信システム(300)の簡略化したブロック図を示す。通信システム(300)は、例えば、ネットワーク(350)を介して互いに通信できる複数の端末デバイスを含む。例えば、通信システム(300)は、ネットワーク(350)を介して相互接続された第1の対の端末デバイス(310)及び(320)を含む。図3の例では、第1の対の端末デバイス(310)及び(320)は、データの一方方向伝送を実行する。例えば、端末デバイス(310)は、ネットワーク(350)を介して他の端末デバイス(320)に送信するために、ビデオデータ(例えば、端末デバイス(310)によってキャプチャされたビデオピクチャのストリーム)を符号化してもよい。符号化されたビデオデータは、1つ以上の符号化ビデオビットストリームの形式で送信されてもよい。端末デバイス(320)は、ネットワーク(350)から符号化ビデオデータを受信し、符号化ビデオデータを復号して、ビデオピクチャを復元して復元されたビデオデータに従ってビデオピクチャを表示してもよい。一方方向データ伝送は、メディア提供アプリケーション等において一般的でもよい。

40

【0035】

他の例では、通信システム(300)は、例えば、テレビ会議中に発生し得る符号化ビデオデータの双方向伝送を実行する第2の対の端末デバイス(330)及び(340)を含む。データの

50

双方向伝送のために、一例では、端末デバイス(330)及び(340)の各端末デバイスは、ネットワーク(350)を介して端末デバイス(330)及び(340)の他方の端末デバイスに送信するために、ビデオデータ(例えば、端末デバイスによってキャプチャされたビデオピクチャのストリーム)を符号化してもよい。また、端末デバイス(330)及び(340)の各端末デバイスは、端末デバイス(330)及び(340)の他方の端末デバイスによって送信された符号化ビデオデータを受信してもよく、符号化ビデオデータを復号してビデオピクチャを復元してもよく、復元されたビデオデータに従って、アクセス可能な表示デバイスにビデオピクチャを表示してもよい。

#### 【0036】

図3の例では、端末デバイス(310)、(320)、(330)及び(340)は、サーバ、パーソナルコンピュータ及びスマートフォンとして示されることがあるが、本開示の原理はこれらに限定されない。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、メディアプレイヤ及び/又は専用のテレビ会議機器に適用がある。ネットワーク(350)は、例えば、有線(配線接続)及び/又は無線通信ネットワークを含む、端末デバイス(310)、(320)、(330)及び(340)の間で符号化ビデオデータを伝達するいずれかの数のネットワークを表す。通信ネットワーク(350)は、回線交換チャンネル及び/又はパケット交換チャンネルにおいてデータを交換してもよい。代表的なネットワークは、電気通信ネットワーク、ローカルエリアネットワーク、広域ネットワーク及び/又はインターネットを含む。本説明の目的では、ネットワーク(350)のアーキテクチャ及びトポロジは、本明細書において以下に説明しない限り、本開示の動作には重要ではない。

#### 【0037】

図4は、開示の対象物のアプリケーションの例として、ストリーミング環境におけるビデオエンコーダ及びビデオデコーダの配置を示す。開示の対象物は、例えば、テレビ会議、デジタルTV、デジタルメディア(CD、DVD、メモリスティック等を含む)上の圧縮ビデオの記憶等を含む、他のビデオ可能なアプリケーションにも同様に適用可能である。

#### 【0038】

ストリーミングシステムはキャプチャサブシステム(413)を含んでもよく、当該キャプチャサブシステム(413)は、例えば、非圧縮のビデオピクチャのストリーム(402)を生成するビデオソース(401)(例えば、デジタルカメラ)を含んでもよい。一例では、ビデオピクチャのストリーム(402)は、デジタルカメラによって撮影されたサンプルを含む。符号化ビデオデータ(404)(又は符号化ビデオビットストリーム)と比較したときに高いデータ量であることを強調する太線として描かれるビデオピクチャのストリーム(402)は、ビデオソース(401)に結合されたビデオエンコーダ(403)を含む電子デバイス(420)によって処理されてもよい。ビデオエンコーダ(403)は、以下により詳細に説明するように、開示の対象物の態様を可能にするため或いは実装するために、ハードウェア、ソフトウェア又はこれらの組み合わせを含んでもよい。ビデオピクチャのストリーム(402)と比較したときにより低いデータ量であることを強調するために細線として描かれる符号化ビデオデータ(404)(又は符号化ビデオビットストリーム(404))は、将来の使用のためにストリーミングサーバ(405)に記憶されてもよい。図4におけるクライアントサブシステム(406)及び(408)のような1つ以上のストリーミングクライアントサブシステムは、ストリーミングサーバ(405)にアクセスして符号化ビデオデータ(404)のコピー(407)及び(409)を取得してもよい。クライアントサブシステム(406)は、例えば、電子デバイス(430)内にビデオデコーダ(410)を含んでもよい。ビデオデコーダ(410)は、符号化ビデオデータの入力コピー(407)を復号し、ディスプレイ(412)(例えば、表示画面)又は他のレンダリングデバイス(図示せず)上にレンダリングできるビデオピクチャの出力ストリーム(411)を生成する。いくつかのストリーミングシステムでは、符号化ビデオデータ(404)、(407)及び(409)(例えば、ビデオビットストリーム)は、特定のビデオ符号化/圧縮標準に従って符号化されてもよい。これらの標準の例は、ITU-T勧告H.265を含む。一例では、開発中のビデオ符号化標準は、VVC(Versatile Video Coding)として非公式に知られている。開示の対象物は、VVCの背景において使用されてもよい。

10

20

30

40

50

## 【 0 0 3 9 】

電子デバイス(420)及び(430)は、他の構成要素(図示せず)を含んでもよい点に留意すべきである。例えば、電子デバイス(420)は、ビデオデコーダ(図示せず)を含んでもよく、また、電子デバイス(430)は、ビデオエンコーダ(図示せず)を含んでもよい。

## 【 0 0 4 0 】

図5は、本開示の一実施形態によるビデオデコーダ(510)のブロック図を示す。ビデオデコーダ(510)は、電子デバイス(530)に含まれてもよい。電子デバイス(530)は、受信機(531)(例えば、受信回路)を含んでもよい。図4の例におけるビデオデコーダ(410)の代わりにビデオデコーダ(510)が使用されてもよい。

## 【 0 0 4 1 】

受信機(531)は、ビデオデコーダ(510)によって復号されるべき1つ以上の符号化ビデオシーケンスを受信してもよく、同一又は他の実施形態では、一度に1つの符号化ビデオシーケンスを受信してもよく、各符号化ビデオシーケンスの復号は、他の符号化ビデオシーケンスとは独立している。符号化ビデオシーケンスは、チャンネル(501)から受信されてもよく、当該チャンネルは、符号化ビデオデータを記憶する記憶デバイスへのハードウェア/ソフトウェアリンクでもよい。受信機(531)は、符号化ビデオデータを、他のデータ(例えば、符号化オーディオデータ及び/又は補助データストリーム)と共に受信してもよく、これらは、それぞれの使用エンティティ(図示せず)に転送されてもよい。受信機(531)は、符号化ビデオシーケンスを他のデータから分離してもよい。ネットワークジッタを防止するために、バッファメモリ(515)は、受信機(531)とエントロピーデコーダ/パーサ(520)(以下、「パーサ(520)」という)との間に結合されてもよい。特定のアプリケーションでは、バッファメモリ(515)はビデオデコーダ(510)の一部である。他の場合には、ビデオデコーダ(510)の外側にあってもよい(図示せず)。更に他の場合には、例えば、ネットワークジッタを防止するために、ビデオデコーダ(510)の外側にバッファメモリ(図示せず)が存在してもよく、加えて、例えば、再生タイミングに対処するために、ビデオデコーダ(510)の内側に他のバッファメモリ(515)が存在してもよい。受信機(531)が、十分な帯域幅及び制御可能性を有する記憶/転送デバイスから、或いは、アイソクロナスネットワークからデータを受信している場合、バッファメモリ(515)は必要なくともよく或いは小さくすることができる。インターネットのようなベストエフォート型パケットネットワークでの使用については、バッファメモリ(515)が必要とされてもよく、比較的大きくすることができ、有利には適応的なサイズとすることができ、ビデオデコーダ(510)の外側のオペレーティングシステム又は同様の要素(図示せず)に少なくとも部分的に実装されてもよい。

## 【 0 0 4 2 】

ビデオデコーダ(510)は、符号化ビデオシーケンスからシンボル(521)を復元するためのパーサ(520)を含んでもよい。これらのシンボルのカテゴリは、ビデオデコーダ(510)の動作を管理するために使用される情報を含み、レンダリングデバイス(512)(例えば、表示画面)のようなレンダリングデバイスを制御するための情報を潜在的に含む。当該レンダリングデバイス(512)は、図5に示されているように、電子デバイス(530)の一体的な部分ではないが、電子デバイス(530)に結合されてもよい。レンダリングデバイスの制御情報は、補足エンハンスメント情報(SEI, Supplemental Enhancement Information)(SEIメッセージ)又はビデオユーザビリティ情報(VUI, Video Usability Information)パラメータセットフラグメント(図示せず)の形式でもよい。パーサ(520)は、受信した符号化ビデオシーケンスを解析/エントロピー復号してもよい。符号化ビデオシーケンスの符号化は、ビデオ符号化技術又は標準に従ってもよく、可変長符号化、ハフマン符号化、コンテキスト感度を伴う或いは伴わない算術符号化等を含む様々な原理に従ってもよい。パーサ(520)は、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、符号化ビデオシーケンスから、ビデオデコーダ内の画素のサブグループのうち少なくとも1つについてのサブグループパラメータのセットを抽出してもよい。サブグループは、グループオブピクチャ(GOP, Group of Picture)、ピクチャ、タイル、スライス、マクロブロック、符号化ユニット(CU, Coding Unit)、ブロック、変換ユニット(TU, Transformation U

10

20

30

40

50

nit)、予測ユニット(PU, Prediction Unit)等を含んでもよい。また、パーサ(520)は、符号化ビデオシーケンスから、変換係数、量子化パラメータ値、動きベクトル等のような情報を抽出してもよい。

【0043】

パーサ(520)は、シンボル(521)を生成するために、バッファメモリ(515)から受信したビデオシーケンスに対してエントロピー復号/解析動作を実行してもよい。

【0044】

シンボル(521)の復元には、符号化ビデオピクチャ又はその部分のタイプ(例えば、インターピクチャ及びイントラピクチャ、インターブロック及びイントラブロック)及び他の要因に依存して、複数の異なるユニットが関与してもよい。どのユニットがどのように関与するかは、パーサ(520)によって符号化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てもよい。パーサ(520)と以下の複数ユニットとの間のこのようなサブグループ制御情報の流れは、明確にするために図示されていない。

10

【0045】

上記の機能ブロックの他に、ビデオデコーダ(510)は、概念的に、以下に説明するような複数の機能ユニットに細分されてもよい。商用的な制約の下で動作する実用的な実装では、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互いに統合されてもよい。しかし、開示の対象物を説明する目的で、以下の機能ユニットに概念的に細分することが適切である。

【0046】

第1のユニットは、スケーラ/逆変換ユニット(551)である。スケーラ/逆変換ユニット(551)は、パーサ(520)からシンボル(521)として、制御情報(どの変換を使用すべきか、ブロックサイズ、量子化係数、量子化スケーリング行列等を含む)と共に、量子化された変換係数を受信する。スケーラ/逆変換ユニット(551)は、アグリゲータ(555)に入力できるサンプル値を含むブロックを出力してもよい。

20

【0047】

場合によっては、スケーラ/逆変換(551)の出力サンプルは、イントラ符号化ブロックに関連してもよく、すなわち、前に復元されたピクチャからの予測情報を使用していないが、カレントピクチャの前に復元された部分からの予測情報を使用できるブロックに関連してもよい。このような予測情報は、イントラピクチャ予測ユニット(552)によって提供されてもよい。場合によっては、イントラピクチャ予測ユニット(552)は、カレントピクチャバッファ(558)から取り出された周囲の既に復元された情報を使用して、復元中のブロックの同じサイズ及び形状のブロックを生成する。カレントピクチャバッファ(558)は、例えば、部分的に復元されたカレントピクチャ及び/又は完全に復元されたカレントピクチャをバッファする。場合によっては、アグリゲータ(555)は、サンプル毎に、イントラ予測ユニット(552)が生成した予測情報を、スケーラ/逆変換ユニット(551)によって提供された出力サンプル情報に追加する。

30

【0048】

他の場合には、スケーラ/逆変換ユニット(551)の出力サンプルは、インター符号化されて潜在的に動き補償されたブロックに関連してもよい。このような場合、動き補償予測ユニット(553)は、参照ピクチャメモリ(557)にアクセスして、予測に使用されるサンプルを取り出してもよい。ブロックに関連するシンボル(521)に従って、取り出されたサンプルを動き補償した後に、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(555)によってスケーラ/逆変換ユニット(551)の出力(この場合には、残差サンプル又は残差信号と呼ばれる)に追加されてもよい。動き補償予測ユニット(553)に利用可能な、動き補償予測ユニット(553)が予測サンプルを取り出す参照ピクチャメモリ(557)内のアドレスは、例えば、X、Y及び参照ピクチャ成分を有することができるシンボル(521)の形式で、動きベクトルによって制御されてもよい。また、動き補償は、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(557)から取り出されるサンプル値の補間、動きベクトル予測メカニズム等を含んでもよい。

40

50

## 【 0 0 4 9 】

アグリゲータ(555)の出力サンプルは、ループフィルタユニット(556)内の様々なループフィルタリング技術を受けてもよい。ビデオ圧縮技術はループ内フィルタ技術を含んでもよく、当該ループ内フィルタ技術は、符号化ビデオシーケンス(符号化ビデオビットストリームとも呼ばれる)に含まれるパラメータによって制御され、パーサ(520)からシンボル(521)としてループフィルタユニット(556)に利用可能にされるが、符号化ピクチャ又は符号化ビデオシーケンスの(復号順に)前の部分の復号の間に取得されたメタ情報に応答すると共に、前に復元されてループフィルタリングされたサンプル値にも応答してもよい。

## 【 0 0 5 0 】

ループフィルタユニット(556)の出力はサンプルストリームでもよく、当該サンプルストリームは、レンダリングデバイス(512)に出力されると共に、将来のインターピクチャ予測に使用するために参照ピクチャメモリ(557)に記憶されてもよい。

10

## 【 0 0 5 1 】

特定の符号化ピクチャは、完全に復元されると、将来の予測のための参照ピクチャとして使用されてもよい。例えば、カレントピクチャに対応する符号化ピクチャが完全に復元され、符号化ピクチャが(例えば、パーサ(520)によって)参照ピクチャとして識別されると、カレントピクチャバッファ(558)は参照ピクチャメモリ(557)の一部となってもよく、新たなカレントピクチャバッファが、後続の符号化ピクチャの復元を開始する前に再割り当てされてもよい。

## 【 0 0 5 2 】

ビデオデコーダ(510)は、ITU-T Rec. H.265のような標準における所定のビデオ圧縮技術に従って復号動作を実行してもよい。符号化ビデオシーケンスがビデオ圧縮技術又は標準のシンタックス及びビデオ圧縮技術又は標準に文書化されているプロファイルの双方に従うという意味で、符号化ビデオシーケンスは、使用されているビデオ圧縮技術又は標準によって指定されたシンタックスに適合してもよい。具体的には、プロファイルは、ビデオ圧縮技術又は標準で利用可能な全てのツールから特定のツールを、そのプロファイルで使用するのに利用可能な唯一のツールとして選択してもよい。また、コンプライアンスのために必要なことは、符号化ビデオシーケンスの複雑さが、ビデオ圧縮技術又は標準のレベルによって定義される範囲内にあることである。場合によっては、レベルは、最大ピクチャサイズ、最大フレームレート、最大復元サンプルレート(例えば、毎秒当たりのメガサンプル単位で測定される)、最大参照ピクチャサイズ等を制限する。場合によっては、レベルによって設定される制限は、仮想参照デコーダ(HRD, Hypothetical Reference Decoder)仕様及び符号化ビデオシーケンスで伝達されるHRDバッファ管理についてのメタデータを通じて更に制限されてもよい。

20

## 【 0 0 5 3 】

一実施形態では、受信機(531)は、符号化ビデオと共に更なる(冗長な)データを受信してもよい。更なるデータは、符号化ビデオシーケンスの一部として含まれてもよい。更なるデータは、データを適切に復号するために、及び/又は元のビデオデータをより正確に復元するために、ビデオデコーダ(510)によって使用されてもよい。更なるデータは、例えば、時間、空間又は信号雑音比(SNR, signal noise ratio)エンハンスメント層、冗長スライス、冗長ピクチャ、前方誤り訂正コード等の形式でもよい。

30

40

## 【 0 0 5 4 】

図6は、本開示の一実施形態によるビデオエンコーダ(603)のブロック図を示す。ビデオエンコーダ(603)は、電子デバイス(620)に含まれる。電子デバイス(620)は、送信機(640)(例えば、送信回路)を含む。図4の例におけるビデオエンコーダ(403)の代わりにビデオエンコーダ(603)が使用されてもよい。

## 【 0 0 5 5 】

ビデオエンコーダ(603)は、ビデオソース(601)(図6の例では電子デバイス(620)の一部ではない)からビデオサンプルを受信してもよく、当該ビデオソース(601)は、ビデオエンコーダ(603)によって符号化されるべきビデオ画像をキャプチャしてもよい。他の例で

50

は、ビデオソース(601)は電子デバイス(620)の一部である。

【0056】

ビデオソース(601)は、デジタルビデオサンプルストリームの形式でビデオエンコーダ(603)によって符号化されるべきソースビデオシーケンスを提供してもよく、当該デジタルビデオサンプルストリームは、いずれかの適切なビット深度(例えば、8ビット、10ビット、12ビット等)、いずれかの色空間(例えば、BT.601 Y CrCb、RGB等)及びいずれかの適切なサンプリング構造(例えば、Y CrCb 4:2:0、Y CrCb 4:4:4)でもよい。メディア提供システムにおいて、ビデオソース(601)は、事前に準備されたビデオを記憶する記憶デバイスでもよい。テレビ会議システムでは、ビデオソース(601)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もよい。ビデオデータは、順に見たときに動きを伝える複数の個々のピクチャとして提供されてもよい。ピクチャ自体は、画素の空間配列として構成されてもよく、各画素は、使用中のサンプリング構造、色空間等に依存して、1つ以上のサンプルを含んでもよい。当業者は、画素とサンプルとの関係を容易に理解することができる。以下の説明は、サンプルに焦点を当てる。

10

【0057】

一実施形態によれば、ビデオエンコーダ(603)は、リアルタイムで或いはアプリケーションによって要求されるいずれかの他の時間制約下で、ソースビデオシーケンスのピクチャを、符号化ビデオシーケンス(643)に符号化及び圧縮してもよい。適切な符号化速度を実現することは、コントローラ(650)の1つの機能である。いくつかの実施形態では、コントローラ(650)は、以下に説明するように、他の機能ユニットを制御し、他の機能ユニットに機能的に結合される。結合は、明確にするために図示されていない。コントローラ(650)によって設定されるパラメータは、レート制御関連パラメータ(ピクチャスキップ、量子化、レート歪み最適化技術のラムダ値等)、ピクチャサイズ、グループオブピクチャ(GOP)のレイアウト、最大動きベクトル探索範囲等を含んでもよい。コントローラ(650)は、特定のシステム設計のために最適化されたビデオエンコーダ(603)に関連する他の適切な機能を有するように構成されてもよい。

20

【0058】

いくつかの実施形態では、ビデオエンコーダ(603)は、符号化ループで動作するように構成される。非常に簡略化した説明として、一例では、符号化ループは、ソースコーダ(630)(例えば、符号化されるべき入力ピクチャ及び参照ピクチャに基づいて、シンボルストリームのようなシンボルを生成することを担う)と、ビデオエンコーダ(603)に埋め込まれた(ローカル)デコーダ(633)とを含んでもよい。デコーダ(633)は、(リモート)デコーダが生成するのと同様に(シンボルと符号化ビデオビットストリームとの間のいずれかの圧縮が、開示の対象物において検討されるビデオ圧縮技術において可逆であるように)、サンプルデータを生成するようにシンボルを復元する。復元されたサンプルストリーム(サンプルデータ)は、参照ピクチャメモリ(634)に入力される。シンボルストリームの復号は、デコーダの位置(ローカル又はリモート)と独立したビット単位の正確な結果をもたらすので、参照ピクチャメモリ(634)内の内容も、ローカルエンコーダとリモートエンコーダとの間でビット単位で正確である。言い換えると、エンコーダの予測部分は、デコーダが復号中に予測を使用するときに「見る」のと全く同じサンプル値を参照ピクチャサンプルとして「見る」。参照ピクチャの同期(例えば、チャンネルエラーの理由で同期が維持できない場合の結果として生じるドリフトを含む)のこの基本原理は、いくつかの関連技術においても同様に使用される。

30

40

【0059】

「ローカル」デコーダ(633)の動作は、ビデオデコーダ(410)のような「リモート」デコーダと同じでもよく、これは、図5に関連して上記において既に詳細に説明した。しかし、図5を簡単に参照すると、シンボルが利用可能であり、エン트로ピーコーダ(645)及びパーサ(520)による符号化ビデオシーケンスへのシンボルの符号化/復号が可逆になり得るので、バッファメモリ(515)及びパーサ(520)を含むビデオデコーダ(510)のエン트로ピー復号部分は、ローカルデコーダ(633)に完全には実装されなくてもよい。

50

## 【0060】

この時点で行うことができる考察は、デコーダ内に存在する解析/エントロピー復号を除く如何なるデコーダ技術も、必然的に対応するエンコーダ内に実質的に同一の機能形式で存在する必要があることである。このため、開示の対象物はデコーダ動作に焦点を当てる。エンコーダ技術の説明は、包括的に記載されるデコーダ技術の逆であるので、省略できる。特定の領域においてのみ、より詳細な説明が必要であり、以下に提供される。

## 【0061】

いくつかの例では、動作中に、ソースコーダ(630)は、動き補償予測符号化を実行してもよく、当該動き補償予測符号化は、「参照ピクチャ」として指定されたビデオシーケンスからの1つ以上の前に符号化されたピクチャを参照して入力ピクチャを予測的に符号化する。このように、符号化エンジン(632)は、入力ピクチャの画素ブロックと、入力ピクチャに対する予測参照として選択され得る参照ピクチャの画素ブロックとの間の差を符号化する。

10

## 【0062】

ローカルビデオデコーダ(633)は、ソースコーダ(630)によって生成されたシンボルに基づいて、参照ピクチャとして指定され得るピクチャの符号化ビデオデータを復号してもよい。符号化エンジン(632)の動作は、有利には、不可逆処理でもよい。符号化ビデオデータがビデオデコーダ(図6に図示せず)で復号され得る場合、復元されたビデオシーケンスは、典型的には、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になり得る。ローカルビデオデコーダ(633)は、参照ピクチャに対してビデオデコーダによって実行され得る復号処理を複製し、復元された参照ピクチャを参照ピクチャキャッシュ(634)に記憶させてもよい。このように、ビデオエンコーダ(603)は、遠端のビデオデコーダによって取得される(送信エラーのない)復元された参照ピクチャとして、共通の内容を有する復元された参照ピクチャのコピーをローカルに記憶してもよい。

20

## 【0063】

予測器(635)は、符号化エンジン(632)のための予測探索を実行してもよい。すなわち、符号化されるべき新たなピクチャについて、予測器(635)は、(候補参照画素ブロックとしての)サンプルデータ又は特定のメタデータ(参照ピクチャ動きベクトル、ブロック形状等)を求めて参照ピクチャメモリ(634)を検索してもよい。これらは、新たなピクチャについての適切な予測参照として機能してもよい。予測器(635)は、適切な予測参照を検出するために、サンプルブロック毎画素ブロック毎(sample block-by-pixel block)に動作してもよい。場合によっては、予測器(635)によって取得された検索結果によって決定された入力ピクチャは、参照ピクチャメモリ(634)に記憶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てもよい。

30

## 【0064】

コントローラ(650)は、例えば、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及びサブグループパラメータの設定を含む、ソースコーダ(630)の符号化動作を管理してもよい。

## 【0065】

全ての上記の機能ユニットの出力は、エントロピーコーダ(645)におけるエントロピー符号化を受けてもよい。エントロピーコーダ(645)は、ハフマン符号化、可変長符号化、算術符号化等のような技術に従って、シンボルを可逆圧縮することによって、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ビデオシーケンスに変換する。

40

## 【0066】

送信機(640)は、エントロピーコーダ(645)によって生成された符号化ビデオシーケンスをバッファして、通信チャネル(660)を介した送信の準備をしてもよく、当該通信チャネル(660)は、符号化ビデオデータを記憶する記憶デバイスへのハードウェア/ソフトウェアリンクでもよい。送信機(640)は、ビデオコーダ(603)からの符号化ビデオデータを、送信されるべき他のデータ(例えば、符号化オーディオデータ及び/又は補助データストリーム(図示せず))とマージしてもよい。

50

## 【0067】

コントローラ(650)は、ビデオエンコーダ(603)の動作を管理してもよい。符号化中に、コントローラ(650)は、各符号化ピクチャに、特定の符号化ピクチャタイプを割り当ててもよい。当該符号化ピクチャタイプは、各ピクチャに適用され得る符号化技術に影響を与えてもよい。例えば、ピクチャは、しばしば、以下のピクチャタイプのうち1つとして割り当てられてもよい。

## 【0068】

イントラピクチャ(Iピクチャ)は、予測のソースとしてシーケンス内の他のピクチャを使用せずに、符号化及び復号され得るものでもよい。いくつかのビデオコーデックは、例えば、独立デコーダリフレッシュ(IDR, Independent Decoder Refresh)ピクチャを含む、異なるタイプのイントラピクチャを許容する。当業者は、Iピクチャのこれらの変形例と、それぞれの用途及び特徴を認識する。

10

## 【0069】

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つの動きベクトル及び参照インデックスを使用して、イントラ予測又はインター予測を使用して符号化及び復号され得るものでもよい。

## 【0070】

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つの動きベクトル及び参照インデックスを使用して、イントラ予測又はインター予測を使用して符号化及び復号され得るものでもよい。同様に、複数の予測ピクチャは、単一のブロックの復元のために、2つより多くの参照ピクチャ及び関連するメタデータを使用してもよい。

20

## 【0071】

一般的に、ソースピクチャは、空間的に複数のサンプルブロック(例えば、それぞれ $4 \times 4$ 、 $8 \times 8$ 、 $4 \times 8$ 又は $16 \times 16$ のサンプルのブロック)に細分され、ブロック毎に符号化されてもよい。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る通り、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てもよい。例えば、Iピクチャのブロックは、非予測的に符号化されてもよく、或いは、同じピクチャ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てもよい(空間予測又はイントラ予測)。Pピクチャの画素ブロックは、1つ前に符号化された参照ピクチャを参照して、空間予測又は時間予測を介して予測的に符号化されてもよい。Bピクチャのブロックは、1つ又は2つ前に符号化された参照ピクチャを参照して、空間予測又は時間予測を介して予測的に符号化されてもよい。

30

## 【0072】

ビデオエンコーダ(603)は、ITU-T Rec. H.265のような所定のビデオ符号化技術又は標準に従って符号化動作を実行してもよい。その動作において、ビデオエンコーダ(603)は、入力ビデオシーケンスにおける時間的及び空間的冗長性を利用する予測符号化動作を含む様々な圧縮動作を実行してもよい。したがって、符号化ビデオデータは、使用されているビデオ符号化技術又は標準によって指定されたシンタックスに適合してもよい。

## 【0073】

一実施形態では、送信機(640)は、符号化ビデオと共に更なるデータを送信してもよい。ソースコーデック(630)は、符号化ビデオシーケンスの一部としてこのようなデータを含んでもよい。更なるデータは、時間/空間/SNRエンハンスメント層、冗長ピクチャ及びスライス、SEIメッセージ、VUIパラメータセットフラグメント等のような他の形式の冗長データを含んでもよい。

40

## 【0074】

ビデオは、時系列において複数のソースピクチャ(ビデオピクチャ)としてキャプチャされてもよい。イントラピクチャ予測(しばしばイントラ予測と略される)は、所与のピクチャ内の空間的相関を利用し、インターピクチャ予測は、ピクチャ間の(時間的又は他の)相関を利用する。一例では、カレントピクチャと呼ばれる符号化/復号中の特定のピクチャは

50

、ブロックに分割される。カレントピクチャ内のブロックがビデオにおける前に符号化されて依然としてバッファされている参照ピクチャ内の参照ブロックに類似する場合、カレントピクチャ内のブロックは、動きベクトルと呼ばれるベクトルによって符号化されてもよい。動きベクトルは、参照ピクチャ内の参照ブロックを指し、複数の参照ピクチャが使用されている場合には、参照ピクチャを識別する第3の次元を有してもよい。

【0075】

いくつかの実施形態では、双方向予測技術は、インターピクチャ予測において使用されてもよい。双方向予測技術によれば、ビデオにおけるカレントピクチャへの復号順で双方とも先行する(しかし、表示順ではそれぞれ過去及び将来のものでよい)第1の参照ピクチャ及び第2の参照ピクチャのような2つの参照ピクチャが使用される。カレントピクチャ内のブロックは、第1の参照ピクチャ内の第1の参照ブロックを指す第1の動きベクトルと、第2の参照ピクチャ内の第2の参照ブロックを指す第2の動きベクトルとによって符号化されてもよい。ブロックは、第1の参照ブロックと第2の参照ブロックとの組み合わせによって予測されてもよい。

10

【0076】

さらに、符号化効率を改善するために、インターピクチャ予測においてマージモード技術が使用できる。

【0077】

本開示のいくつかの実施形態によれば、インターピクチャ予測及びイントラピクチャ予測のような予測は、ブロックの単位で実行される。例えば、HEVC標準によれば、ビデオピクチャのシーケンス内のピクチャは、圧縮のために符号化ツリーユニット(CTU, coding tree unit)に分割され、ピクチャ内のCTUは、 $64 \times 64$ の画素、 $32 \times 32$ の画素又は $16 \times 16$ の画素のように、同じサイズを有する。一般的に、CTUは、1つの輝度CTBと2つの色差CTBである3つの符号化ツリーブロック(CTB, coding tree block)を含む。各CTUは、1つ又は複数の符号化ユニット(CU, coding unit)に再帰的に四分木分割されてもよい。例えば、 $64 \times 64$ の画素のCTUは、 $64 \times 64$ の画素の1つのCU、 $32 \times 32$ の画素の4つのCU又は $16 \times 16$ の画素の16個のCUに分割できる。一例では、各CUは、インター予測タイプ又はイントラ予測タイプのようなCUの予測タイプを決定するために分析される。CUは、時間的及び/又は空間的予測可能性に依存して1つ以上の予測ユニット(PU, prediction unit)に分割される。一般的に、各PUは、輝度予測ブロック(PB, prediction block)と2つの色差PBとを含む。一実施形態では、符号化(符号化/復号)における予測動作は、予測ブロックの単位で実行される。予測ブロックの一例として輝度予測ブロックを使用すると、予測ブロックは、 $8 \times 8$ の画素、 $16 \times 16$ の画素、 $8 \times 16$ の画素、 $16 \times 8$ の画素等のように、画素の値(例えば、輝度値)の行列を含む。

20

30

【0078】

図7は、本開示の他の実施形態によるビデオエンコーダ(703)の図を示す。ビデオエンコーダ(703)は、ビデオピクチャのシーケンス内のカレントビデオピクチャ内のサンプル値の処理ブロック(例えば、予測ブロック)を受信し、処理ブロックを符号化ビデオシーケンスの一部である符号化ピクチャに符号化するように構成される。一例では、ビデオエンコーダ(703)は、図4の例のビデオエンコーダ(403)の代わりに使用される。

40

【0079】

HEVCの例では、ビデオエンコーダ(703)は、 $8 \times 8$ のサンプルの予測ブロック等のような処理ブロックのサンプル値の行列を受信する。ビデオエンコーダ(703)は、処理ブロックが、例えば、レート歪み最適化を使用して、イントラモードを使用して最も良く符号化されるか、インターモードを使用して最も良く符号化されるか、双方向予測モードを使用して最も良く符号化されるかを決定する。処理ブロックがイントラモードで符号化される場合、ビデオエンコーダ(703)は、処理ブロックを符号化ピクチャに符号化するためにイントラ予測技術を使用してもよい。処理ブロックがインターモード又は双方向予測モードで符号化される場合、ビデオエンコーダ(703)は、処理ブロックを符号化ピクチャに符号化するために、それぞれインター予測技術又は双方向予測技術を使用してもよい。特定の

50

ビデオ符号化技術では、マージモード(merge mode)は、動きベクトル予測子以外の符号化された動きベクトル成分の恩恵を受けずに、動きベクトルが1つ以上の動きベクトル予測子から導出されるインターピクチャ予測サブモードでもよい。特定の他のビデオ符号化技術では、対象のブロックに適用可能な動きベクトル成分が存在してもよい。一例では、ビデオエンコーダ(703)は、処理ブロックのモードを決定するためのモード決定モジュール(図示せず)のような他の構成要素を含む。

**【0080】**

図7の例では、ビデオエンコーダ(703)は、図7に示されるように共に結合されたインターエンコーダ(730)と、イントラエンコーダ(722)と、残差計算器(723)と、スイッチ(726)と、残差エンコーダ(724)と、全体コントローラ(721)と、エントロピーエンコーダ(725)とを含む。

10

**【0081】**

インターエンコーダ(730)は、カレントブロック(例えば、処理ブロック)のサンプルを受信し、当該ブロックを参照ピクチャ内の1つ以上の参照ブロック(例えば、前のピクチャ及び後のピクチャ内のブロック)と比較し、インター予測情報(例えば、インター符号化技術による冗長情報の記述、動きベクトル、マージモード情報)を生成し、いずれかの適切な技術を使用して、インター予測情報に基づいてインター予測結果(例えば、予測ブロック)を計算するように構成される。いくつかの例では、参照ピクチャは、符号化ビデオ情報に基づいて復号された復号参照ピクチャである。

**【0082】**

イントラエンコーダ(722)は、カレントブロック(例えば、処理ブロック)のサンプルを受信し、場合によっては、当該ブロックを、同じピクチャ内で既に符号化されたブロックと比較し、変換後に量子化係数を生成し、場合によっては、イントラ予測情報(例えば、1つ以上のイントラ符号化技術によるイントラ予測方向情報)も生成するように構成される。また、一例では、イントラエンコーダ(722)は、同じピクチャ内のイントラ予測情報及び参照ブロックに基づいて、イントラ予測結果(例えば、予測ブロック)を計算する。

20

**【0083】**

全体コントローラ(721)は、全体制御データを決定し、全体制御データに基づいてビデオエンコーダ(703)の他の構成要素を制御するように構成される。一例では、全体コントローラ(721)は、ブロックのモードを決定し、当該モードに基づいて制御信号をスイッチ(726)に提供する。例えば、モードがイントラモードである場合、全体コントローラ(721)は、残差計算器(723)によって使用されるイントラモード結果を選択するようにスイッチ(726)を制御し、イントラ予測情報を選択してイントラ予測情報をビットストリームに含めるようにエントロピーエンコーダ(725)を制御する。モードがインターモードである場合、全体コントローラ(721)は、残差計算器(723)によって使用されるインター予測結果を選択するようにスイッチ(726)を制御し、インター予測情報を選択してインター予測情報をビットストリームに含めるようにエントロピーエンコーダ(725)を制御する。

30

**【0084】**

残差計算器(723)は、受信したブロックと、イントラエンコーダ(722)又はインターエンコーダ(730)から選択された予測結果との差(残差データ)を計算するように構成される。残差エンコーダ(724)は、残差データに基づいて動作し、残差データを符号化して変換係数を生成するように構成される。一例では、残差エンコーダ(724)は、残差データを空間ドメインから周波数ドメインに変換し、変換係数を生成するように構成される。次いで、変換係数は、量子化された変換係数を取得するための量子化処理を受ける。また、様々な実施形態では、ビデオエンコーダ(703)は、残差デコーダ(728)も含む。残差デコーダ(728)は、逆変換を実行し、復号された残差データを生成するように構成される。復号された残差データは、イントラエンコーダ(722)及びインターエンコーダ(730)によって適切に使用されてもよい。例えば、インターエンコーダ(730)は、復号された残差データ及びインター予測情報に基づいて復号ブロックを生成してもよく、イントラエンコーダ(722)は、復号された残差データ及びイントラ予測情報に基づいて復号ブロックを生成してもよ

40

50

い。復号ブロックは、復号ピクチャを生成するように適切に処理され、復号ピクチャは、メモリ回路(図示せず)にバッファされ、いくつかの例では参照ピクチャとして使用されてもよい。

**【 0 0 8 5 】**

エントロピーエンコーダ(725)は、符号化ブロックを含めるようにビットストリームをフォーマットするように構成される。エントロピーエンコーダ(725)は、HEVC標準のような適切な標準に従った様々な情報を含めるように構成される。一例では、エントロピーエンコーダ(725)は、全体制御データと、選択された予測情報(例えば、イントラ予測情報又はインター予測情報)と、残差情報と、他の適切な情報とをビットストリームに含めるように構成される。開示の対象物によれば、インターモード又は双方向予測モードのいずれかのマージサブモードでブロックを符号化する場合、残差情報は存在しない点に留意すべきである。

10

**【 0 0 8 6 】**

図8は、本開示の他の実施形態によるビデオデコーダ(810)の図を示す。ビデオデコーダ(810)は、符号化ビデオシーケンスの一部である符号化ピクチャを受信し、符号化ピクチャを復号して復元ピクチャを生成するように構成される。一例では、ビデオデコーダ(810)は、図4の例のビデオデコーダ(410)の代わりに使用される。

**【 0 0 8 7 】**

図8の例では、ビデオデコーダ(810)は、図8に示されるように共に結合されたエントロピーデコーダ(871)と、インターデコーダ(880)と、残差デコーダ(873)と、復元モジュール(874)と、イントラデコーダ(872)とを含む。

20

**【 0 0 8 8 】**

エントロピーデコーダ(871)は、符号化ピクチャから、当該符号化ピクチャが構成されるシンタックスエレメントを表す特定のシンボルを復元するように構成されてもよい。このようなシンボルは、例えば、ブロックが符号化されるモード(例えば、イントラモード、インターモード、双方向予測モード、マージサブモード又は他のサブモードにおける後者の2つ等)、それぞれイントラデコーダ(872)又はインターデコーダ(880)によって予測のために使用される特定のサンプル又はメタデータを識別できる予測情報(例えば、イントラ予測情報又はインター予測情報等)、例えば、量子化された変換係数の形式の残差情報等を含んでもよい。一例では、予測モードがインターモード又は双方向予測モードである場合、インター予測情報はインターデコーダ(880)に提供され、予測タイプがイントラ予測タイプである場合には、イントラ予測情報がイントラデコーダ(872)に提供される。残差情報は、逆量子化を受けてもよく、残差デコーダ(873)に提供される。

30

**【 0 0 8 9 】**

インターデコーダ(880)は、インター予測情報を受信し、インター予測情報に基づいてインター予測結果を生成するように構成される。

**【 0 0 9 0 】**

イントラデコーダ(872)は、イントラ予測情報を受信し、イントラ予測情報に基づいて予測結果を生成するように構成される。

**【 0 0 9 1 】**

残差デコーダ(873)は、逆量子化された変換係数を抽出するための逆量子化を実行し、逆量子化された変換係数を処理して残差を周波数ドメインから空間ドメインに変換するように構成される。また、残差デコーダ(873)は、特定の制御情報(量子化パラメータ(QP, Quantizer Parameter)を含む)を必要としてもよく、その情報は、エントロピーデコーダ(871)によって提供されてもよい(これは低ボリュームの制御情報のみである可能性があるため、データ経路は図示されていない)。

40

**【 0 0 9 2 】**

復元モジュール(874)は、空間ドメインにおいて、残差デコーダ(873)によって出力された残差と、予測結果(場合によっては、インター予測モジュール又はイントラ予測モジュールによって出力されたもの)とを結合して復元ブロックを形成するように構成され、当該

50

復元ブロックは、復元ピクチャの一部でもよく、また、復元ビデオの一部でもよい。視覚品質を改善するために、デブッキング動作のような他の適切な動作が実行されてもよい点に留意すべきである。

【 0 0 9 3 】

ビデオエンコーダ(403)、(603)及び(703)並びにビデオデコーダ(410)、(510)及び(810)は、いずれかの適切な技術を使用して実装されてもよい点に留意すべきである。一実施形態では、ビデオエンコーダ(403)、(603)及び(703)並びにビデオデコーダ(410)、(510)及び(810)は、1つ以上の集積回路を使用して実装されてもよい。他の実施形態では、ビデオエンコーダ(403)、(603)及び(703)並びにビデオデコーダ(410)、(510)及び(810)は、ソフトウェア命令を実行する1つ以上のプロセッサを使用して実装されてもよい。

10

【 0 0 9 4 】

HEVCのようないくつかの実施形態では、一次変換は4ポイント、8ポイント、16ポイント及び32ポイントの離散コサイン変換(DCT, discrete cosine transform)タイプ2(DCT-2)を含んでもよく、変換コア行列は、8ビットの整数(すなわち、8ビットの変換コア)を使用して表されてもよい。付録Iに示すように、より小さいDCT-2の変換コア行列は、より大きいDCT-2の変換コア行列の一部である。

【 0 0 9 5 】

DCT-2コア行列は対称的/反対称的な特性を示す。したがって、「部分的バタフライ(partial butterfly)」実装は、演算のカウント数(例えば、乗算、加算、減算、シフト等)を低減するためにサポートされてもよく、行列乗算の同じ結果が部分的バタフライを使用して取得できる。

20

【 0 0 9 6 】

VVCのようないくつかの実施形態では、上記の4ポイント、8ポイント、16ポイント及び32ポイントのDCT-2変換の他に、更なる2ポイント及び64ポイントのDCT-2も含まれてもよい。VVCで使用されているような64ポイントのDCT-2コアの例は、64×64の行列として付録IIに示されている。

【 0 0 9 7 】

HEVCで使用されるようなDCT-2及び4×4のDST-7に加えて、VVCで使用されるような適応多重変換(AMT, Adaptive Multiple Transform)(拡張多重変換(EMT, Enhanced Multiple Transform)又は多重変換選択(MTS, Multiple Transform Selection)としても知られる)方式が、インター及びイントラ符号化ブロックの双方の残差符号化のために使用できる。AMT方式は、HEVCにおける現在の変換以外のDCT/DSTファミリーからの複数の選択された変換を使用してもよい。新たに導入された変換行列はDST-7、DCT-8である。表1は、Nポイントの入力についての選択されたDST/DCTの基本関数の例を示す。

30

【表1】

変換タイプ	基本関数 $T_i(j), i, j=0, 1, \dots, N-1$
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	$T_i(j) = \sqrt{\frac{4}{2N + 1}} \cdot \sin\left(\frac{\pi \cdot (2i + 1) \cdot (j + 1)}{2N + 1}\right)$

40

【 0 0 9 8 】

VVCで使用されるような一次変換行列は、8ビット表現で使用されてもよい。AMTは、

50

幅及び高さの双方が32以下の変換ブロックに変換行列を適用する。AMTが適用されるか否かは、フラグ(例えば、mts\_flag)によって制御されてもよい。mts\_flagが0に等しい場合、いくつかの例では、DCT-2のみが残差データを符号化するために適用される。mts\_flagが1に等しい場合、例えば、表2に従って使用される水平及び垂直変換を識別するために2つのピンを使用して、インデックス(例えば、mts\_idx)が更に伝達されてもよい。1のタイプ値はDST-7が使用されることを意味し、2のタイプ値はDCT-8が使用されることを意味する。表2では、trTypeHor及びtrTypeVerの様子はmts\_idx[x][y][cldx]に依存する。

【表2】

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

10

20

## 【0099】

いくつかの実施形態では、暗示的なMTSは、上記の信号伝達に基づくMTS(すなわち、明示的なMTS)が使用されない場合に適用されてもよい。暗示的なMTSでは、変換選択は、信号伝達の代わりにブロック幅及び高さに従って行われる。例えば、暗示的なMTSでは、DST-7は、M×Nのブロックの短い辺(すなわち、M及びNのうち最小のもの)について選択され、DCT-2は、ブロックの長い辺(すなわち、M及びNのうち最大のもの)について選択される。

## 【0100】

各々がDST-7及びDCT-8の基本ベクトルによって構成される行列である例示的な変換コアが、付録IIIに示されている。

30

## 【0101】

VVCのようないくつかの例では、符号化ブロックの高さ及び幅の双方が64以下である場合、TBサイズは符号化ブロックサイズと同じである。符号化ブロックの高さ又は幅のいずれかが64よりも大きい場合、変換(逆変換、逆一次変換等)又はイントラ予測を実行するときに、符号化ブロックは、複数のサブブロックに更に分割され、各サブブロックの幅及び高さは64以下である。各サブブロックに対して1つの変換が実行できる。

## 【0102】

VVCにおけるいくつかの例におけるMTSの関連するシンタックス及び意味は、図9及び図10A～10Bにおいて以下に記載され得る(例えば、関連するシンタックスは、フレーム901及び1001を使用して強調表示されている)。図9は、変換ユニットのシンタックスの例を示す。図10A～10Bは、残差符号化のシンタックスの例を示す。変換ユニットの意味の例は以下の通りである。1に等しいcu\_mts\_flag[x0][y0]は、多重変換選択が関連する輝度変換ブロックの残差サンプルに適用されることを指定する。0に等しいcu\_mts\_flag[x0][y0]は、多重変換選択が関連する輝度変換ブロックの残差サンプルに適用されないことを指定する。配列インデックスx0、y0は、ピクチャの左上の輝度サンプルに対して考慮される変換ブロックの左上の輝度サンプルの位置(x0,y0)を指定する。cu\_mts\_flag[x0][y0]が存在しない場合、これは0に等しいと推定される。

40

## 【0103】

残差符号化の意味の例は以下の通りである。mts\_idx[x0][y0]は、どの変換カーネルが

50

カレント変換ブロックの水平方向及び垂直方向に沿った輝度残差サンプルに適用されるかを指定する。配列インデックス $x_0$ 、 $y_0$ は、ピクチャの左上の輝度サンプルに対して考慮される変換ブロックの左上の輝度サンプルの位置( $x_0, y_0$ )を指定する。 $mts\_idx[x_0][y_0]$ が存在しない場合、これは-1に等しいと推定される。

#### 【0104】

いくつかの実施形態では、符号化ブロックは、例えば、 $4 \times 4$ のサイズを有するサブブロックに分割できる。符号化ブロック内のサブブロック及び各サブブロック内の変換係数は、予め定義された走査順序に従って符号化されてもよい。少なくとも1つの非ゼロ変換係数を有するサブブロックについて、変換係数の符号化は、4つの走査パスのような複数の走査パスに分離されてもよい。各パスの中で、各サブブロック内の変換係数(係数とも呼ばれる)は、例えば、逆対角線走査順序で走査されてもよい。

10

#### 【0105】

図11は、変換係数の異なるタイプのシンタックスエレメントが生成できるサブブロック走査プロセス(1100)の例を示す。サブブロックは、16個の係数(1110)を有する $4 \times 4$ のサブブロックでもよい。16個の係数(1110)は、図11に示すように、0~15のような走査順序に基づいて走査されてもよい。第1のパスの中で、係数(1110)が走査され、3つのタイプのシンタックスエレメント(1101)-(1103)が、係数(1110)のそれぞれについて生成されてもよい。

(i)第1のタイプのシンタックスエレメント(1101)は、それぞれの変換係数の絶対変換係数レベル又は値(例えば $absLevel$ )がゼロよりも大きいかなを示す有意フラグ(例えば $sig\_coeff\_flag$ )でもよい。第1のタイプのシンタックスエレメントは、バイナリシンタックスエレメントでもよい。

20

(ii)第2のタイプのシンタックスエレメント(1102)は、それぞれの変換係数の絶対変換係数レベルのパリティを示すパリティフラグ(例えば、 $par\_level\_flag$ )でもよい。一例では、パリティフラグは、それぞれの変換係数の絶対変換係数レベルがゼロでない場合にのみ生成される。第2のタイプのシンタックスエレメントは、バイナリシンタックスエレメントでもよい。

(iii)第3のタイプのシンタックスエレメント(1103)は、それぞれの変換係数について( $absLevel-1$ ) 1が0よりも大きいかなを示す、1よりも大きいフラグ(例えば、 $rem\_abs\_gt1\_flag$ )でもよい。一例では、1よりも大きいフラグは、それぞれの変換係数の絶対変換係数レベルがゼロでない場合にのみ生成される。第3のタイプのシンタックスエレメントは、バイナリシンタックスエレメントでもよい。

30

#### 【0106】

第2のパスの中で、第4のタイプのシンタックスエレメント(1104)が生成されてもよい。第4のタイプのシンタックスエレメント(1104)は、2よりも大きいフラグ(例えば、 $rem\_abs\_gt2\_flag$ )でもよい。第4のタイプのシンタックスエレメント(1104)は、それぞれの変換係数の絶対変換係数レベルが4よりも大きいかなを示す。一例では、それぞれの変換係数について( $absLevel-1$ ) 1が0よりも大きい場合のみ、2よりも大きいフラグが生成される。第4のタイプのシンタックスエレメントは、バイナリシンタックスエレメントでもよい。

40

#### 【0107】

第3のパスの中で、第5のタイプのシンタックスエレメント(1105)が生成されてもよい。第5のタイプのシンタックスエレメント(1105)は $abs\_remainder$ で示され、それぞれの変換係数の絶対変換係数レベルの残りの値が4よりも大きいことを示す。一例では、第5のタイプのシンタックスエレメント(1105)は、それぞれの変換係数の絶対変換係数レベルが4よりも大きい場合にのみ生成される。第5のタイプのシンタックスエレメントは、非バイナリシンタックスエレメントでもよい。

#### 【0108】

第4のパスの中で、第6のタイプのシンタックスエレメント(1106)が、係数(1110)についてそれぞれの変換係数(1110)の符号を示す非ゼロ係数レベルで生成されてもよい。

50

## 【0109】

上記の様々なタイプのシンタックスエレメント(1101-1106)は、パスの順序及び各パスにおける走査順序に従ってエントロピーエンコーダに提供されてもよい。異なるタイプのシンタックスエレメントを符号化するために、異なるエントロピー符号化方式が使用されてもよい。例えば、一実施形態では、有意フラグ、パリティフラグ、1よりも大きいフラグ及び2よりも大きいフラグが、CABACベースのエントロピーエンコーダで符号化されてもよい。対照的に、第3及び第4のパスの中で生成されたシンタックスエレメントは、CABACバイパス型エントロピーエンコーダ(例えば、入力ビンについて固定の確率推定値を有するバイナリ算術エンコーダ)で符号化されてもよい。

## 【0110】

コンテキストモデル化は、いくつかのタイプの変換係数シンタックスエレメントのビンのコンテキストモデルを決定するために実行されてもよい。一実施形態では、コンテキストモデルは、ローカルテンプレート及び現在の係数の対角位置に従って、場合によっては他の要因と組み合わせて決定されてもよい。

## 【0111】

図12は、現在の係数(1220)のコンテキスト選択に使用されるローカルテンプレート(1230)の例を示す。変換係数の間の相関を利用するために、ローカルテンプレート(1230)によりカバーされる、前に符号化された係数は、現在のコンテキストのコンテキスト選択において使用されてもよい。ローカルテンプレート(1230)は、係数ブロック(1210)内の現在の係数(1220)の隣接する位置又は係数のセットをカバーしてもよい。係数ブロック(1210)は、 $8 \times 8$ の係数のサイズを有してもよい。係数ブロック(1210)は、 $4 \times 4$ の位置のサイズをそれぞれ有する4つのサブブロックに分割される。図12の例では、ローカルテンプレート(1230)は、現在の係数(1220)の右下側の5つの係数レベルをカバーする5位置テンプレートであると定義される。係数ブロック(1210)内の係数上の複数のパスに逆対角線の走査順序が使用される場合、ローカルテンプレート(1230)内の隣接係数は、現在の係数(1220)の前に処理される。

## 【0112】

コンテキストモデル化の間に、ローカルテンプレート(1230)内の係数レベルの情報は、コンテキストモデルを決定するために使用されてもよい。この目的のため、いくつかの実施形態では、テンプレートの大きさと呼ばれる尺度が、ローカルテンプレート(1230)内の変換係数又は変換係数レベルの大きさを測定するため或いは示すために定義される。次いで、テンプレートの大きさは、コンテキストモデルを選択するための基礎として使用されてもよい。

## 【0113】

一例では、テンプレートの大きさは、ローカルテンプレート(1230)内の部分的に復元された絶対変換係数レベルの、和(例えば、sumAbs1)であると定義される。部分的に復元された絶対変換係数レベルabsLevel1[x][y]は、それぞれの変換係数のシンタックスエレメント(sig\_coeff\_flag、par\_level\_flag及びrem\_abs\_gt1\_flag)のピンに従って決定されてもよい。これらの3つのタイプのシンタックスエレメントは、エントロピーエンコーダ又はエントロピーデコーダ内で実行されるサブブロックの変換係数上で第1のパスの後に取得される。一実施形態では、第1のパスの後の位置(x, y)における部分的に復元された絶対変換係数レベルabsLevel1[x][y]は、

$$\text{absLevel1}[x][y] = \text{sig\_coeff\_flag}[x][y] + \text{par\_level\_flag}[x][y] + 2 * \text{rem\_abs\_gt1\_flag}[x][y]$$

に従って決定されてもよく、ここで、x及びyは係数ブロック(1210)の左上角に関する座標であり、absLevel1[x][y]は位置(x, y)における部分的に復元された絶対変換係数レベルを表す。

## 【0114】

dは現在の係数の対角位置を表し、dはx及びyの和であり、numSigはローカルテンプレート(1230)における非ゼロ係数の数を表す。例えば、sumAbs1はローカルテンプレート(

10

20

30

40

50

1230)によってカバーされる係数について部分的に復元された絶対レベルabsLevel1[x][y]の和を表す。

【0115】

現在の係数(1220)のsig\_coeff\_flagを符号化する場合、コンテキストモデルインデックスは、sumAbs1及び現在の係数(1220)の対角位置dに依存して選択される。dはx及びyの和である。sumAbs1はローカルテンプレート(1230)によってカバーされる係数について部分的に復元された絶対レベルabsLevel1[x][y]の和を表す。一実施形態では、輝度成分について、コンテキストモデルインデックスは、

ctxSig=18\*max(0,state-1)+min(sumAbs1,5)+(d 2?12:(d 5?6:0)) 式1

に従って決定される。ここで、ctxSigは有意フラグシンタックスエレメントのコンテキストインデックスを表し、「state」は依存する量子化方式のスカラー量子化の状態を指定する。一例では、「state」は状態遷移プロセスを使用して導出される。式(1)は、

ctxldBase=18\*max(0,state-1)+(d 2?12:(d 5?6:0)) 式2

ctxSig=ctxldSigTable[min(sumAbs1,5)]+ctxldBase 式3

と等価である。ここで、ctxldBaseはコンテキストインデックススペースを表す。コンテキストインデックススペースは、state及び対角位置dに基づいて決定されてもよい。例えば、stateは0、1、2又は3の値を有してもよく、したがってmax(0,state-1)は3つの可能な値(0、1又は2)のうち1つを有してもよい。例えば、(d 2?12:(d 5?6:0))は、異なる範囲のd(d 2、2 =d 5、5 =d)に対応して、12、6又は0の値を取ってもよい。ctxldSigTable[]は配列データ構造を表してもよく、ctxldBaseに関する有意フラグのコンテキストインデックスオフセットを記憶する。例えば、異なるsumAbs1値について、min(sumAbs1,5)はsumAbs1値を5以下になるようにクリッピングする。次いで、クリッピングされた値がコンテキストインデックスオフセットにマッピングされる。例えば、ctxldSigTable[0~5]={0,1,2,3,4,5}の定義で、クリッピングされた値0、1、2、3、4又は5はそれぞれ0、1、2、3、4又は5にマッピングされる。

【0116】

一実施形態では、色差成分について、コンテキストインデックスは、

ctxSig=12\*max(0,state-1)+min(sumAbs1,5)+(d 2?6:0) 式4

に従って決定できる。式(4)は、以下の式(5)及び式(6)と等価である。

ctxldBase=12\*max(0,state-1)+(d 2?6:0) 式5

ctxSig=ctxldSigTable[min(sumAbs1,5)]+ctxldBase 式6

ここで、stateは依存する量子化が有効にされた場合に使用されるスカラー量子化を指定し、satateは状態遷移プロセスを使用して導出される。テーブルctxldSigTableは、コンテキストモデルインデックスオフセットctxldSigTable[0~5]={0,1,2,3,4,5}を記憶する。

【0117】

現在の係数(1220)のpar\_level\_flagを符号化する場合、コンテキストインデックスは、sumAbs1、numSig及び対角位置dに依存して選択される。numSigは、ローカルテンプレート(1230)内の非ゼロ係数の数を表す。例えば、輝度成分について、コンテキストインデックスは、

ctxPar=1+min(sumAbs1-numSig,4)+(d==0?15:(d 3?10:(d 10?5:0))) 式7

に従って決定されてもよい。式(7)は、

ctxldBase=(d==0?15:(d 3?10:(d 10?5:0))) 式8

ctxPar=1+ctxldTable[min(sumAbs1-numSig,4)]+ctxldBase 式9

と等価である。ここで、ctxParはパリティフラグのコンテキストインデックスを表し、ctxldTable[]は配列データ構造を表し、それぞれのctxldBaseに関するコンテキストインデックスオフセットを記憶する。例えば、ctxldTable[0~4]={0,1,2,3,4}である。

【0118】

色差について、コンテキストインデックスは

ctxPar=1+min(sumAbs1-numSig,4)+(d==0?5:0) 式10

に従って決定されてもよい。式(10)は、

10

20

30

40

50

$ctxIdBase=(d==0?5:0)$  式11

$ctxPar=1+ctxIdTable[\min(\text{sumAbs1}-\text{numSig},4)]+ctxIdBase$  式12

と等価である。ここで、テーブル $ctxIdTable$ はコンテキストモデルインデックスオフセットを記憶し、一例では、 $ctxIdTable[0\sim 4]=\{0,1,2,3,4\}$ である。

【0119】

現在の係数(1120)の $rem\_abs\_gt1\_flag$ 及び $rem\_abs\_gt2\_flag$ を符号化する場合、コンテキストモデルインデックスは、 $par\_level\_flag$ と同じ方法で決定されてもよい。

$ctxGt1=ctxPar$

$ctxGt2=ctxPar$

ここで、 $ctxGt1$ 及び $ctxGt2$ は、それぞれ、1よりも大きいフラグ及び2よりも大きいフラグのコンテキストインデックスを表す。

10

【0120】

$rem\_abs\_gt1\_flag$ 及び $rem\_abs\_gt2\_flag$ のような異なるタイプのシンタックスエレメントについて、異なるセットのコンテキストモデルが使用されてもよい点に留意すべきである。したがって、 $rem\_abs\_gt1\_flag$ に使用されるコンテキストモデルは、 $ctxGt1$ の値が $ctxGt2$ の値と等しい場合であっても、 $rem\_abs\_gt2\_flag$ のコンテキストモデルと異なる。

【0121】

順方向一次変換又は逆方向一次変換のような一次変換は、以下に説明するゼロアウト(zero-out)方法又はゼロアウト方式を利用してよい。VVCのようないくつかの例では、64ポイント(又は64長)のDCT-2について、最初の32個の係数のみが計算され、残りの係数は0に設定される。したがって、DCT-2変換を使用して符号化される $M\times N$ のブロックについて、左上の $\min(M,32)\times\min(N,32)$ の低周波数係数が保持又は計算される。残りの係数は0として設定され、伝達されなくてもよい。一例では、残りの係数は計算されない。係数ブロックのエントロピー符号化は、係数ブロックサイズを $\min(M,32)\times\min(N,32)$ として設定することによって実行されてもよく、それにより、 $M\times N$ のブロックの係数符号化が $\min(M,32)\times\min(N,32)$ の係数ブロックとしてみなされる。

20

【0122】

MTSが使用されるいくつかの例では、32ポイントのDST-7又はDCT-8について、最初の16個の係数のみが計算され、残りの係数は0として設定される。したがって、DST-7又はDCT-8変換を使用して符号化される $M\times N$ のブロックについて、左上の $\min(M,16)\times\min(N,16)$ の低周波数係数が保持される。残りの係数は0として設定され、伝達されなくてもよい。しかし、64ポイントのゼロアウトDCT-2が適用される場合に使用される係数符号化方式とは異なり、32ポイントのMTSでは、 $M$ 又は $N$ が16よりも大きい場合であっても、依然として全体の $M\times N$ のブロックに対して係数符号化が実行される。しかし、係数グループ(CG, coefficient group)が左上の $16\times 16$ の低周波数領域の外側にある場合(すなわち、係数グループがゼロアウト領域にある場合)、係数グループが非ゼロ係数を有するか否かを示すフラグ(例えば、 $coded\_sub\_block\_flag$ )は伝達されない。ゼロアウト領域は、係数がゼロである係数ブロック内の領域を示し、したがって、ゼロアウト領域内の係数はゼロである。残差符号化のシンタックスの例が以下に記載され、図13のフレーム1301を使用して強調表示されたテキストによって示される。

30

40

【0123】

一実施形態では、モード依存の分離不可能二次変換(NSST, non-separable secondary transform)が、エンコーダ側における順方向コア変換と量子化(例えば、係数の量子化)との間で使用され、デコーダ側における逆量子化(例えば、量子化係数の逆量子化)と逆方向コア変換との間で使用されてもよい。例えば、低い複雑性を保持するために、NSSTは、一次変換(又はコア変換)の後の低周波数係数に適用される。変換係数ブロックの幅( $W$ )及び高さ( $H$ )の双方が8以上である場合、 $8\times 8$ のNSSTが変換係数ブロックの左上の $8\times 8$ の領域に適用される。そうでなく、変換係数ブロックの幅 $W$ 又は高さ $H$ のいずれかが4である場合、 $4\times 4$ のNSSTが適用され、 $4\times 4$ のNSSTが変換係数ブロックの左上の $\min(8,W)\times\min(8,H)$

50

in(8,H)の領域に対して実行される。上記の変換選択方法は輝度成分及び色差成分の双方に適用される。

【 0 1 2 4 】

4 × 4の入力ブロックを例として使用して、NSSTの行列乗算の実装について以下に説明する。4 × 4の入力ブロックXは式(13)のように記載される。

【数 1】

$$X = \begin{bmatrix} X_{00} & X_{01} & X_{02} & X_{03} \\ X_{10} & X_{11} & X_{12} & X_{13} \\ X_{20} & X_{21} & X_{22} & X_{23} \\ X_{30} & X_{31} & X_{32} & X_{33} \end{bmatrix} \quad \text{式13}$$

10

【 0 1 2 5 】

入力ブロックXは、式14においてベクトル

【数 2】

$$\vec{X}$$

20

として表されてもよく、

【数 3】

$$\vec{X} = [X_{00} X_{01} X_{02} X_{03} X_{10} X_{11} X_{12} X_{13} X_{20} X_{21} X_{22} X_{23} X_{30} X_{31} X_{32} X_{33}]^T \quad \text{式14}$$

である。分離不可能変換は、

【数 4】

$$\vec{F} = T \cdot \vec{X}$$

30

として計算され、

【数 5】

$$\vec{F}$$

は変換係数ベクトルを表し、Tは16 × 16の変換行列である。16 × 1の変換係数ベクトル

【数 6】

$$\vec{F}$$

40

は、その後、入力ブロックXの走査順序(例えば、水平走査順序、垂直走査順序又は対角走査順序)を使用して4 × 4のブロックに再編成される。より小さいインデックスを有する係数は、4 × 4の係数ブロックにおいてより小さい走査インデックスと置き換えられてもよい。

【 0 1 2 6 】

50

二次変換の変換コアは、イントラ予測モードによって選択されてもよい。各イントラ予測モードについて、二次変換コアのセットが定義されてもよく、選択はビットストリームで伝達されるインデックスによって示されてもよく、及び/又は、他のシンタックスエレメント(例えば、MTSインデックス)によって示されてもよい。

【0127】

二次変換における係数の記憶及び計算の複雑さを低減するために、ゼロアウト方式ベースの二次変換が適用されてもよい。二次変換がゼロアウト方式を含む場合、二次変換は $M \times N$ のブロックの第1の $K$ 個の係数のみを計算する。ここで $K$ は $M \times N$ よりも小さい。残りの $(M \times N - K)$ 個の係数は0として設定されてもよい。例えば、残りの $(M \times N - K)$ 個の係数は計算されない。

10

【0128】

いくつかの実施形態では、空間変化変換(SVT, spatially varying transform)とも呼ばれるサブブロック変換(SBT, sub-block transform)が使用される。SBTはインター予測残差に適用されてもよい。いくつかの例では、残差ブロックは符号化ブロックに含まれ、符号化ブロックよりも小さい。したがって、SBTにおける変換サイズは、符号化ブロックサイズよりも小さい。残差ブロックによってカバーされない領域については、ゼロの残差を仮定してもよく、変換処理は実行されない。

【0129】

図14A~14Dは、SBTにおいてサポートされるサブブロックタイプ(SVT-H、SVT-V)(例えば、水平又は垂直分割)、サイズ及び位置(例えば、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)を示す。文字「A」で記された斜線の領域は、変換による残差ブロックであり、他の領域は、変換のないゼロの残差と仮定する。

20

【0130】

一例として、図15A~15Fは、SBTが使用される場合のビデオ符号化標準(例えば、VVC)の仕様テキストへの変更を示す。追加されたテキストを含む変更は、(2101)~(2108)のフレームで示されている。図示のように、更なるシンタックスエレメント(例えば、更なるオーバーヘッドビット`cu_sbt_flag`、`cu_sbt_quad_flag`、`cu_sbt_horizontal_flag`及び`cu_sbt_pos_flag`)は、それぞれ、サブブロックタイプ(水平又は垂直)、サイズ(半分又は四半分)及び位置(左、右、上又は下)を示すために伝達されてもよい。

【0131】

シーケンスパラメータセットRBSPの意味の例が以下に示される。0に等しい`sps_sbt_enabled_flag`は、インター予測されるCUについて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、インター予測されるCUについてのサブブロック変換が有効であることを指定する。

30

【0132】

一般的なスライスヘッダの意味の例が以下に示される。0に等しい`slice_sbt_max_size_64_flag`は、サブブロック変換を許容する最大CU幅及び高さが32であることを指定する。1に等しい`slice_sbt_max_size_64_flag`は、サブブロック変換を許容する最大CU幅及び高さが64であることを指定する。`maxSbtSize`は`maxSbtSize=slice_sbt_max_size_64_flag?64:32`として決定されてもよい。

40

【0133】

符号化ユニットの意味の例が以下に示される。1に等しい`cu_sbt_flag[x0][y0]`は、カレント符号化ユニットについ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、カレント符号化ユニットについ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に等しいと推定される。サブブロック変換が使用される場合、符号化ユニットは、2つの変換ユニットにタイル設定され、一方の変換ユニットは残差を有し、他方は残差を有さない。

【0134】

いくつかの例では、1に等しい`cu_sbt_quad_flag[x0][y0]`は、カレント符号化ユニットについて、サブブロック変換がカレント符号化ユニットの1/4のサイズの変換ユニット

50

を含むことを指定する。いくつかの例では、0に等しい`cu_sbt_quad_flag[x0][y0]`は、カレント符号化ユニットについて、サブブロック変換がカレント符号化ユニットの1/2の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に等しいと推定される。

【0135】

いくつかの例では、1に等しい`cu_sbt_horizontal_flag[x0][y0]`は、カレント符号化ユニットが水平分割によって2つの変換ユニットにタイル設定されることを指定する。0に等しい`cu_sbt_horizontal_flag[x0][y0]`は、カレント符号化ユニットが垂直分割によって2つの変換ユニットにタイル設定されることを指定する。

【0136】

いくつかの例では、`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`に等しく設定される。そうでない場合(`cu_sbt_quad_flag[x0][y0]`が0に等しい場合)、`cu_sbt_horizontal_flag[x0][y0]`は`allowSbtHoriHalf`に等しく設定される。

【0137】

いくつかの例では、1に等しい`cu_sbt_pos_flag[x0][y0]`は、カレント符号化ユニットにおける第1の変換ユニットの`tu_cbf_luma`、`tu_cbf_cb`及び`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、カレント符号化ユニットにおける第2の変換ユニットの`tu_cbf_luma`、`tu_cbf_cb`及び`tu_cbf_cr`が

【0138】

スケールされた変換係数についての変換プロセスの例について以下に説明する。

【0139】

プロセスへの入力は、カレントピクチャの左上の輝度サンプルに対するカレント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`)、カレント変換ブロックの幅を指定する変数`nTbW`、カレント変換ブロックの高さを指定する変数`nTbH`、カレントブロックの色成分を指定する変数`cldx`、スケールされた変換係数の $(nTbW) \times (nTbH)$ の配列`d[x][y]`( $x=0..nTbW-1$ ,  $y=0..nTbH-1$ )である。

【0140】

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]`( $x=0..nTbW-1$ ,  $y=0..nTbH-1$ )である。

【0141】

`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`及び垂直変換カーネルを指定する変数`trTypeVer`は、表3(図15F)において`cu_sbt_horizontal_flag[xTbY][yTbY]`及び`cu_sbt_pos_flag[xTbY][yTbY]`に依存して導出される。

【0142】

そうでない場合(`cu_sbt_flag[xTbY][yTbY]`が0に等しい場合)、水平変換カーネルを指定する変数`trTypeHor`及び垂直変換カーネルを指定する変数`trTypeVer`は、表4(図15Fに示す)において`mts_idx[xTbY][yTbY]`及び`CuPredMode[xTbY][yTbY]`に依存して導出される。

【0143】

残差サンプルの $(nTbW) \times (nTbH)$ の配列`r`は以下のように導出される。

【0144】

スケールされた変換係数`d[x][y]`( $x=0..nTbW-1$ ,  $y=0..nTbH-1$ )の各(垂直)列は、入力として、変換ブロックの高さ`nTbH`、リスト`d[x][y]`( $y=0..nTbH-1$ )、及び`trTypeVer`に等しく設定された変換タイプ変数`trType`によって、各列 $x=0..nTbW-1$ についての一次元変換プロセスを呼び出すことによって、`e[x][y]`( $x=0..nTbW-1$ ,  $y=0..nTbH-1$ )に変換され、出力は、リスト`e[x][y]`( $y=0..nTbH-1$ )である。

10

20

30

40

50

## 【 0 1 4 5 】

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

## 【 0 1 4 6 】

$g[x][y]=Clip3(CoeffMin,CoeffMax,(e[x][y]+256) \quad 9)$

## 【 0 1 4 7 】

結果の配列 $g[x][y]$ ( $x=0..nTbW-1$ ,  $y=0..nTbH-1$ )の各(水平)行は、入力として、変換ブロックの幅 $nTbW$ 、リスト $g[x][y]$ ( $x=0..nTbW-1$ )、及び $trTypeHor$ に等しく設定された変換タイプ変数 $trType$ によって、各行 $y=0..nTbH-1$ についての一次元変換プロセスを呼び出すことによって、 $r[x][y]$ ( $x=0..nTbW-1$ ,  $y=0..nTbH-1$ )に変換され、出力は、リスト $r[x][y]$ ( $x=0..nTbW-1$ )である。

10

## 【 0 1 4 8 】

いくつかの例では、2つの異なる方式の係数符号化がゼロアウトを有する64ポイントのDCT-2及びゼロアウトを有する32ポイントのMTSに適用され、したがって、係数符号化は統一された設計ではない。係数符号化は変換係数のエントロピー符号化を参照してもよい。

## 【 0 1 4 9 】

ゼロアウトを有する32ポイントのMTSに適用される係数符号化について、変換係数はゼロと仮定しているが、ゼロアウト領域での変換係数が走査されてもよい。エンコーダが完全に設計されていない場合、エンコーダは、ゼロアウト領域において非ゼロ係数を計算できる。したがって、左上の非ゼロ領域の境界におけるCGは、依然としてゼロアウト領域にアクセスし、変換係数のエントロピー符号化に使用されるコンテキスト値を導出してよい。エンコーダ及びデコーダは、異なるコンテキスト値(例えば、エンコーダについての非ゼロ値及びデコーダについてのゼロ値)を取得し、したがって、クラッシュ又は不一致のような、デコーダの予測不可能な挙動を生成する可能性がある。

20

## 【 0 1 5 0 】

いくつかの例では、32ポイントのDST-7/DCT-8を使用し得るSBTが適用されるので、SBTが適用される場合の変換のゼロアウトも適切に設計される必要がある。

## 【 0 1 5 1 】

二次変換はゼロアウト方法を含むことができるので、二次変換が適用される場合の変換のゼロアウトは、適切に設計される必要がある。

30

## 【 0 1 5 2 】

本明細書に記載の実施形態は、別々に或いはいずれかの順序で組み合わせて使用されてもよい。さらに、実施形態は、エンコーダ、デコーダ等の中の処理回路(例えば、1つ以上のプロセッサ又は1つ以上の集積回路)によって実装されてもよい。一例では、1つ以上のプロセッサは、非一時的なコンピュータ読み取り可能媒体に記憶されたプログラムを実行することができる。

## 【 0 1 5 3 】

本開示において、MTS候補のDST-7に適用可能な実施形態はDST-4に適用可能であり、その逆も同様である。同様に、MTS候補のDCT-8に適用可能な実施形態もまたDCT-4に適用可能であり、その逆も同様である。

40

## 【 0 1 5 4 】

本開示において、二次変換は、NSST、又はNSSTの代替設計である縮小二次変換(RST, Reduced Secondary Transform)を示してもよい。NSSTに適用可能な実施形態は、RSTに適用可能である。

## 【 0 1 5 5 】

デコーダ側では、変換係数を含む係数ブロックは、残差データを有する残差ブロックに復号されてもよく、したがって、TUは係数ブロック又は残差ブロックを示してもよい。本開示において、係数ユニット(又は係数セット)は、TU又は係数ブロック内の全ての係数又は一部の係数を含む変換係数(係数とも呼ばれる)のセットとして定義されてもよい。例えば、係数ユニットは、係数ブロック内の全ての非ゼロ係数を含む最小のサブブロックであ

50

るサブブロックを示してもよい。例えば、DCT-2のみを使用して符号化された $64 \times 64$ のTUについて、左上の $32 \times 32$ の領域内の低周波数係数のみが保持され、 $64 \times 64$ のTU内の残りの係数は0に設定され、したがって、係数ユニットは $64 \times 64$ のTU内の左上の $32 \times 32$ の領域である。他の例では、DST-7又はDCT-8を使用して符号化された $32 \times 32$ のTUについて、左上の $16 \times 16$ の領域内の低周波数係数のみが保持され、 $32 \times 32$ のTU内の残りの係数は0に設定され、したがって、係数ユニットは $32 \times 32$ のTU内の左上の $16 \times 16$ の領域である。

【0156】

TUは複数の係数ユニットを有してもよく、複数の係数ユニットはTU内の全ての非ゼロ係数をむ(すなわち、TU内の全ての非ゼロ係数は複数の係数ユニットに含まれる)。したがって、複数の係数ユニットの外側にあるTU内の係数はゼロである。

10

【0157】

図16A~16Dは、本開示の実施形態による係数ユニットの例を示す。図16Aを参照すると、 $16 \times 16$ の係数ブロック(1610)は、斜線の領域0、1A、2A及び3Aと、白色の領域5とを含む。斜線の領域0、1A、2A及び3Aは、非ゼロ変換係数を含んでもよく、白色の領域5は、非ゼロ変換係数を含まない(すなわち、白色の領域5における変換係数はゼロである)。図16Aを参照すると、係数ブロック(1610)に4つの係数ユニット(例えば、斜線の領域0、1A、2A及び3A)が存在する。

【0158】

図16Bを参照すると、係数ブロック(1610)は、斜線の領域0及び2Aと、斜線の領域1Bと、白色の領域5とに分割されてもよい。斜線の領域0、1B及び2Aは、非ゼロ変換係数を含んでもよく、白色の領域5は、非ゼロ変換係数を含まない。図16Bを参照すると、係数ブロック(1610)に3つの係数ユニット(例えば、斜線の領域0、1B及び2A)が存在する。図16A及び図16Bを参照すると、斜線の領域1Bは斜線の領域1A及び3Aを含む。

20

【0159】

図16Cを参照すると、係数ブロック(1610)は、斜線の領域0及び1Aと、斜線の領域2Cと、白色の領域5とに分割されてもよい。斜線の領域0、1A及び2Cは、非ゼロ変換係数を含んでもよく、白色の領域5は、非ゼロ変換係数を含まない。図16Cを参照すると、係数ブロック(1610)に3つの係数ユニット(例えば、斜線の領域0、1A及び2C)が存在する。図16A及び図16Cを参照すると、斜線の領域2Cは斜線の領域2A及び3Aを含む。

30

【0160】

図16Dを参照すると、係数ブロック(1610)は、斜線の領域0と、斜線の領域1Dと、白色の領域5とに分割されてもよい。斜線の領域0及び1Dは、非ゼロ変換係数を含んでもよく、白色の領域5は、非ゼロ変換係数を含まない。図16Dを参照すると、係数ブロック(1610)に2つの係数ユニット(例えば、斜線の領域0及び1D)が存在する。図16A及び図16Dを参照すると、斜線の領域1Dは斜線の領域1A、2A及び3Aを含む。

【0161】

一般的に、係数ユニットは、いずれか適切な形状又はサイズを有してもよい。図16A~16Dを参照すると、係数ユニットは、正方形の形状(例えば、係数ユニット1A、2A、3A)、矩形の形状(例えば、係数ユニット1B、2C)、不規則な形状(例えば、係数ユニット1Dの「L」の形状)を有してもよい。したがって、白色の領域5のようなゼロアウト領域内の係数がビットストリームで伝達されないように、係数符号化が適合されてもよい。

40

【0162】

係数ブロックは、異なる数の係数ユニットに分割されてもよい。例えば、係数ブロック(1610)は、それぞれ図16A~16Dに示すように、4つ、3つ、3つ及び2つの係数ユニットに分割されてもよい。

【0163】

本開示の実施形態によれば、TUの係数符号化は、各係数ユニットが独立して符号化される係数ユニットの単位で処理されてもよい。したがって、TU内の1つの係数ユニットの係数符号化は、TU内の他の係数ユニットの符号化情報にアクセスする必要はない。

50

## 【0164】

様々な処理順序が、係数ブロック内の複数の係数ユニットを処理するために使用されてもよい。図16Dを参照すると、処理回路は、係数ユニット0及び1Dを同時に処理し始めてもよい。一例では、処理回路は、係数ユニット1Dを処理し始める前に係数ユニット0を処理し始めてもよい。一例では、処理回路は、係数ユニット0を処理し始める前に係数ユニット1Dを処理し始めてもよい。

## 【0165】

いくつかの例では、フラグ(すなわち、係数ユニットフラグ)が、係数ユニットが少なくとも非ゼロ係数を含むか否かを示すために使用されてもよい。例えば、それぞれ図16Aの係数ユニット0及び1A~3Aについて、4つのフラグが使用されてもよい。同様に、それぞれ図16Bの係数ユニット0、1B及び2Aについて、3つのフラグが使用されてもよく、それぞれ図16Cの係数ユニット0、1A及び2Cについて、3つのフラグが使用されてもよく、それぞれ図16Dの係数ユニット0及び1Dについて、2つのフラグが使用されてもよい。

10

## 【0166】

いくつかの例では、各係数ユニットについて、係数ユニットフラグが伝達されてもよい。一例では、TUが1つの係数ユニットのみを有する場合、当該1つの係数ユニットの係数ユニットフラグは、TUの係数ブロックフラグ(CBF, coefficient block flag)から推定されてもよく、したがって、係数ユニットフラグは伝達されない。TUのCBFは、TUが少なくとも1つの非ゼロ係数を有するか否かを示してもよい。

20

## 【0167】

一例では、TUは複数の係数ユニットに分割され、TUは非ゼロ係数を有する。処理順序における最後の係数ユニットのみが非ゼロ係数を有する場合、最後の係数ユニットの係数ユニットフラグは、最後の係数ユニットが非ゼロ係数を有することを示し、信号伝達されずに推定される。

## 【0168】

一実施形態では、係数ユニットの係数符号化は、係数グループ(CG, coefficient group)の単位で処理されてもよい。CGは、係数ユニット内に16個の係数を含むことができる。CGは、正方形、矩形等のようないずれか適切な形状を有してもよい。例えば、CGは、係数ユニット内の4×4のサブブロック、2×8のサブブロック又は8×2のサブブロックでもよい。

30

## 【0169】

一例では、係数ユニットは、複数のCGを含み、非ゼロ係数を有する。処理順序における最後のCGのみが非ゼロ係数を有する場合、最後のCGについてのフラグ(例えば、coded\_sub\_block\_flag)は、最後のCGが非ゼロ係数を有するか否かを示すか、或いは、信号伝達されずに推定できず、最後のCGについてのフラグは、最後のCGが少なくとも1つの非ゼロ係数を有することを示す。

## 【0170】

一例では、二次変換なしに符号化されたTUについて、TUは、1つの係数ユニットのみを有する。

40

## 【0171】

一例では、ゼロアウト方法を含む(或いは使用する)一次変換、SBT、又はゼロアウト方法を含む一次変換とSBTとの組み合わせがTUに適用される場合、係数ユニットのサイズ及び位置は、ゼロアウト方法を含む一次変換及び/又はSBTの後に係数ユニットが非ゼロ領域を含むように、ゼロアウト方法を含む一次変換及び/又はSBTから推定されてもよい。非ゼロ領域は、少なくとも1つの非ゼロ係数を含む。

## 【0172】

一例では、TUが、ゼロアウト方法を含む一次変換、SBT等によるもののような既知のゼロアウト領域を有さない場合、上記のように、係数ユニットのサイズ及び位置は、TUのサイズ及び位置と同じであると推定されてもよい。一例では、係数ユニットはTUである。

50

## 【 0 1 7 3 】

いくつかの実施形態では、一次変換が $M \times N$ の係数ブロックに適用され、ゼロアウト方法を使用し、一次変換は、非DCT-2変換、又はDCT-2変換と非DCT-2変換との組み合わせでもよい。一次変換は水平 $M$ ポイント変換と垂直 $N$ ポイント変換とを含んでもよい。したがって、 $X$ が水平 $M$ ポイント変換について保持又は計算される係数の数であり、 $Y$ が垂直 $N$ ポイント変換について保持又は計算される係数の数である場合、比 $X/M$ は1未満であり、及び/又は、比 $Y/N$ は1未満である。本開示の態様によれば、 $M \times N$ の係数ブロックのエントロピー符号化は、エントロピー符号化ブロックサイズを $\min(M, X) \times \min(N, Y)$ として設定することによって実装されてもよい。したがって、 $M \times N$ の係数ブロックをエントロピー符号化する場合、 $M \times N$ の係数ブロックは、 $\min(M, X) \times \min(N, Y)$ の領域としてみなされ、例えば、左上の $\min(M, X) \times \min(N, Y)$ 領域内の変換係数のみがエントロピー符号化され、残りの係数はエントロピー符号化されない(例えば、残りの係数はゼロとして設定される)。さらに、一例では、左上の $\min(M, X) \times \min(N, Y)$ の領域の外側の残りの係数は、左上の $\min(M, X) \times \min(N, Y)$ の領域をエントロピー符号化する場合にアクセスされない。

10

## 【 0 1 7 4 】

いくつかの例では、非DCT-2変換は、MTSにおいて使用される32ポイントのDST-7/DCT-8、SBTにおいて使用される32ポイントのDST-7/DCT-8、及び/又は暗示的なMTSで使用される32ポイントのDST-7/DCT-8である。 $X$ 及び $Y$ が16である場合、エントロピー符号化ブロックサイズは $16 \times 16$ であり、左上の $16 \times 16$ の領域のみがエントロピー符号化される。さらに、一例では、左上の $16 \times 16$ の領域の外側の残りの係数は、左上の $16 \times 16$ の領域をエントロピー符号化する場合にアクセスされない。

20

## 【 0 1 7 5 】

上記のように、一次変換(例えば、非DCT-2変換、又はDCT-2変換と非DCT-2変換の組み合わせ)が $M \times N$ の係数ブロックに適用され、ゼロアウト方法を使用してもよい。比 $X/M$ は1未満であり、及び/又は、比 $Y/N$ は1未満である。 $M \times N$ の係数ブロック内の左上の $\min(M, X) \times \min(N, Y)$ の領域の外側の領域の係数はゼロとみなされてもよく、当該領域はゼロアウト領域と呼ばれてもよい。本開示の態様によれば、 $M \times N$ の係数ブロックのエントロピー符号化は、全体の $M \times N$ の係数ブロックに対して実装されてもよい。しかし、例えば、現在の係数のエントロピー符号化に使用されるコンテキスト値を導出するためにゼロアウト領域に位置する変換係数のシンタックスエレメントにアクセスする場合、変換係数のデフォルト値が使用されてもよい。変換係数のシンタックスエレメントは、係数関連シンタックスエレメントとも呼ばれてもよい。

30

## 【 0 1 7 6 】

一例では、現在のCGのCGフラグ(例えば、`coded_sub_block_flag`)をエントロピー符号化する場合、それぞれの1つ以上の隣接するCGについての1つ以上の隣接するCGフラグが、現在のCGのCGフラグをエントロピー符号化するためのコンテキスト値を導出するために使用されてもよい。しかし、1つ以上の隣接するCGのうち1つがゼロアウト領域に位置する場合、コンテキスト値を導出するために1つ以上の隣接するCGのうち1つについてデフォルト値(例えば、0)が使用されてもよい。

## 【 0 1 7 7 】

一例では、現在の変換係数のシンタックスエレメント(例えば、`sig_coeff_flag`、`par_level_flag`、`rem_abs_gt1_flag`、`rem_abs_gt2_flag`等のような係数関連のフラグ)をエントロピー符号化する場合、隣接係数のシンタックスエレメントが、現在の変換係数のコンテキスト値を導出するために使用されてもよい。隣接係数がゼロアウト領域に位置する場合、デフォルト値(例えば、0)が代わりに使用されてもよい。

40

## 【 0 1 7 8 】

図17は、第1の領域(1720)及び第2の領域(1730)を含む $32 \times 32$ の係数ブロック(1710)を示す。第1の領域(1720)は、係数ブロック(1710)内の左上の $16 \times 16$ の領域であり、第2の領域(1730)は、第1の領域(1720)の外側である。第1の領域(1720)はA~CのCGを含み、第2の領域(1730)はAB、AR、BB及びCRのCGを含む。一例では、エントロピー符

50

号化順序は、係数ブロック(1710)の右下角から始まり、係数ブロック(1710)の左上角で終わる逆対角線順序である。したがって、AB及びARのCGはAのCGの前にエントロピー符号化され、BBのCGはBのCGの前にエントロピー符号化され、CRのCGはCのCGの前にエントロピー符号化される。

#### 【0179】

一例では、第1の領域(1720)における変換係数のみが保持され、第2の領域(1730)における残りの係数は0として設定され、したがって、第2の領域(1730)は、ゼロアウト領域とも呼ばれる。AのCGの第1のCGフラグ(例えば、`coded_sub_block_flag`)のエントロピー符号化のために、ARのCG(右側のCG)及びABのCG(下側のCG)が、第1のCGフラグについてのコンテキスト値を導出するためにアクセスされてもよい。AR及びABのCGはゼロアウト領域(1730)に位置するので、デフォルト値(例えば、0)が、それぞれAR及びABのCGのCGフラグに割り当てられてもよい。BのCGについての第2のCGフラグ(例えば、`coded_sub_block_flag`)のエントロピー符号化のために、BBのCGがコンテキスト値を導出するためにアクセスされるべきである。BBのCGはゼロアウト領域(1730)に位置するので、デフォルト値(例えば、0)が、BBのCGのCGフラグに割り当てられてもよい。CのCGの第3のCGフラグ(例えば、`coded_sub_block_flag`)のエントロピー符号化のために、CRのCGがコンテキスト値を導出するためにアクセスされるべきである。CRのCGはゼロアウト領域(1730)に位置するので、デフォルト値(例えば、0)が、CRのCGのCGフラグに割り当てられてもよい。

#### 【0180】

いくつかの実施形態では、図18に示すように、ゼロアウト方法を含む二次変換が、 $X \times Y$ の係数ブロック(1810)の左上の $M \times N$ の領域(1820)に適用される。例えば、最初の16個の係数は保持され、左上の $M \times N$ の領域(1820)内の残りの $(M \times N - 16)$ 個の係数は0に設定される。一例では、 $X$ 及び $Y$ は32であり、 $M$ 及び $N$ は16であり、したがって、 $4 \times 4$ の領域0内の最初の16個の係数は保持され、領域2内の残りの $(16 \times 16 - 16)$ 個の係数は0として設定される。したがって、領域2は、ゼロアウト領域又はゼロ領域と呼ばれる。図18を参照すると、 $X \times Y$ の係数ブロック(1810)は、領域0、領域1及び領域2の3つの領域を含む。領域2は、二次変換におけるゼロアウト方法によるゼロアウト領域である。領域0及び/又は領域1は、非ゼロ変換係数を有してもよい。二次変換は領域1には適用されず、すなわち、領域1は二次変換によって処理されない。

#### 【0181】

$X \times Y$ の係数ブロック(1810)のエントロピー符号化は、全体の $X \times Y$ の係数ブロック(1810)に対して実行されてもよい。しかし、エントロピー符号化に使用されるコンテキスト値を導出するために係数がゼロアウト領域2に位置する係数関連シンタックスエレメント(すなわち、係数のシンタックスエレメント)にアクセスする場合、係数についてデフォルト値が使用されてもよい。

#### 【0182】

一実施形態では、現在のCGのCGフラグ(例えば、`coded_sub_block_flag`)をエントロピー符号化する場合、それぞれの1つ以上の隣接するCGの1つ以上の隣接するCGフラグが、CGフラグについてのコンテキスト値を導出するために使用されてもよい。しかし、1つ以上の隣接するCGのうち1つが、ゼロアウト方法を使用する二次変換によって形成されるゼロアウト領域2に完全に位置する場合、デフォルト値(例えば、0)が代わりに使用されてもよい。図18を参照すると、領域0は、A、B及びCのCGを含み、領域2(例えば、ゼロアウト領域)は、AB、AR及びBBのCGを含む。AB及びARのCGはゼロアウト領域2に完全に位置するので、第1のCGフラグのコンテキスト値を導出するために、AB及びARのCGのCGフラグの代わりにデフォルト値(例えば、0)が使用されてもよい。BBのCGはゼロアウト領域2に完全に位置するので、第2のCGフラグのコンテキスト値を導出するために、BBのCGのCGフラグの代わりにデフォルト値(例えば、0)が使用されてもよい。

#### 【0183】

一実施形態では、現在のCGのCGフラグ(例えば、`coded_sub_block_flag`)をエントロ

ピー符号化する場合、それぞれの1つ以上の隣接するCGの1つ以上の隣接するCGフラグが、CGフラグについてのコンテキスト値を導出するために使用されてもよい。しかし、1つ以上の隣接するCGのうち1つ(例えば、図18におけるCRのCG)がゼロアウト領域2に部分的に位置する場合、1つ以上の隣接するCGのうち1つ(例えば、図18におけるCRのCG)における非ゼロ係数が、コンテキスト値を導出するために使用されてもよい。

【0184】

一実施形態では、現在の変換係数についてのシンタックスエレメント(例えば、sig\_coeff\_flag、par\_level\_flag、rem\_abs\_gt1\_flag、rem\_abs\_gt2\_flag等のような係数関連フラグ)をエントロピー符号化する場合、シンタックスエレメントは伝達されない。現在の係数は、現在の係数がゼロアウト領域2のようなゼロアウト領域に位置する場合に現在の係数が0であることを示すデフォルト値として推定されてもよい。

10

【0185】

一実施形態では、現在の変換係数についてのシンタックスエレメント(例えば、sig\_coeff\_flag、par\_level\_flag、rem\_abs\_gt1\_flag、rem\_abs\_gt2\_flag等のような係数関連フラグ)をエントロピー符号化する場合、1つ以上の隣接係数が、シンタックスエレメントについてのコンテキスト値を導出するために使用されてもよい。1つ以上の隣接係数のうち1つがゼロ領域2に位置する場合、デフォルト値(例えば、0)が代わりに使用されてもよい。

【0186】

図19を参照すると、一実施形態では、係数ブロック(1910)は、第1の領域(1920)と、第1の領域(1920)の外側にある第2の領域(1930)とを含む。二次変換(例えば、NSST又はSBT)が第1の領域(1920)に適用され、部分的な係数(例えば、左上の領域0)のみが保持されるゼロアウト方法を使用する場合、二次変換が適用されない第2の領域(1930)もゼロアウト領域と考えられてもよい。図19に示す例では、領域0と第2の領域(1930)との間にある領域1は、二次変換で使用されるゼロアウト方法によるゼロアウト領域である。したがって、結合したゼロアウト領域は、領域1及び第2の領域(1930)を含む。一例では、係数ブロック(1910)は、1つの係数ユニット(例えば、領域0)のみを含み、したがって、係数ブロック(1910)内の係数ユニット(例えば、領域0)のみがエントロピー符号化されるべきである。

20

【0187】

一実施形態では、係数ブロック(1910)内の係数ユニットのサイズ及び位置は、ゼロアウト方法を含む二次変換に基づいて推定されてもよい。一例では、二次変換が $8 \times 8$ の二次変換である場合、係数ユニットは、係数ブロック(1910)の左上の $8 \times 8$ の領域(1920)である。一例では、二次変換が、領域0内の第1の16個の係数を保持するゼロアウト方法を含む第1の領域(1920)に対する $8 \times 8$ の二次変換である場合、係数ユニットは、第1の領域(1920)の左上の $4 \times 4$ の領域0である。

30

【0188】

一例では、二次変換は、第1の領域(1920)(例えば、係数ブロック(1910)内の左上の $8 \times 8$ の領域)に適用される $8 \times 8$ の二次変換であり、第2の領域(1930)は、左上の $8 \times 8$ の領域(1920)の外側である。二次変換において16個の係数のみが保持される場合、結合したゼロアウト領域は左上の $4 \times 4$ の領域0の外側である。

40

【0189】

一例では、係数ブロックは、 $4 \times 4$ の第1の領域と、第1の領域の外側にある第2の領域とを含む。二次変換が第1の領域に適用され、部分的な係数のみが保持されるゼロアウト方法を使用する場合、二次変換が適用されない第2の領域は、ゼロアウト領域と考えられてもよい。

【0190】

図20は、本開示の一実施形態によるプロセス(2000)の概略を示すフローチャートを示す。プロセス(2000)は、イントラモードで符号化されたブロックの復元に使用でき、それにより、復元中のブロックについて予測ブロックを生成する。いくつかの例では、プロセ

50

ス(2000)は、インターモードで符号化されたブロックの復元に使用できる。様々な実施形態では、プロセス(2000)は、端末デバイス(310)、(320)、(330)及び(340)内の処理回路、ビデオエンコーダ(403)の機能を実行する処理回路、ビデオデコーダ(410)の機能を実行する処理回路、ビデオデコーダ(510)の機能を実行する処理回路、ビデオエンコーダ(603)の機能を実行する処理回路等のような処理回路によって実行される。いくつかの実施形態では、プロセス(2000)は、ソフトウェア命令で実装され、したがって、処理回路がソフトウェア命令を実行すると、処理回路は、プロセス(2000)を実行する。プロセスは(S2001)から始まり、(S2010)に進む。

【0191】

(S2010)において、符号化ビデオビットストリームから、係数ブロックのようなTBの符号化情報が復号できる。符号化情報は、二次変換が適用されるTBの領域を示すことができ、当該領域は、二次変換によって計算された変換係数を有する第1の部分領域と、第2の部分領域とを含む。一例では、二次変換はゼロアウト方法を使用し、したがって、第2の領域内の変換係数は二次変換によって計算されず、ゼロに設定される。

10

【0192】

(S2020)において、TBにおける変換係数について、変換係数を決定するために使用される隣接変換係数が第2の部分領域に位置するか否かが決定できる。隣接変換係数が第2の部分領域に位置すると決定された場合、プロセス(2000)は(S2030)に進む。そうでない場合、プロセス(2000)は(S2040)に進む。

【0193】

(S2030)において、隣接変換係数のデフォルト値に従って変換係数が決定できる。いくつかの例では、変換係数の1つ以上のシンタックスエレメントは、隣接変換係数のデフォルト値(例えば、0)に従って決定できる。シンタックスエレメントは、変換係数が非ゼロ変換係数であるか否かを示すsig\_coeff\_flag、par\_level\_flag、rem\_abs\_gt1\_flag、rem\_abs\_gt2\_flag、変換係数のパリティ、変換係数が2よりも大きいか否か、及び変換係数が4よりも大きいか否かのような係数関連フラグをそれぞれ含んでもよい。

20

【0194】

(S2040)において、変換係数は、隣接変換係数に従って決定できる。いくつかの例では、変換係数の1つ以上のシンタックスエレメントが決定できる。

【0195】

(S2050)において、TBにおけるサンプルは、サンプルについての変換係数に基づいて復元できる。

30

【0196】

プロセス(2000)は、適切に適合されてもよい。例えば、1つ以上のステップが変更、省略又は組合されてもよい。プロセス(2000)が実行される順序も変更されてもよい。

【0197】

更なるステップも追加されてもよい。例えば、TBにおける変換係数は、第1のCG内の複数の変換係数のうち1つでもよく、第1のCGについての第1のCGフラグは、変換係数のうち少なくとも1つが非ゼロ変換係数であるか否かを示す。変換係数を含む第2のCGは、前にエントロピー復号されており、第1のCGの隣接するCGである。プロセス(2000)は、第2のCGの位置を決定するステップを含んでもよい。第2のCGが第2の部分領域に位置すると決定された場合、プロセス(2000)は、第2のCGフラグのデフォルト値に基づいて第1のCGフラグを決定するステップを含んでもよい。代替として、第2のCGの一部が第2の部分領域に位置し、第2のCGの他の部分が少なくとも1つの非ゼロ係数である場合、プロセス(2000)は、少なくとも1つの非ゼロ係数に基づいて第1のCGフラグを決定するステップを含んでもよい。

40

【0198】

いくつかの例では、更なるステップは、変換係数が第2の部分領域に位置するか否かを決定するステップを含んでもよい。変換係数が第2の部分領域に位置すると決定された場合、変換係数は伝達されず、ゼロであると決定されてもよい。

50

## 【 0 1 9 9 】

図 2 1 は、本開示の一実施形態によるプロセス(2100)の概略を示すフローチャートを示す。プロセス(2100)は、イントラモードで符号化されたブロックの復元に使用でき、それにより、復元中のブロックについて予測ブロックを生成する。いくつかの例では、プロセス(2100)は、インターモードで符号化されたブロックの復元に使用できる。様々な実施形態では、プロセス(2100)は、端末デバイス(310)、(320)、(330)及び(340)内の処理回路、ビデオエンコーダ(403)の機能を実行する処理回路、ビデオデコーダ(410)の機能を実行する処理回路、ビデオデコーダ(510)の機能を実行する処理回路、ビデオエンコーダ(603)の機能を実行する処理回路等のような処理回路によって実行される。いくつかの実施形態では、プロセス(2100)は、ソフトウェア命令で実装され、したがって、処理回路がソフトウェア命令を実行すると、処理回路は、プロセス(2100)を実行する。プロセスは(S2101)から始まり、(S2110)に進む。

10

## 【 0 2 0 0 】

(S2110)において、符号化ビデオビットストリームから変換ブロック(TB)の符号化情報が復号できる。

## 【 0 2 0 1 】

(S2120)において、二次変換がTBの第1の領域に対して実行されるか否かが、符号化情報に基づいて決定でき、第1の領域は、二次変換によって計算された変換係数を有する第1の部分領域と、第2の部分領域とを含む。一例では、二次変換はゼロアウト方法を使用し、したがって、第2の領域内の変換係数は二次変換によって計算されず、ゼロに設定される。二次変換が実行されると決定された場合、プロセス(2100)は(S2130)に進む。そうでない場合、プロセス(2100)は(S2199)に進む。

20

## 【 0 2 0 2 】

(S2130)において、TBにおける第2の領域内の変換係数がゼロであると決定され、第2の領域は第1の領域の外側である。

## 【 0 2 0 3 】

プロセス(2000)は、適切に適合されてもよい。例えば、1つ以上のステップが変更、省略又は組合されてもよい。更なるステップも追加されてもよい。例えば、TBにおける係数ユニットのサイズ及び位置は、第1の領域に基づいて決定され、係数ユニットの外側の変換係数はゼロである。プロセス(2000)が実行される順序も変更されてもよい。

30

## 【 0 2 0 4 】

上記の技術は、コンピュータ読み取り可能命令を使用してコンピュータソフトウェアとして実装され、1つ以上のコンピュータ読み取り可能媒体に物理的に記憶されてもよい。例えば、図 2 2 は、開示の対象物の特定の実施形態を実装するのに適したコンピュータシステム(2200)を示す。

## 【 0 2 0 5 】

コンピュータソフトウェアは、いずれかの適切な機械コード又はコンピュータ言語を使用して符号化されてもよく、当該機械コード又はコンピュータ言語は、命令を含むコードを生成するために、アセンブリ、コンパイル、リンク又は類似のメカニズムを受けてもよく、当該命令は、1つ以上のコンピュータ中央処理装置(CPU, central processing unit)、グラフィックス処理ユニット(GPU, Graphics Processing Unit)等によって、直接的に或いはインタープリタ、マイクロコード実行等を通じて実行されてもよい。

40

## 【 0 2 0 6 】

命令は、例えば、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、スマートフォン、ゲームデバイス、モノのインターネットのデバイス等を含む様々なタイプのコンピュータ又はその構成要素上で実行されてもよい。

## 【 0 2 0 7 】

コンピュータシステム(2200)について図 2 2 に示される構成要素は、本質的に例示的なものであり、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範囲又は機能に関する如何なる限定も示唆することを意図するものではない。また、構成要素の構成も

50

、コンピュータシステム(2200)の例示的な実施形態に示される構成要素のいずれか1つ又は組み合わせに関する如何なる依存性又は要件も有するものとして解釈されるべきではない。

【0208】

コンピュータシステム(2200)は、特定のヒューマンインタフェース入力デバイスを含んでもよい。このようなヒューマンインタフェース入力デバイスは、例えば、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等)、オーディオ入力(音声、拍手等)、視覚入力(ジェスチャ等)、嗅覚入力(図示せず)を通じて、1人以上の人間のユーザによる入力にตอบสนองしてもよい。また、ヒューマンインタフェースデバイスは、オーディオ(例えば、会話、音楽、周辺音)、画像(スキャンされた画像、静止画カメラから取得された写真画像等)、

10

【0209】

入力ヒューマンインタフェースデバイスは、キーボード(2201)、マウス(2202)、トラックパッド(2203)、タッチ画面(2210)、データグローブ(図示せず)、ジョイスティック(2205)、マイクロフォン(2206)、スキャナ(2207)、カメラ(2208)のうち1つ以上を含んでもよい。

【0210】

また、コンピュータシステム(2200)は、特定のヒューマンインタフェース出力デバイスを含んでもよい。このようなヒューマンインタフェース出力デバイスは、例えば、触覚出力、音、光及び嗅覚/味覚を通じて、1人以上の人間のユーザの感覚を刺激してもよい。このようなヒューマンインタフェース出力デバイスは、触覚出力デバイス(例えば、タッチ画面(2210)、データグローブ(図示せず)又はジョイスティック(2205)による触覚フィードバック、ただし、入力デバイスとして機能しない触覚フィードバックデバイスが存在してもよい)と、オーディオ出力デバイス(スピーカ(2209)、ヘッドフォン(図示せず)等)と、視覚出力デバイス(それぞれがタッチ画面入力機能を有しても有さなくてもよく、それぞれが触覚フィードバック機能を有しても有さなくてもよく、いくつかは2次元視覚出力又は立体出力のような手段を通じた3次元以上の出力を出力可能でもよいCRT画面、LCD画面、プラズマ画面、OLED画面を含む画面(2210)、仮想現実メガネ(図示せず)、ホログラフィックディスプレイ及びスモークタンク(図示せず))と、プリンタ(図示せず)とを含んでも

20

30

【0211】

また、コンピュータシステム(2200)は、CD/DVD又は同様の媒体(2221)を有するCD/DVD ROM/RW(2220)を含む光媒体のような人間がアクセス可能な記憶デバイス及び関連する媒体、サムドライブ(2222)、取り外し可能ハードドライブ又はソリッドステートドライブ(2223)、テープ及びフロッピーディスク(図示せず)のようなレガシー磁気媒体、セキュリティ dongle(図示せず)のような特殊なROM/ASIC/PLDに基づくデバイス等を含んでもよい。

【0212】

また、当業者は、ここに開示の対象物に関連して使用される用語「コンピュータ読み取り可能媒体」が伝送媒体、搬送波又は他の非一時的な信号を含まないことを理解すべきである。

40

【0213】

また、コンピュータシステム(2200)は、1つ以上の通信ネットワークへのインタフェースを含んでもよい。ネットワークは、例えば、無線、有線、光でもよい。ネットワークは、ローカル、広域、メトロポリタン、車両及び産業、リアルタイム、遅延耐性等でもよい。ネットワークの例は、イーサネット、無線LAN、セルラネットワーク(GSM、3G、4G、5G、LTE等を含む)、TV有線又は無線広域デジタルネットワーク(ケーブルTV、衛星TV、及び地上放送TVを含む)、車両及び産業(CANBusを含む)等を含む。特定のネットワークは、一般的に、特定の汎用データポート又は周辺バス(2249)に取り付けられる外部ネットワ

50

ークインタフェースアダプタ(例えば、コンピュータシステム(2200)のUSBポート等)を必要とし、他のネットワークインタフェースアダプタは、一般的に、以下に説明するシステムバス(例えば、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又はスマートフォンコンピュータシステムへのセルラネットワーク)に取り付けられることによって、コンピュータシステム(2200)のコアに統合される。これらのネットワークのいずれかを使用して、コンピュータシステム(2200)は、他のエンティティと通信することができる。このような通信は、一方向の受信のみ(例えば、放送TV)、一方向の送信のみ(例えば、特定のCANbusデバイスへのCANbus)でもよく、或いは、例えば、ローカル又は広域デジタルネットワークを使用する他のコンピュータシステムへの双方向でもよい。特定のプロトコル及びプロトコルスタックは、上記のようなネットワーク及びネットワークインタフェースのそれぞれにおいて使用されてもよい。

10

**【0214】**

上記のヒューマンインタフェースデバイス、人間がアクセス可能な記憶デバイス及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(2200)のコア(2240)に取り付けられてもよい。

**【0215】**

コア(2240)は、1つ以上の中央処理装置(CPU)(2241)、グラフィックス処理ユニット(GPU)(2242)、フィールドプログラマブルゲートアレイ(FPGA, Field Programmable Gate Area)(2243)の形式の特殊なプログラム可能処理ユニット、特定のタスク用のハードウェアアクセラレータ(2244)等を含んでもよい。これらのデバイスは、読み取り専用メモリ(ROM)(2245)、ランダムアクセスメモリ(2246)、内部大容量記憶装置(内部のユーザアクセス不可能なハードドライブ、SSD等)(2247)と共に、システムバス(2248)を通じて接続されてもよい。いくつかのコンピュータシステムでは、システムバス(2248)は、更なるCPU、GPU等による拡張を可能にするために、1つ以上の物理プラグの形式でアクセス可能でもよい。周辺デバイスは、コアのシステムバス(2248)に直接取り付けられてもよく、或いは、周辺バス(2249)を通じて取り付けられてもよい。周辺バスのアーキテクチャは、PCI、USB等を含む。

20

**【0216】**

CPU(2241)、GPU(2242)、FPGA(2243)及びアクセラレータ(2244)は特定の命令を実行してもよく、当該特定の命令は、組み合わせによって上記のコンピュータコードを構成してもよい。当該コンピュータコードは、ROM(2245)又はRAM(2246)に記憶されてもよい。また、一時的なデータは、RAM(2246)に記憶されてもよいが、永続的なデータは、例えば、内部大容量記憶装置(2247)に記憶されてもよい。1つ以上のCPU(2241)、GPU(2242)、大容量記憶装置(2247)、ROM(2245)、RAM(2246)等と密接に関連してもよいキャッシュメモリを使用することによって、メモリデバイスのいずれかへの高速記憶及び検索が可能になってもよい。

30

**【0217】**

コンピュータ読み取り可能媒体は、様々なコンピュータに実装された動作を実行するためのコンピュータコードを有してもよい。媒体及びコンピュータコードは、本開示の目的のために特に設計及び構築されたものでよく、或いは、コンピュータソフトウェア分野における当業者に周知で入手可能なようなものでもよい。

40

**【0218】**

限定ではなく一例として、アーキテクチャ(2200)、具体的には、コア(2240)を有するコンピュータシステムは、1つ以上の有形のコンピュータ読み取り可能媒体に具現されたソフトウェアを実行するプロセッサ(CPU、GPU、FPGA、アクセラレータ等を含む)の結果として機能を提供できる。このようなコンピュータ読み取り可能媒体は、コア内部の大容量記憶装置(2247)又はROM(2245)のような非一時的な性質のコア(2240)の特定の記憶装置と同様に、上記のようなユーザがアクセス可能な大容量記憶装置に関連する媒体でもよい。本開示の様々な実施形態を実装するソフトウェアは、このようなデバイスに記憶されてコア(2240)によって実行されてもよい。コンピュータ読み取り可能媒体は、特定の

50

ニーズに従って、1つ以上のメモリデバイス又はチップを含んでもよい。ソフトウェアは、コア(2240)、具体的には、その中のプロセッサ(CPU、GPU、FPGA等を含む)に、RAM(2246)に記憶されたデータ構造を定義し、ソフトウェアによって定義された処理に従ってこのようなデータ構造を修正することを含む、本明細書に記載の特定の処理又は特定の処理の特定の部分を実行させてもよい。さらに或いは代替として、コンピュータシステムは、回路(例えば、アクセラレータ(2244))内に配線されたロジック又は他の方法で具現されたロジックの結果として、機能を提供してもよく、当該回路は、本明細書に記載の特定の処理又は特定の処理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或いはソフトウェアと共に動作してもよい。ソフトウェアへの言及は、ロジックを含み、必要に応じて、その逆も可能である。コンピュータ読み取り可能媒体への言及は、必要に応じて、実行するためのソフトウェアを記憶する回路(集積回路(IC)等)、実行するためのロジックを具現する回路又はこれらの双方を含んでもよい。本開示は、ハードウェア及びソフトウェアのいずれかの適切な組み合わせを含む。

10

## 【 0 2 1 9 】

&lt; 付録A:略語 &gt;

JEM: joint exploration model

VVC: versatile video coding

BMS: benchmark set

MV: Motion Vector

HEVC: High Efficiency Video Coding

20

SEI: Supplementary Enhancement Information

VUI: Video Usability Information

GOP: Group of Picture

TU: Transform Unit

PU: Prediction Unit

CTU: Coding Tree Unit

CTB: Coding Tree Block

PB: Prediction Block

HRD: Hypothetical Reference Decoder

SNR: Signal Noise Ratio

30

CPU: Central Processing Unit

GPU: Graphics Processing Unit

CRT: Cathode Ray Tube

LCD: Liquid-Crystal Display

OLED: Organic Light-Emitting Diode

CD: Compact Disc

DVD: Digital Video Disc

ROM: Read-Only Memory

RAM: Random Access Memory

ASIC: Application-Specific Integrated Circuit

40

PLD: Programmable Logic Device

LAN: Local Area Network

GSM: Global System for Mobile communications

LTE: Long-Term Evolution

CANBus: Controller Area Network Bus

USB: Universal Serial Bus

PCI: Peripheral Component Interconnect

FPGA: Field Programmable Gate Area

SSD: solid-state drive

IC: Integrated Circuit

50

CU: Coding Unit

【0220】

本開示は、いくつかの例示的な実施形態を記載しているが、本開示の範囲内に入る変更、置換及び様々な代替の等価物が存在する。したがって、当業者は、本明細書に明示的に図示又は記載されていないが、本開示の原理を具現し、したがって、本開示の真意及び範囲内にある多数のシステム及び方法を考案することができることが認識される。

【0221】

<付録I>

4×4変換

【表3】

10

$$\{64, 64, 64, 64\}$$

$$\{83, 36, -36, -83\}$$

$$\{64, -64, -64, 64\}$$

$$\{36, -83, 83, -36\}$$

20

【0222】

8×8変換

【表4】

$$\{64, 64, 64, 64, 64, 64, 64, 64\}$$

$$\{89, 75, 50, 18, -18, -50, -75, -89\}$$

$$\{83, 36, -36, -83, -83, -36, 36, 83\}$$

$$\{75, -18, -89, -50, 50, 89, 18, -75\}$$

$$\{64, -64, -64, 64, 64, -64, -64, 64\}$$

$$\{50, -89, 18, 75, -75, -18, 89, -50\}$$

$$\{36, -83, 83, -36, -36, 83, -83, 36\}$$

$$\{18, -50, 75, -89, 89, -75, 50, -18\}$$

30

【0223】

16×16変換

40

【表 5】

```

{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}

```

10

【 0 2 2 4】

32 × 32 変換

【表 6】

```

{64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64}
{90 90 88 85 82 78 73 67 61 54 46 38 31 22 13 4 -4-13-22-31-38-46-54-61-67-73-78-82-85-88-90-90}
{90 87 80 70 57 43 25 9 -9-25-43-57-70-80-87-90-90-87-80-70-57-43-25 -9 9 25 43 57 70 80 87 90}
{90 82 67 46 22 -4-31-54-73-85-90-88-78-61-38-13 13 38 61 78 88 90 85 73 54 31 4-22-46-67-82-90}
{89 75 50 18-18-50-75-89-89-75-50-18 18 50 75 89 89 75 50 18-18-50-75-89-89-75-50-18 18 50 75 89}
{88 67 31-13-54-82-90-78-46 -4 38 73 90 85 61 22-22-61-85-90-73-38 4 46 78 90 82 54 13-31-67-88}
{87 57 9-43-80-90-70-25 25 70 90 80 43 -9-57-87-87-57 -9 43 80 90 70 25-25-70-90-80-43 9 57 87}
{85 46-13-67-90-73-22 38 82 88 54 -4-61-90-78-31 31 78 90 61 4-54-88-82-38 22 73 90 67 13-46-85}
{83 36-36-83-83-36 36 83 83 36-36-83-83-36 36 83 83 36-36-83-83-36 36 83 83 36-36-83-83-36 36 83}
{82 22-54-90-61 13 78 85 31-46-90-67 4 73 88 38-38-88-73 -4 67 90 46-31-85-78-13 61 90 54-22-82}
{80 9-70-87-25 57 90 43-43-90-57 25 87 70 -9-80-80 -9 70 87 25-57-90-43 43 90 57-25-87-70 9 80}
{78 -4-82-73 13 85 67-22-88-61 31 90 54-38-90-46 46 90 38-54-90-31 61 88 22-67-85-13 73 82 4-78}
{75-18-89-50 50 89 18-75-75 18 89 50-50-89-18 75 75-18-89-50 50 89 18-75-75 18 89 50-50-89-18 75}
{73-31-90-22 78 67-38-90-13 82 61-46-88 -4 85 54-54-85 4 88 46-61-82 13 90 38-67-78 22 90 31-73}
{70-43-87 9 90 25-80-57 57 80-25-90 -9 87 43-70-70 43 87 -9-90-25 80 57-57-80 25 90 9-87-43 70}
{67-54-78 38 85-22-90 4 90 13-88-31 82 46-73-61 61 73-46-82 31 88-13-90 -4 90 22-85-38 78 54-67}
{64-64-64 64 64-64-64 64 64-64-64 64 64-64-64 64 64-64-64 64 64-64-64 64 64-64-64 64 64-64-64 64}
{61-73-46 82 31-88-13 90 -4-90 22 85-38-78 54 67-67-54 78 38-85-22 90 4-90 13 88-31-82 46 73-61}
{57-80-25 90 -9-87 43 70-70-43 87 9-90 25 80-57-57 80 25-90 9 87-43-70 70 43-87 -9 90-25-80 57}
{54-85 -4 88-46-61 82 13-90 38 67-78-22 90-31-73 73 31-90 22 78-67-38 90-13-82 61 46-88 4 85-54}
{50-89 18 75-75-18 89-50-50 89-18-75 75 18-89 50 50-89 18 75-75-18 89-50-50 89-18-75 75 18-89 50}
{46-90 38 54-90 31 61-88 22 67-85 13 73-82 4 78-78 -4 82-73-13 85-67-22 88-61-31 90-54-38 90-46}
{43-90 57 25-87 70 9-80 80 -9-70 87-25-57 90-43-43 90-57-25 87-70 -9 80-80 9 70-87 25 57-90 43}
{38-88 73 -4-67 90-46-31 85-78 13 61-90 54 22-82 82-22-54 90-61-13 78-85 31 46-90 67 4-73 88-38}
{36-83 83-36-36 83-83 36 36-83 83-36-36 83-83 36 36-83 83-36-36 83-83 36 36-83 83-36-36 83-83 36}
{31-78 90-61 4 54-88 82-38-22 73-90 67-13-46 85-85 46 13-67 90-73 22 38-82 88-54 -4 61-90 78-31}
{25-70 90-80 43 9-57 87-87 57 -9-43 80-90 70-25-25 70-90 80-43 -9 57-87 87-57 9 43-80 90-70 25}
{22-61 85-90 73-38 -4 46-78 90-82 54-13-31 67-88 88-67 31 13-54 82-90 78-46 4 38-73 90-85 61-22}
{18-50 75-89 89-75 50-18-18 50-75 89-89 75-50 18 18-50 75-89 89-75 50-18-18 50-75 89-89 75-50 18}
{13-38 61-78 88-90 85-73 54-31 4 22-46 67-82 90-90 82-67 46-22 -4 31-54 73-85 90-88 78-61 38-13}
{ 9-25 43-57 70-80 87-90 90-87 80-70 57-43 25 -9 -9 25-43 57-70 80-87 90-90 87-80 70-57 43-25 9}
{ 4-13 22-31 38-46 54-61 67-73 78-82 85-88 90-90 90-90 88-85 82-78 73-67 61-54 46-38 31-22 13 -4}

```

20

30

40

【 0 2 2 5】

< 付録 II >

64ポイントのDCT-2コア



-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 }

{ 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 }

10

{ 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 }

20

{ 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 }

30

{ bi, bp, bw, cd, ck, -ce, -bx, -bq, -bj, -bh, -bo, -bv, -cc, -cj, cf, by, br, bk, bg,

40

50

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

10

{ 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 }

20

{ 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 }

30

{ 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 }

40

50

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

10

{ 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, -ap, -aw, -bd, az, as }

20

{ bm, cb, -cf, -bq, -bi, -bx, cj, bu, bf, bt, ci, -by, -bj, -bp, -ce, 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 }

30

{ ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab, ab, ac, -ac, -ab, -ab, -ac, ac, ab }

40

{ bn, ce, -ca, -bj, -br, -ci, bw, bf, bv, -cj, -bs, -bi, -bz, cf, bo, 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 }

10

{ 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, -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 }

{ 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, -bu, -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 }

20

{ 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, -ak, ao, aj, 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, -ak, ao, aj }

30

{ 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, -

40

50

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

10

{ 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, 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 }

20

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

{ 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, -cd, bt, bp, -ch, -bi, -ca, bw, bm, -ck, -bl, -bx, bz, bj, ci, -bo, -bu, cc, bg, cf, -br }

30

{ 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, -av, bb, ap, bc, -au, -aw,

40

50

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 }

{ 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, -bt, -br, cd, bh, ci, -bm, -by, bw, bo, -cg, -bf, -cf, bp,  
bv, -bz, -bl, cj, bi, cc, -bs }

10

{ 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 }

20

{ 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, -bw, -bq, ce, 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 }

30

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

{ bu, -bw, -bs, by, bq, -ca, -bo, cc, bm, -ce, -bk, cg, bi, -ci, -bg, ck, bf, cj, -bh,

40

50

-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, aa, -aa, -aa, aa }

10

{ 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 }

20

{ 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, bb, at, -az, -av, ax }

30

{ 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 }

40

50

{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 }

{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 }

10

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

20

{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 }

30

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

40

50

{ 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 }

10

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

{ 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, 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 }

20

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

30

{ 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, -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, -

40

50

bn, bp, -ci, -bu, bi, -cb }

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

10

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

20

{ ac, -ab, ab, -ac, -ac, ab, -ab, ac, ac, -ab, ab, -ac, -ac, ab, -ab, ac, ac, -ab,  
ab, -ac, -ac, ab, -ab, ac, ac, -ab, ab, -ac, -ac, ab, -ab, ac, ac, -ab, ab, -ac, -ac,  
ab, -ab, ac, ac, -ab, ab, -ac, -ac, ab, -ab, ac, ac, -ab, ab, -ac, -ac, ab, -ab, ac,  
ac, -ab, ab, -ac, -ac, ab, -ab, ac }

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

30

{ 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, -

40

50

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 }

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

10

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

20

{ 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, av, -aq, as, -ax, bc }

30

{ cg, -bx, bo, -bf, bn, -bw, cf, ch, -by, bp, -bg, bm, -bv, ce, ci, -bz, bq, -bh,

40

50

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, -bn, 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 }

10

{ 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 }

20

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

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 }

40

50

{ 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 }

{ 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, -bx, ca, -cd, cg, -cj }

10

{ 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, -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, ba, -bb, bc, -bd, be }

20

{ 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 }

30

}

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,b  
y,bz,ca,cb,cc,cd,ce,cf,cg,ch,ci,cj,ck}=

{64,83,36,89,75,50,18,90,87,80,70,57,43,25,9,90,90,88,85,82,78,73,67,61,5  
4,46,38,31,22,13,4,91,90,90,90,88,87,86,84,83,81,79,77,73,71,69,65,62,59,  
56,52,48,44,41,37,33,28,24,20,15,11,7,2}である。

【 0 2 2 6 】

40

< 付録III >

4ポイントのDST-7

【表 8】

$$\{ a, b, c, d \}$$

$$\{ c, c, 0, -c \}$$

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

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

10

20

ここで、 $\{a, b, c, d\} = \{29, 55, 74, 84\}$ である。

【 0 2 2 7】

8ポイントのDST-7

30

40

50

【表 9】

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

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

10

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

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

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

20

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

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

30

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

ここで、 $\{a, b, c, d, e, f, g, h\} = \{17, 32, 46, 60, 71, 78, 85, 86\}$ である。

【 0 2 2 8 】

16ポイントのDST-7

40

50

## 【表 1 0】

{ 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,}

10

{ i, o, f, -c, -l, -l, -c, f, o, i, 0, -i, -o, -f, c, l,}

{ 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,}

20

{ 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,}

30

{ 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,}

40

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}である。

## 【 0 2 2 9】

32ポイントのDST-7

【表 1 1】

{ 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}

{ 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}

10

{ 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}

20

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

{ 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}

30

{ u, w, b, -s, -y, -d, q, A, f, -o, -C, -h, m, E, j, -k, -F, -l, i, D, n, -g, -B, -

40

50

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,}

{ 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,}

10

{ 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,}

20

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

30

{ 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,}

{ 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,}

40

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,}

10

{ 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,}

{ 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,}

20

{ 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}である。

【 0 2 3 0 】

4ポイントのDCT-8

40

50

## 【表 1 2】

$$\{ a, b, c, d, \}$$

$$\{ b, 0, -b, -b, \}$$

10

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

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

20

ここで、 $\{a, b, c, d\} = \{84, 74, 55, 29\}$ である。

## 【0 2 3 1】

8ポイントのDCT-8

## 【表 1 3】

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

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

30

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

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

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

40

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

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

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

50

ここで、 $\{a,b,c,d,e,f,g,h\}=\{86,85,78,71,60,46,32,17\}$ である。

【 0 2 3 2 】

16ポイントのDCT-8

【表 1 4】

{ 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, }

10

{ 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, }

20

{ 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, }

30

{ 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, }

40

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

ここで、 $\{a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p\}=\{90,89,87,83,81,77,72,66,62,56,49,41,33,25,17,9\}$ である。

【 0 2 3 3 】

32ポイントのDCT-8

## 【表 1 5】

{ 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,}	10
{ 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,}	20
{ 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,}	30
{ 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,}	40

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,}

10

{ 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,}

{ 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,}

20

{ 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, t, -g,}

30

{ 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,}

{ 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,}

40

50

{ 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,}

{ 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,}

{ 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}である。

10

20

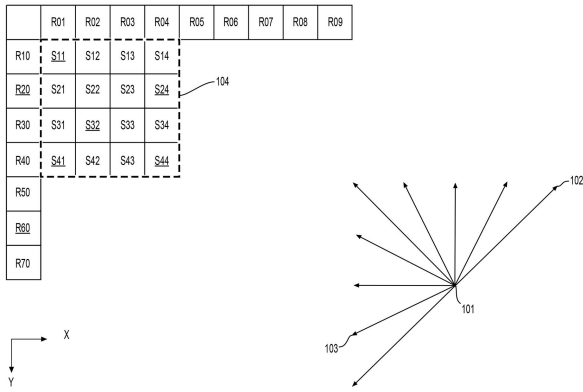
30

40

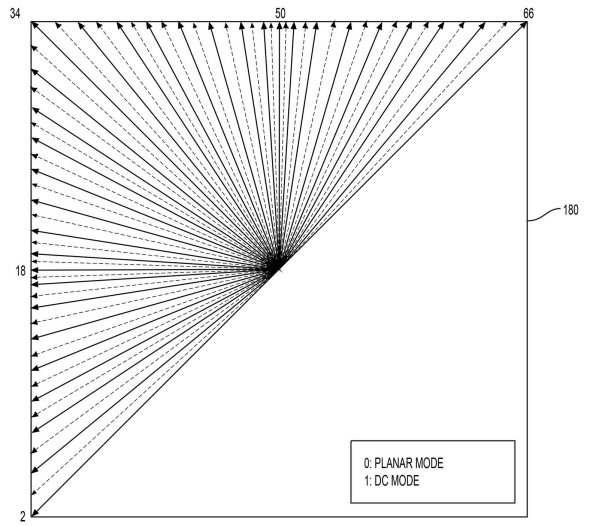
50

【図面】

【図 1 A】

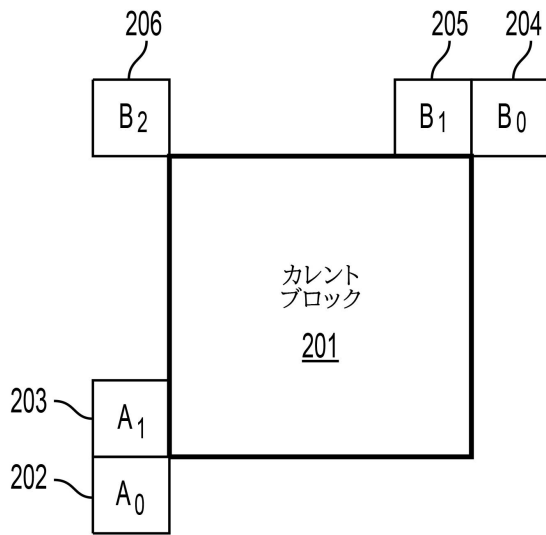


【図 1 B】

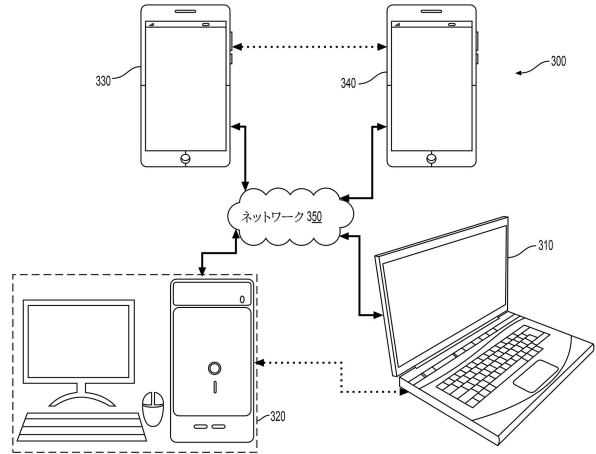


10

【図 2】



【図 3】



20

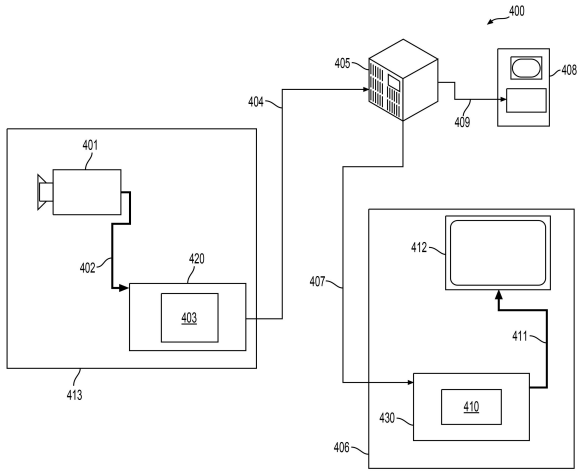
30

関連技術

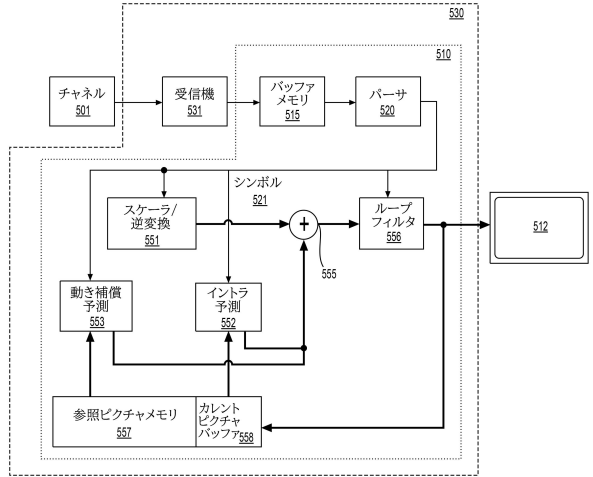
40

50

【図4】

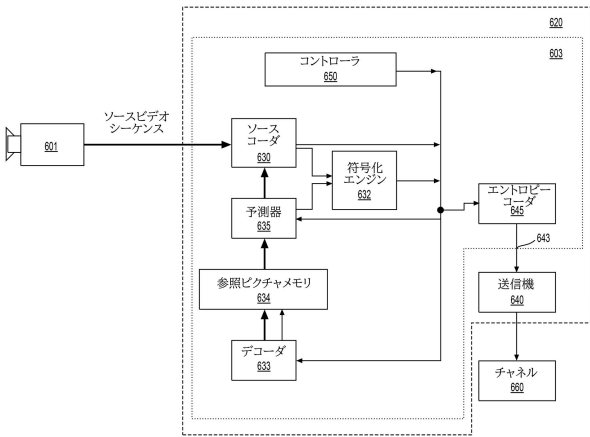


【図5】

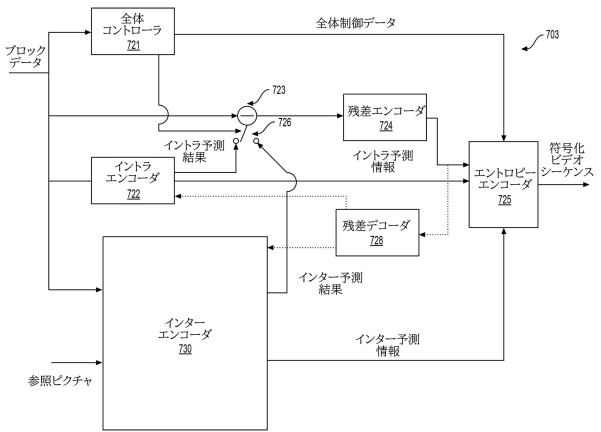


10

【図6】



【図7】



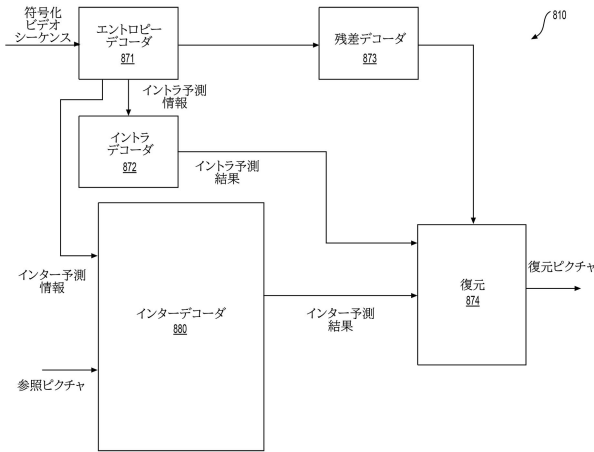
20

30

40

50

【図 8】



【図 9】

	Descriptor
transform_unit(x0, y0, tbWidth, tbHeight, treeType) {	
if (treeType == SINGLE_TREE    treeType == DUAL_TREE_LUMA) {	
tu_cbf_luma[x0][y0]	ae(v)
if (treeType == SINGLE_TREE    treeType == DUAL_TREE_CHROMA) {	
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_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)	
}	

10

【図 10 A】

if (transform_skip_enabled_flag && ( cldx != 0    cu_mts_flag[x0][y0] == 0 ) &&	
(log2TbWidth <= 2) && (log2TbHeight <= 2) )	
transform_skip_flag[x0][y0][cldx]	ae(v)
last_sig_coeff_x_prefix	ae(v)
last_sig_coeff_y_prefix	ae(v)
if (last_sig_coeff_x_prefix > 3)	
last_sig_coeff_x_suffix	ae(v)
if (last_sig_coeff_y_prefix > 3)	
last_sig_coeff_y_suffix	ae(v)
.....	
if (dep_quant_enabled_flag) {	
QState = startQStateSb	
for (n = numSbCoeff - 1; n >= 0; n - -) {	
xC = (xS << log2SbSize) +	
DiagScanOrder[log2SbSize][log2SbSize][n][0]	
yC = (yS << log2SbSize) +	
DiagScanOrder[log2SbSize][log2SbSize][n][1]	
if (sig_coeff_flag[xC][yC])	
TransCoeffLevel[x0][y0][cldx][xC][yC] =	
(2 * AbsLevel[xC][yC] - (QState > 1 ? 1 : 0)) * (1 - 2 * coeff_sign_flag[n])	
QState = QStateTransTable[QState][par_level_flag[n]]	

【図 10 B】

図10Aからの続き

} else {	
sumAbsLevel = 0	
for (n = numSbCoeff - 1; n >= 0; n - -) {	
xC = (xS << log2SbSize) +	
DiagScanOrder[log2SbSize][log2SbSize][n][0]	
yC = (yS << log2SbSize) +	
DiagScanOrder[log2SbSize][log2SbSize][n][1]	
if (sig_coeff_flag[xC][yC]) {	
TransCoeffLevel[x0][y0][cldx][xC][yC] =	
AbsLevel[xC][yC] * (1 - 2 * coeff_sign_flag[n])	
if (signHidden) {	
sumAbsLevel += AbsLevel[xC][yC]	
if (n == firstSigScanPosSb) && (sumAbsLevel > 2) == 1) )	
TransCoeffLevel[x0][y0][cldx][xC][yC] =	
- TransCoeffLevel[x0][y0][cldx][xC][yC]	
}	
}	
}	
}	
if (cu_mts_flag[x0][y0] && ( cldx == 0 ) &&	
! transform_skip_flag[x0][y0][cldx] &&	
(( CuPredMode[x0][y0] == MODE_INTRA && numSigCoeff > 2 )	
( CuPredMode[x0][y0] == MODE_INTER ) ) {	
mts_ids[x0][y0]	ae(v)
}	

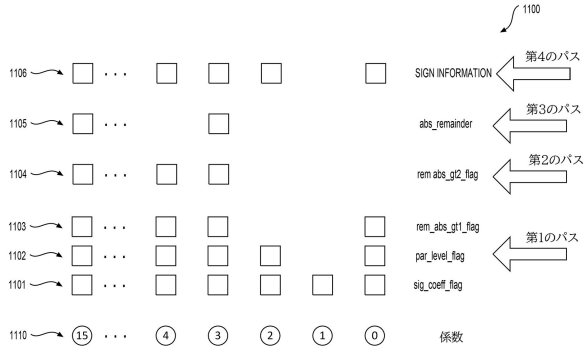
20

30

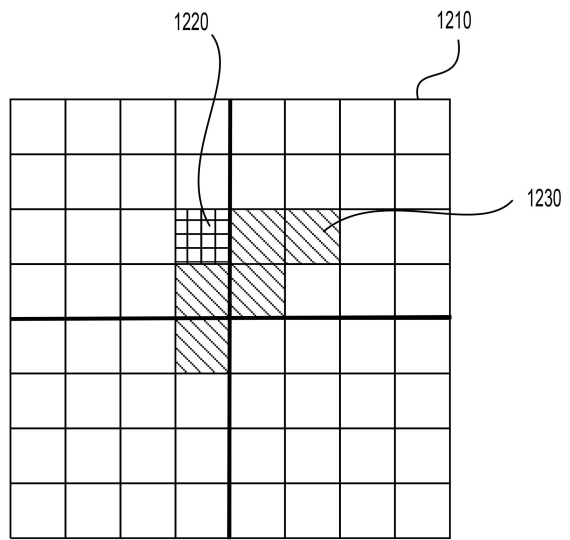
40

50

【図 1 1】



【図 1 2】



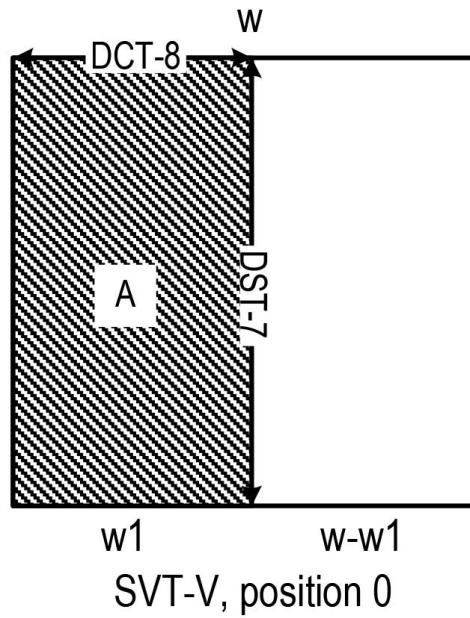
10

【図 1 3】

	Descriptor
residual_coding[x0, y0, log2TbWidth, log2TbHeight, cldx] {	
if ( transform_skip_enabled_flag && ( cldx != 0    tu_mts_flag[x0][y0] == 0 ) && ( log2TbWidth <= 2 ) && ( log2TbHeight <= 2 ) )	
transform_skip_flag[x0][y0][cldx]	ae(v)
last_sig_coeff_x_prefix	ae(v)
last_sig_coeff_y_prefix	ae(v)
if ( last_sig_coeff_x_prefix > 3 )	
last_sig_coeff_x_suffix	ae(v)
if ( last_sig_coeff_y_prefix > 3 )	
last_sig_coeff_y_suffix	ae(v)
.....	
for( i = lastSubBlock; i >= 0; i-- ) {	
.....	
if ( i < lastSubBlock ) && ( i > 0 ) {	
if ( ( tu_mts_flag[x0][y0] == 0    cldx != 0 ) && ( xS << log2SbSize ) < 32 && ( yS << log2SbSize ) < 32   ( tu_mts_flag[x0][y0] == 1 && ( xS << log2SbSize ) < 16 && ( yS << log2SbSize ) < 16 ) )	
coded_sub_block_flag[xS][yS]	ae(v)
inferSbDeSigCoeffFlag = 1	
}	
.....	
}	
if ( tu_mts_flag[x0][y0] && ( cldx == 0 ) )	
mts_idx[x0][y0][cldx]	ae(v)
}	
}	

1301

【図 1 4 A】



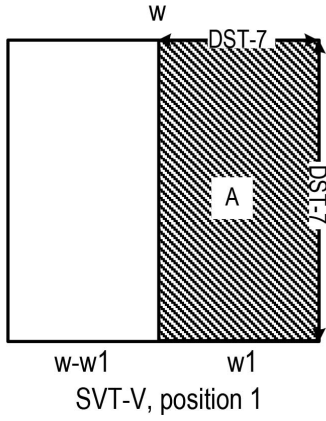
20

30

40

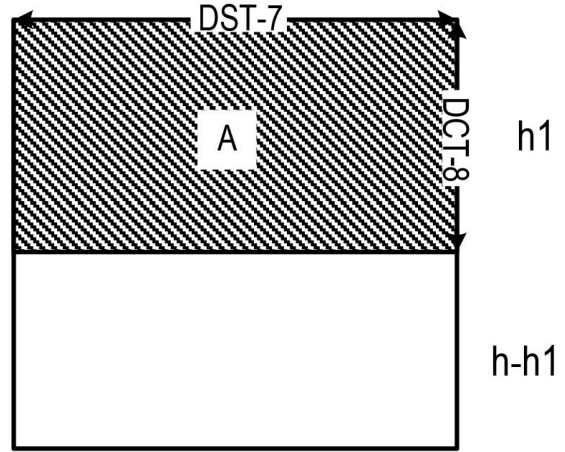
50

【 1 4 B 】



$w_1 = \frac{1}{2} w$  or  $\frac{1}{4} w$

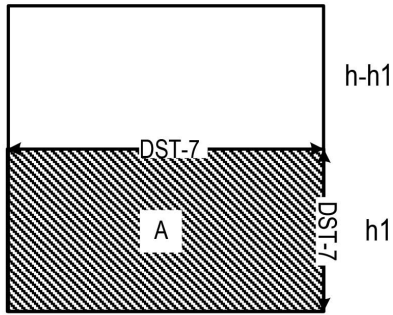
【 1 4 C 】



10

SVT-H, position 0

【 1 4 D 】



$h_1 = \frac{1}{2} h$  or  $\frac{1}{4} h$

SVT-H, position 1

【 1 5 A 】

シーケンスパラメータセットRBSPシンタックス

	Descriptor
seq_parameter_set_rbsp() {	
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()	
}	

A callout '2101' points to the `sps_sbt_enable_flag` field in the table.

20

30

40

50

【 15 B 】

一般的なスライスヘッダのシンタックス

	Descriptor
slice_header() {	
slice_pic_parameter_set_id	ue(v)
slice_address	u(v)
slice_type	ue(v)
if ( slice_type != 1 ) {	
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_11_zero_flag	u(1)
if (slice_temporal_mvp_enabled_flag) {	
if (slice_type == B)	
collocated_from_10_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)
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()	
}	

2102

【 15 C 】

符号化ユニットのシンタックス

	Descriptor
coding_unit(x0, y0, cbWidth, cbHeight, treeType) {	
...	
if (CuPredMode[x0][y0] != MODE_INTRA && cu_skip_flag[x0][y0] == 0)	
cu_cbf	ae(v)
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)
}	
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)
}	
}	
transform_tree(x0, y0, cbWidth, cbHeight, treeType)	
}	
}	

2103

10

20

【 15 D 】

変換ツリーのシンタックス

	Descriptor
transform_tree(x0, y0, tbWidth, tbHeight, treeType) {	
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)	
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	
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)	
}	
} else {	
transform_unit(x0, y0, tbWidth, tbHeight, treeType, 0)	
}	
}	

2104

【 15 E 】

変換ユニットのシンタックス

	Descriptor
transform_unit(x0, y0, tbWidth, tbHeight, treeType, noResi) {	
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)	
}	

2107

30

40

50

【 図 1 5 F 】

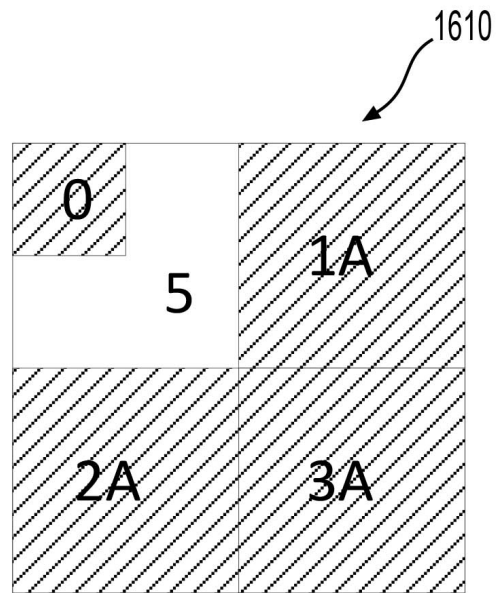
表3

	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

表4

mts_idx[xTbY][yTbY]	CuPredMode[xTbY][yTbY] == MODE_INTRA		CuPredMode[xTbY][yTbY] == MODE_INTER	
	trTypeHor	trTypeVer	trTypeHor	trTypeVer
-1 (inferred)	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

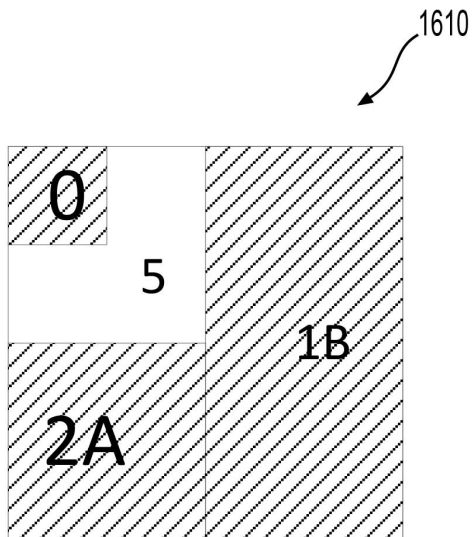
【 図 1 6 A 】



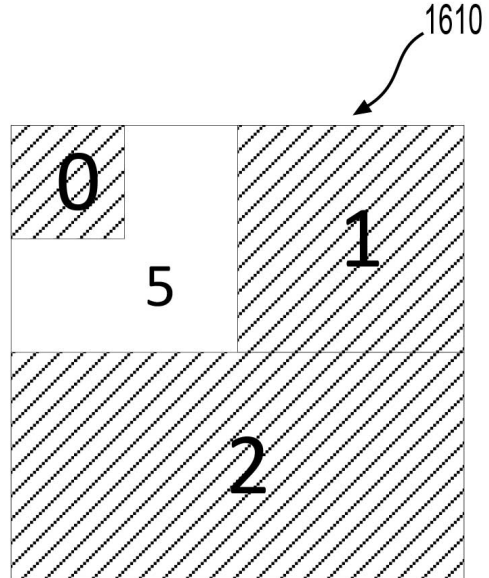
10

20

【 図 1 6 B 】



【 図 1 6 C 】

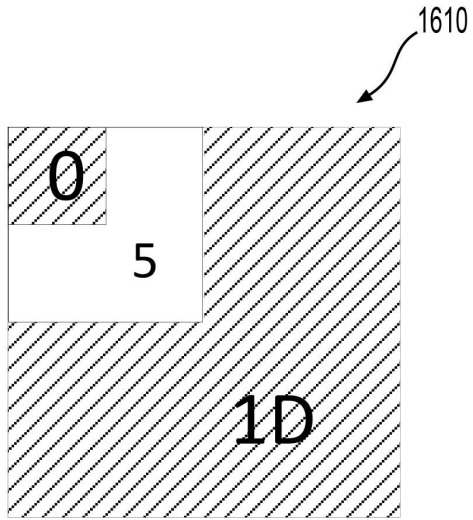


30

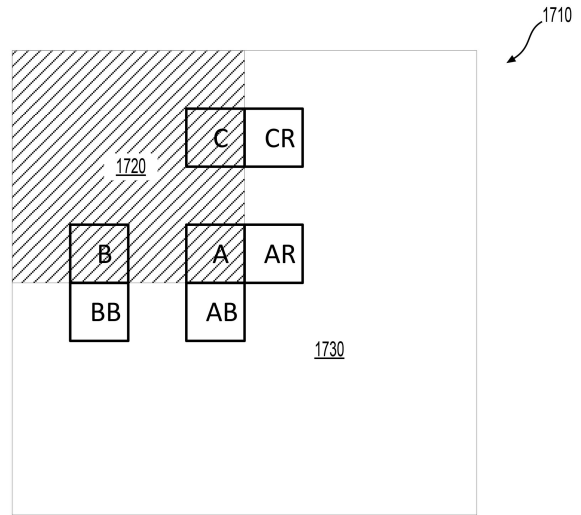
40

50

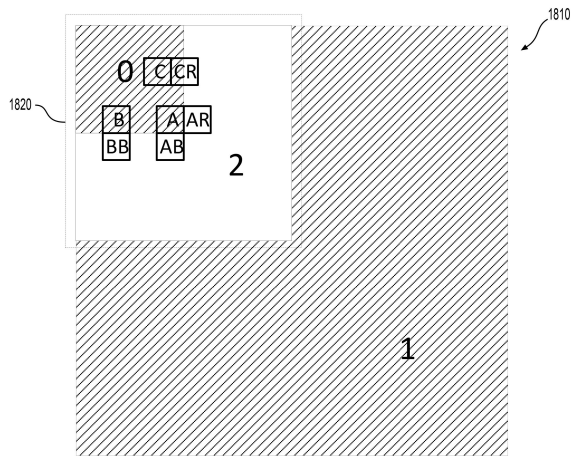
【図 16D】



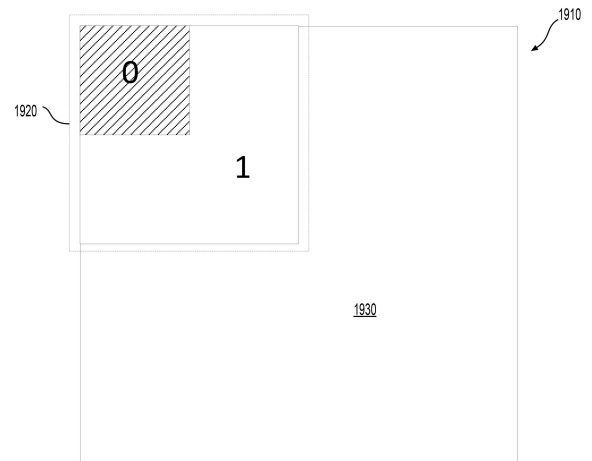
【図 17】



【図 18】



【図 19】



10

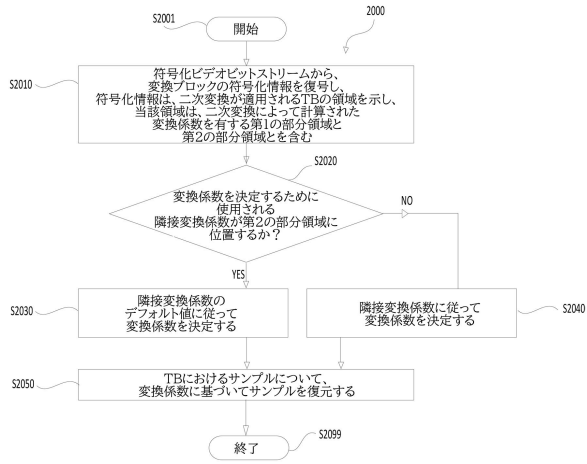
20

30

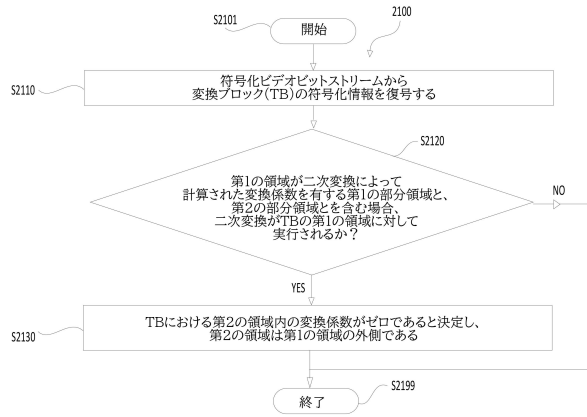
40

50

【図 20】

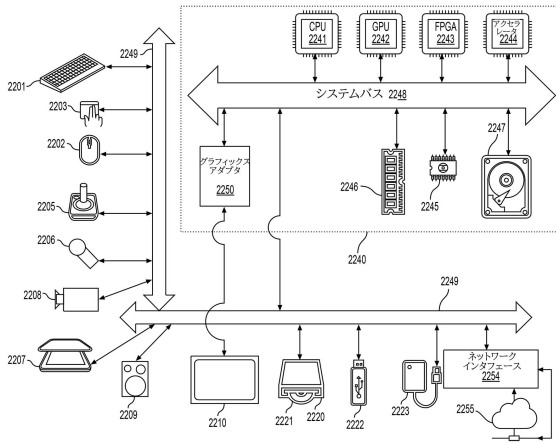


【図 21】



10

【図 22】



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)

エルシー

(72)発明者 リ, シアン

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・ブールバード 2 7 4 7 テンセント アメリカ エルエルシー

(72)発明者 リウ, シャン

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・ブールバード 2 7 4 7 テンセント アメリカ エルエルシー

審査官 久保 光宏

(56)参考文献

Moonmo Koo, et al. , "CE6: Reduced Secondary Transform (RST) (test 6.5.1)" , JVET-M0292 (version 5) , [online], Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 , 2019年01月12日 , Pages 1-14 , [令和4年6月15日検索], インターネット, URL: [http://phenix.it-sudparis.eu/jvet/doc\\_end\\_user/current\\_document.php?id=5099](http://phenix.it-sudparis.eu/jvet/doc_end_user/current_document.php?id=5099) and URL: [http://phenix.it-sudparis.eu/jvet/doc\\_end\\_user/documents/13\\_Marrakech/wg11/JVET-M0292-v5.zip](http://phenix.it-sudparis.eu/jvet/doc_end_user/documents/13_Marrakech/wg11/JVET-M0292-v5.zip) , (See document file "JVET-M0292\_r5.docx" in the zip file "JVET-M0292-v5.zip".)

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161 ~ 185頁, ISBN: 978-4-8443-3468-2.

(58)調査した分野 (Int.Cl. , D B名)

H 0 4 N 1 9 / 0 0 - 1 9 / 9 8

C S D B ( 日本国特許庁 )

学術文献等データベース ( 日本国特許庁 )

I E E E X p l o r e ( I E E E )