

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7414977号
(P7414977)

(45)発行日 令和6年1月16日(2024.1.16)

(24)登録日 令和6年1月5日(2024.1.5)

(51)国際特許分類 F I
H 0 4 N 19/13 (2014.01) H 0 4 N 19/13
H 0 4 N 19/136 (2014.01) H 0 4 N 19/136

請求項の数 13 (全83頁)

(21)出願番号	特願2022-521365(P2022-521365)	(73)特許権者	502032105
(86)(22)出願日	令和2年10月8日(2020.10.8)		エルジー エレクトロニクス インコーポ レイティド
(65)公表番号	特表2022-552298(P2022-552298 A)		L G E L E C T R O N I C S I N C .
(43)公表日	令和4年12月15日(2022.12.15)		大韓民国,ソウル,ヨンドゥンポ-ク, ヨイ-デロ,128
(86)国際出願番号	PCT/KR2020/013758		128, Yeoui-daero, Y eongdeungpo-gu, 07
(87)国際公開番号	WO2021/071297		336 Seoul, Republic of Korea
(87)国際公開日	令和3年4月15日(2021.4.15)	(74)代理人	100099759
審査請求日	令和4年4月7日(2022.4.7)		弁理士 青木 篤
(31)優先権主張番号	62/912,624	(74)代理人	100123582
(32)優先日	令和1年10月8日(2019.10.8)		弁理士 三橋 真二
(33)優先権主張国・地域又は機関	米国(US)	(74)代理人	100165191
(31)優先権主張番号	62/913,678		
(32)優先日	令和1年10月10日(2019.10.10)		
	最終頁に続く		最終頁に続く

(54)【発明の名称】 変換に基づく映像コーディング方法及びその装置

(57)【特許請求の範囲】

【請求項1】

デコーディング装置により実行される映像デコーディング方法において、
ビットストリームからレジデュアル情報を受信するステップと、
前記レジデュアル情報に基づいて現在ブロックに対する変換係数を導出するステップと、
前記変換係数に L F N S T 又は M T S の少なくとも1つを適用してレジデュアルサンプルを導出するステップと、
前記レジデュアルサンプルに基づいて復元ピクチャを生成するステップと、を含み、
前記 L F N S T に対する L F N S T カーネルに関連する L F N S T インデックスの1番目の bin に対するコンテキストインクリメントは、前記現在ブロックのツリータイプがシングルツリーであるか否かに基づいて導出され、前記 L F N S T インデックスの2番目の bin に対するコンテキストインクリメントは、既設定された値として導出され、
前記 M T S に対する M T S カーネルに関連する M T S インデックスをパーシングするか否かは、前記 M T S に対するゼロアウトが実行されるか否かを示す1番目の変数に基づいて決定され、
前記1番目の変数は、コーディングユニットシンタックスにおいて1である初期値に設定され、
少なくとも1つのノンゼロ変換係数が、前記現在ブロック内の左上端 16 x 16 領域以外の領域に存在し、前記現在ブロックのカラーインデックスがルマ成分である場合に基づいて、前記1番目の変数は、レジデュアルコーディングシンタックスにおいて0に変更され、

10

20

前記 L F N S T インデックス及び前記 M T S インデックスは、前記コーディングユニットシンタックスにおいて設定され、

前記 L F N S T インデックスがパーシングされた後、前記 M T S インデックスがパーシングされる、映像デコーディング方法。

【請求項 2】

前記 L F N S T インデックスの 1 番目の b i n 及び 2 番目の b i n に対する前記コンテキストインクリメントを導出するステップと、

前記コンテキストインクリメントに基づいて前記 L F N S T インデックスの b i n ストリングをデコーディングするステップと、

前記 L F N S T インデックスの値を導出するステップと、をさらに含む、請求項 1 に記載の映像デコーディング方法。

10

【請求項 3】

前記現在ブロックのツリータイプが前記シングルツリーであることに基づいて、前記 1 番目の b i n に対する前記コンテキストインクリメントは、第 1 の値として導出され、

前記現在ブロックのツリータイプが前記シングルツリーでないことに基づいて、前記 1 番目の b i n に対する前記コンテキストインクリメントは、第 2 の値として導出される、請求項 2 に記載の映像デコーディング方法。

【請求項 4】

前記 2 番目の b i n に対する前記コンテキストインクリメントは、前記第 1 の値及び前記第 2 の値と異なる第 3 の値として導出される、請求項 3 に記載の映像デコーディング方法。

20

【請求項 5】

L F N S T 変換セットは、二つの L F N S T カーネルを含み、前記 L F N S T インデックスの値は、前記現在ブロックに前記 L F N S T が適用されない場合に関連する 0、前記二つの L F N S T カーネルのうち 1 番目の L F N S T カーネルに関連する 1、前記二つの L F N S T カーネルのうち 2 番目の L F N S T カーネルに関連する 2 のうちのいずれか一つを含む、請求項 2 に記載の映像デコーディング方法。

【請求項 6】

前記 L F N S T インデックスの値は、トランケイテッドユナリコードとして二進化され、0 である値を持つ前記 L F N S T インデックスは「0」として二進化され、1 である値を持つ前記 L F N S T インデックスは「10」として二進化され、2 である値を持つ前記 L F N S T インデックスは「11」として二進化される、請求項 5 に記載の映像デコーディング方法。

30

【請求項 7】

エンコーディング装置により実行される映像エンコーディング方法において、

現在ブロックに対する予測サンプルを導出するステップと、

前記予測サンプルに基づいて前記現在ブロックに対するレジデュアルサンプルを導出するステップと、

前記レジデュアルサンプルに L F N S T 又は M T S の少なくとも 1 つを適用して前記現在ブロックに対する変換係数を導出するステップと、

40

L F N S T カーネルに関連する L F N S T インデックス及び M T S カーネルに関連する M T S インデックスをエンコーディングするステップと、を含み、

前記 L F N S T インデックスの 1 番目の b i n に対するコンテキストインクリメントは、前記現在ブロックのツリータイプがシングルツリーであるか否かに基づいて導出され、前記 L F N S T インデックスの 2 番目の b i n に対するコンテキストインクリメントは、既設定された値として導出され、

前記 M T S インデックスをエンコーディングするか否かは、前記 M T S に対するゼロアウトが実行されるか否かを示す 1 番目の変数に基づいて決定され、

前記 1 番目の変数は、コーディングユニットシンタックスにおいて 1 である初期値に設定され、

50

少なくとも1つのノンゼロ変換係数が、前記現在ブロック内の左上端16×16領域以外の領域に存在し、前記現在ブロックのカラーインデックスがルマ成分である場合に応答して、前記1番目の変数は、レジデュアルコーディングシンタックスにおいて0に変更され、前記LFNSTインデックス及び前記MTSインデックスは、前記コーディングユニットシンタックスにおいて設定され、

前記LFNSTインデックスがエンコーディングされた後、前記MTSインデックスがエンコーディングされる、映像エンコーディング方法。

【請求項8】

前記LFNSTインデックスをエンコーディングするステップは、
前記LFNSTインデックスの値を導出するステップと、
前記LFNSTインデックスの1番目のbin及び2番目のbinに対する前記コンテキストインクリメントを導出するステップと、
前記コンテキストインクリメントに基づいて前記LFNSTインデックスのbinストリングをエンコーディングするステップと、を含む、請求項7に記載の映像エンコーディング方法。

10

【請求項9】

前記現在ブロックのツリータイプが前記シングルツリーであることに基づいて、前記1番目のbinに対する前記コンテキストインクリメントは、第1の値として導出され、
前記現在ブロックのツリータイプが前記シングルツリーでないことに基づいて、前記1番目のbinに対する前記コンテキストインクリメントは、第2の値として導出される、
請求項8に記載の映像エンコーディング方法。

20

【請求項10】

前記2番目のbinに対する前記コンテキストインクリメントは、前記第1の値及び前記第2の値と異なる第3の値として導出される、請求項9に記載の映像エンコーディング方法。

【請求項11】

LFNST変換セットは、二つのLFNSTカーネルを含み、
前記LFNSTインデックスの値は、前記現在ブロックに前記LFNSTが適用されない場合に関連する0、前記二つのLFNSTカーネルのうち1番目のLFNSTカーネルに関連する1、前記二つのLFNSTカーネルのうち2番目のLFNSTカーネルに関連する2のうちのいずれか一つを含む、請求項8に記載の映像エンコーディング方法。

30

【請求項12】

前記LFNSTインデックスの値は、トランケイテッドユナリコードとして二進化され、
0である値を持つ前記LFNSTインデックスは「0」として二進化され、1である値を持つ前記LFNSTインデックスは「10」として二進化され、2である値を持つ前記LFNSTインデックスは「11」として二進化される、請求項11に記載の映像エンコーディング方法。

【請求項13】

映像情報に対するデータを送信する方法において、
LFNSTインデックス及びレジデュアル情報を含む前記映像情報のビットストリームを取得するステップであって、前記ビットストリームは、現在ブロックに対する予測サンプルを導出し、前記予測サンプルに基づいて前記現在ブロックに対するレジデュアルサンプルを導出し、前記レジデュアルサンプルにLFNST又はMTSの少なくとも1つを適用して前記現在ブロックに対する変換係数を導出し、LFNSTカーネルに関連する前記LFNSTインデックス及びMTSカーネルに関連するMTSインデックスをエンコーディングすることにより生成される、ステップと、

40

前記映像情報のビットストリームを含む前記データを送信するステップと、を含み、
前記LFNSTインデックスの1番目のbinに対するコンテキストインクリメントは、前記現在ブロックのツリータイプがシングルツリーであるか否かに基づいて導出され、
前記LFNSTインデックスの2番目のbinに対するコンテキストインクリメントは、

50

既設定された値として導出され、
 前記M T Sインデックスをエンコーディングするか否かは、前記M T Sに対するゼロアウトが実行されるか否かを示す1番目の変数に基づいて決定され、
 前記1番目の変数は、コーディングユニットシンタックスにおいて1である初期値に設定され、
 少なくとも1つのノンゼロ変換係数が、前記現在ブロック内の左上端16×16領域以外の領域に存在し、前記現在ブロックのカラーインデックスがルマ成分である場合に応答して、前記1番目の変数は、レジデュアルコーディングシンタックスにおいて0に変更され、
 前記L F N S Tインデックス及び前記M T Sインデックスは、前記コーディングユニットシンタックスにおいて設定され、
 前記L F N S Tインデックスがエンコーディングされた後、前記M T Sインデックスがエンコーディングされる、データ送信方法。

10

【発明の詳細な説明】

【技術分野】

【0001】

本文書は、映像コーディング技術に関し、より詳しくは、映像コーディングシステムにおいて、変換(transform)に基づく映像コーディング方法及びその装置に関する。

【背景技術】

【0002】

最近、4Kまたは8K以上のUHD(Ultra High Definition)映像/ビデオのような高解像度、高品質の映像/ビデオに対する需要が多様な分野で増加している。映像/ビデオデータが高解像度、高品質になるほど、既存の映像/ビデオデータに比べて相対的に送信される情報量またはビット量が増加するため、既存の有無線広帯域回線のような媒体を利用して映像データを送信し、または既存の格納媒体を利用して映像/ビデオデータを格納する場合、送信費用と格納費用が増加する。

20

【0003】

また、最近、VR(Virtual Reality)、AR(Artificial Reality)コンテンツやホログラムなどの実感メディア(Immersive Media)に対する関心及び需要が増加しており、ゲーム映像のように、現実映像と異なる映像特性を有する映像/ビデオに対する放送が増加している。

30

【0004】

これによって、前記のような多様な特性を有する高解像度、高品質の映像/ビデオの情報を効果的に圧縮して送信または格納し、再生するために、高効率の映像/ビデオ圧縮技術が要求される。

【発明の概要】

【発明が解決しようとする課題】

【0005】

本文書の技術的課題は、映像コーディング効率を上げる方法及び装置を提供することにある。

40

【0006】

本文書の他の技術的課題は、L F N S Tインデックスコーディングの効率を上げる方法及び装置を提供することにある。

【0007】

本文書の他の技術的課題は、L F N S Tインデックスのコーディングを介して2次変換の効率を上げる方法及び装置を提供することにある。

【課題を解決するための手段】

【0008】

本文書の一実施例によると、デコーディング装置により実行される映像デコーディング方法を提供する。前記方法は、変換係数にL F N S TまたはM T Sのうち少なくとも一つ

50

を適用してレジデュアルサンプルを導出するステップ、前記レジデュアルサンプルに基づいて復元ピクチャを生成するステップを含み、前記L F N S Tは、L F N S T変換セット、前記L F N S T変換セットに含まれているL F N S Tカーネル、及び前記L F N S Tカーネルを指示するL F N S Tインデックスに基づいて実行され、前記L F N S Tインデックスに対するシンタックス要素b i nストリングの1番目のb i nは、前記現在ブロックのツリータイプによって異なるコンテキスト情報に基づいて導出され、前記シンタックス要素b i nストリングの2番目のb i nは、既設定されたコンテキスト情報に基づいて導出される。

【0009】

前記L F N S Tインデックスに対するシンタックス要素のコンテキスト情報を導出するステップ、前記コンテキスト情報に基づいて前記L F N S Tインデックスに対するシンタックス要素b i nストリングのb i nをデコーディングするステップ、前記L F N S Tインデックスに対するシンタックス要素の値を導出するステップをさらに含む。

10

【0010】

前記現在ブロックのツリータイプがシングルツリーである場合、前記1番目のb i nは、第1のコンテキスト情報として導出され、前記現在ブロックのツリータイプがシングルツリーでない場合、前記1番目のb i nは、第2のコンテキスト情報として導出される。

【0011】

前記2番目のb i nに対するコンテキスト情報は、前記第1のコンテキスト情報及び前記第2のコンテキスト情報と異なる第3のコンテキスト情報として導出される。

20

【0012】

前記L F N S T変換セットには二つの前記L F N S Tカーネルが含まれ、前記シンタックス要素に対する値は、前記現在ブロックに前記L F N S Tが適用されない場合を指示する0、前記L F N S Tカーネルのうち1番目のL F N S Tカーネルを指示する1、前記L F N S Tカーネルのうち2番目のL F N S Tカーネルを指示する2、のうちいずれか一つを含む。

【0013】

前記シンタックス要素に対する値は、トランケイテッドユナリコードで二進化され、前記シンタックス要素に対する値0は「0」で、前記シンタックス要素に対する値1は「10」で、前記シンタックス要素に対する値2は「11」で、二進化される。

30

【0014】

本文書の一実施例によると、エンコーディング装置により実行される映像エンコーディング方法を提供する。前記方法は、レジデュアルサンプルにL F N S TまたはM T Sのうち少なくとも一つを適用して前記現在ブロックに対する変換係数を導出するステップ、L F N S Tカーネルを指示するL F N S Tインデックス及び量子化されたレジデュアル情報をエンコーディングするステップを含み、前記L F N S Tは、L F N S T変換セット、前記L F N S T変換セットに含まれている前記L F N S Tカーネルに基づいて実行され、前記L F N S Tインデックスに対するシンタックス要素b i nストリングの1番目のb i nは、前記現在ブロックのツリータイプによって異なるコンテキスト情報に基づいて導出され、前記シンタックス要素b i nストリングの2番目のb i nは、既設定されたコンテキスト情報に基づいて導出される。

40

【0015】

本文書の他の一実施例によると、エンコーディング装置により実行された映像エンコーディング方法によって生成されたエンコーディングされた映像情報及びビットストリームが含まれている映像データが格納されたデジタル格納媒体が提供される。

【0016】

本文書の他の一実施例によると、デコーディング装置により前記映像デコーディング方法を実行するようにするエンコーディングされた映像情報及びビットストリームが含まれている映像データが格納されたデジタル格納媒体が提供される。

【発明の効果】

50

【 0 0 1 7 】

本文書によると、全般的な映像 / ビデオ圧縮効率を上げることができる。

【 0 0 1 8 】

本文書によると、L F N S T インデックスコーディングの効率を上げることができる。

【 0 0 1 9 】

本文書によると、L F N S T インデックスのコーディングを介して 2 次変換の効率を上げることができる。

【 0 0 2 0 】

本明細書の具体的な一例を介して得ることができる効果は、以上で羅列された効果に制限されない。例えば、関連した技術分野の通常の知識を有する者 (a p e r s o n h a v i n g o r d i n a r y s k i l l i n t h e r e l a t e d a r t) が本明細書から理解または誘導できる多様な技術的效果が存在できる。これによって、本明細書の具体的な効果は、本明細書に明示的に記載されたことに制限されるものではなく、本明細書の技術的特徴から理解または誘導されることができる多様な効果を含むことができる。

10

【 図面の簡単な説明 】

【 0 0 2 1 】

【 図 1 】本文書が適用されることができるビデオ / 映像コーディングシステムの例を概略的に示す。

【 0 0 2 2 】

【 図 2 】本文書が適用されることができるビデオ / 映像エンコーディング装置の構成を概略的に説明する図である。

20

【 0 0 2 3 】

【 図 3 】本文書が適用されることができるビデオ / 映像デコーディング装置の構成を概略的に説明する図である。

【 0 0 2 4 】

【 図 4 】本文書が適用されるコンテンツストリーミングシステム構造図を例示的に示す。

【 0 0 2 5 】

【 図 5 】本文書の一実施例に係る多重変換技法を概略的に示す。

【 0 0 2 6 】

【 図 6 】 6 5 個予測方向のイントラ方向性モードを例示的に示す。

30

【 0 0 2 7 】

【 図 7 】本文書の一実施例に係る R S T を説明するための図である。

【 0 0 2 8 】

【 図 8 】一例によって順方向 1 次変換の出力データを 1 次元ベクトルに配列する順序を示す図である。

【 0 0 2 9 】

【 図 9 】一例によって順方向 2 次変換の出力データを 2 次元ブロックに配列する順序を示す図である。

【 0 0 3 0 】

【 図 1 0 】本文書の一実施例に係る広角イントラ予測モードを示す図である。

40

【 0 0 3 1 】

【 図 1 1 】 L F N S T が適用されるブロックもようを示す図である。

【 0 0 3 2 】

【 図 1 2 】一例によって順方向 L F N S T の出力データの配列を示す図である。

【 0 0 3 3 】

【 図 1 3 】一例によって 4 × 4 L F N S T が適用されるブロックでのゼロアウトを示す図である。

【 0 0 3 4 】

【 図 1 4 】一例によって 8 × 8 L F N S T が適用されるブロックでのゼロアウトを示す図である。

50

【 0 0 3 5 】

【 図 1 5 】一実施例に係る C A B A C エンコーディングシステムのブロック図を示す図である。

【 0 0 3 6 】

【 図 1 6 】本文書の一実施例に係るビデオデコーディング装置の動作を示す流れ図である。

【 0 0 3 7 】

【 図 1 7 】本文書の一実施例に係るビデオエンコーディング装置の動作を示す流れ図である。

【 発明を実施するための形態 】

【 0 0 3 8 】

本文書は、多様な変更を加えることができ、様々な実施例を有することができ、特定実施例を図面に例示して詳細に説明しようとする。しかし、これは本文書を特定実施例に限定しようとするものではない。本明細書で使用する用語は、単に特定の実施例を説明するために使われたものであって、本文書の技術的思想を限定しようとする意図で使われるものではない。単数の表現は、コンテキスト上明白に異なる意味ではない限り、複数の表現を含む。本明細書において、「含む」または「有する」などの用語は、明細書上に記載された特徴、数字、ステップ、動作、構成要素、部品、またはこれらを組み合わせたものが存在することを指定するものであり、一つまたはそれ以上の他の特徴や数字、ステップ、動作、構成要素、部品、またはこれらを組み合わせたもの等の存在または付加可能性をあらかじめ排除しないと理解されなければならない。

【 0 0 3 9 】

一方、本文書で説明される図面上の各構成は、互いに異なる特徴的な機能に関する説明の便宜のために独立的に図示されたものであって、各構成が互いに別個のハードウェアや別個のソフトウェアで具現されるということの意味するものではない。例えば、各構成のうち、二つ以上の構成が結合されて一つの構成をなすこともでき、一つの構成が複数の構成に分けられることもできる。各構成が統合及び/または分離された実施例も、本文書の本質から逸脱しない限り、本文書の権利範囲に含まれる。

【 0 0 4 0 】

以下、添付図面を参照して、本文書の好ましい実施例をより詳細に説明する。以下、図面上の同じ構成要素に対しては同じ参照符号を使用し、同じ構成要素に対して重複した説明は省略する。

【 0 0 4 1 】

この文書は、ビデオ/映像コーディングに関する。例えば、この文書に開示された方法/実施例は、VVC (Versatile Video Coding) 標準 (ITU-T Rec. H. 266)、VVC 以後の次世代ビデオ/イメージコーディング標準、またはその以外のビデオコーディング関連標準 (例えば、HEVC (High Efficiency Video Coding) 標準 (ITU-T Rec. H. 265)、EVC (essential video coding) 標準、AVS2 標準等) と関連付けられている。

【 0 0 4 2 】

この文書ではビデオ/映像コーディングに関する多様な実施例を提示し、他の言及がない限り、前記実施例は互いに組み合わせられて実行されることもできる。

【 0 0 4 3 】

この文書において、ビデオ (video) は、時間の流れによる一連の映像 (image) の集合を意味することができる。ピクチャ (picture) は、一般的に特定時間帯の一つの映像を示す単位を意味し、スライス (slice) / タイル (tile) は、コーディングにおいてピクチャの一部を構成する単位である。スライス/タイルは、一つ以上のCTU (coding tree unit) を含むことができる。一つのピクチャは、一つ以上のスライス/タイルで構成されることができる。一つのピクチャは、一つ以上のタイルグループで構成されることができる。一つのタイルグループは、一つ以上のタ

10

20

30

40

50

イルを含むことができる。

【0044】

ピクセル (pixel) またはペル (pel) は、一つのピクチャ (または、映像) を構成する最小の単位を意味することができる。また、ピクセルに対応する用語として「サンプル (sample)」が使われることができる。サンプルは、一般的にピクセルまたはピクセルの値を示すことができ、ルマ (luma) 成分のピクセル/ピクセル値のみを示すこともでき、クロマ (chroma) 成分のピクセル/ピクセル値のみを示すこともできる。または、サンプルは、空間ドメインでのピクセル値を意味することもでき、このようなピクセル値が周波数ドメインに変換されると、周波数ドメインでの変換係数を意味することもできる。

10

【0045】

ユニット (unit) は、映像処理の基本単位を示すことができる。ユニットは、ピクチャの特定領域及び該当領域に関連した情報のうち少なくとも一つを含むことができる。一つのユニットは、一つのルマブロック及び二つのクロマ (例えば、cb、cr) ブロックを含むことができる。ユニットは、場合によって、ブロック (block) または領域 (area) などの用語と混用して使われることができる。一般的な場合、M×Nブロックは、M個の列とN個の行からなるサンプル (または、サンプルアレイ) または変換係数 (transform coefficient) の集合 (または、アレイ) を含むことができる。

【0046】

この文書において、「/」と「、」は、「及び/または」と解釈される。例えば、「A/B」は、「A及び/またはB」と解釈され、「A、B」は、「A及び/またはB」と解釈される。追加的に、「A/B/C」は、「A、B及び/またはCのうち少なくとも一つ」を意味する。また、「A、B、C」も「A、B及び/またはCのうち少なくとも一つ」を意味する。(In this document, the term 「/」 and 「、」 should be interpreted to indicate 「and/or」. For instance, the expression 「A/B」 may mean 「A and/or B」. Further, 「A、B」 may mean 「A and/or B」. Further, 「A/B/C」 may mean 「at least one of A, B, and/or C」. Also, 「A/B/C」 may mean 「at least one of A, B, and/or C」.)

20

30

【0047】

追加的に、本文書において、「または」は、「及び/または」と解釈される。例えば、「AまたはB」は、1) 「A」のみを意味し、または2) 「B」のみを意味し、または3) 「A及びB」を意味することができる。他の表現としては、本文書の「または」は、「追加的にまたは代替的に (additionally or alternatively)」を意味することができる。(Further, in the document, the term 「or」 should be interpreted to indicate 「and/or」. For instance, the expression 「A or B」 may comprise 1) only A, 2) only B, and/or 3) both A and B. In other words, the term 「or」 in this document should be interpreted to indicate 「additionally or alternatively」.)

40

【0048】

本明細書において、「少なくとも一つのA及びB (at least one of A and B)」は、「ただA」、「ただB」または「AとBの両方とも」を意味することができる。また、本明細書において、「少なくとも一つのAまたはB (at least one of A or B)」や「少なくとも一つのA及び/またはB (at least one of A and/or B)」という表現は、「少なくとも一つのA及びB (a

50

t least one of A and B)」と同じく解釈されることができる。

【0049】

また、本明細書において、「少なくとも一つのA、B及びC (at least one of A, B and C)」は、「ただA」、「ただB」、「ただC」、または「A、B及びCの任意の全ての組み合わせ (any combination of A, B and C)」を意味することができる。また、「少なくとも一つのA、BまたはC (at least one of A, B or C)」や「少なくとも一つのA、B及び/またはC (at least one of A, B and/or C)」は、「少なくとも一つのA、B及びC (at least one of A, B and C)」を意味することができる。

10

【0050】

また、本明細書で使われる括弧は、「例えば (for example)」を意味することができる。具体的に、「予測 (イントラ予測)」で表示された場合、「予測」の一例として「イントラ予測」が提案されたものである。他の表現としては、本明細書の「予測」は、「イントラ予測」に制限 (limit) されるものではなく、「イントラ予測」が「予測」の一例として提案されたものである。また、「予測 (即ち、イントラ予測)」で表示された場合にも、「予測」の一例として「イントラ予測」が提案されたものである。

【0051】

本明細書において、一つの図面内で個別的に説明される技術的特徴は、個別的に具現されることもでき、同時に具現されることもできる。

20

【0052】

図1は、本文書が適用されることができるビデオ/映像コーディングシステムの例を概略的に示す。

【0053】

図1を参照すると、ビデオ/映像コーディングシステムは、ソースデバイス及び受信デバイスを含むことができる。ソースデバイスは、エンコーディングされたビデオ (video) / 映像 (image) 情報またはデータをファイルまたはストリーミング形態でデジタル格納媒体またはネットワークを介して受信デバイスに伝達できる。

【0054】

前記ソースデバイスは、ビデオソース、エンコーディング装置、送信部を含むことができる。前記受信デバイスは、受信部、デコーディング装置、及びレンダラを含むことができる。前記エンコーディング装置は、ビデオ/映像エンコーディング装置と呼ばれることができ、前記デコーディング装置は、ビデオ/映像デコーディング装置と呼ばれることができる。送信機は、エンコーディング装置に含まれることができる。受信機は、デコーディング装置に含まれることができる。レンダラは、ディスプレイ部を含むこともでき、ディスプレイ部は、別個のデバイスまたは外部コンポーネントで構成されることもできる。

30

【0055】

ビデオソースは、ビデオ/映像のキャプチャ、合成または生成過程などを介してビデオ/映像を取得することができる。ビデオソースは、ビデオ/映像キャプチャデバイス及び/またはビデオ/映像生成デバイスを含むことができる。ビデオ/映像キャプチャデバイスは、例えば、一つ以上のカメラ、以前にキャプチャされたビデオ/映像を含むビデオ/映像アーカイブなどを含むことができる。ビデオ/映像生成デバイスは、例えば、コンピュータ、タブレット、及びスマートフォンなどを含むことができ、(電子的に) ビデオ/映像を生成することができる。例えば、コンピュータなどを介して仮想のビデオ/映像が生成されることができ、この場合、関連データが生成される過程にビデオ/映像キャプチャ過程が代替されることができる。

40

【0056】

エンコーディング装置は、入力ビデオ/映像をエンコーディングすることができる。エンコーディング装置は、圧縮及びコーディング効率のために、予測、変換、量子化など、一連の手順を実行することができる。エンコーディングされたデータ (エンコーディング

50

されたビデオ／映像情報)は、ビットストリーム (b i t s t r e a m) 形態で出力されることができる。

【 0 0 5 7 】

送信部は、ビットストリーム形態で出力されたエンコーディングされたビデオ／映像情報またはデータをファイルまたはストリーミング形態でデジタル格納媒体またはネットワークを介して受信デバイスの受信部に伝達できる。デジタル格納媒体は、USB、SD、CD、DVD、ブルーレイ、HDD、SSDなど、多様な格納媒体を含むことができる。送信部は、あらかじめ決められたファイルフォーマットを介してメディアファイルを生成するためのエレメントを含むことができ、放送／通信ネットワークを介した送信のためのエレメントを含むことができる。受信部は、前記ビットストリームを受信／抽出してデコーディング装置に伝達できる。

10

【 0 0 5 8 】

デコーディング装置は、エンコーディング装置の動作に対応する逆量子化、逆変換、予測など、一連の手順を実行してビデオ／映像をデコーディングすることができる。

【 0 0 5 9 】

レンダラは、デコーディングされたビデオ／映像をレンダリングすることができる。レンダリングされたビデオ／映像は、ディスプレイ部を介してディスプレイされることができる。

【 0 0 6 0 】

図2は、本文書が適用されることができるビデオ／映像エンコーディング装置の構成を概略的に説明する図である。以下、ビデオエンコーディング装置とは、映像エンコーディング装置を含むことができる。

20

【 0 0 6 1 】

図2を参照すると、エンコーディング装置200は、映像分割部 (i m a g e p a r t i t i o n e r) 210、予測部 (p r e d i c t o r) 220、レジデュアル処理部 (r e s i d u a l p r o c e s s o r) 230、エントロピーエンコーディング部 (e n t r o p y e n c o d e r) 240、加算部 (a d d e r) 250、フィルタリング部 (f i l t e r) 260、及びメモリ (m e m o r y) 270を含んで構成されることができる。予測部220は、インター予測部221及びイントラ予測部222を含むことができる。レジデュアル処理部230は、変換部 (t r a n s f o r m e r) 232、量子化部 (q u a n t i z e r) 233、逆量子化部 (d e q u a n t i z e r) 234、逆変換部 (i n v e r s e t r a n s f o r m e r) 235を含むことができる。レジデュアル処理部230は、減算部 (s u b t r a c t o r) 231をさらに含むことができる。加算部250は、復元部 (r e c o n s t r u c t o r) または復元ブロック生成部 (r e c o n t r u c t e d b l o c k g e n e r a t o r) と呼ばれることができる。前述した映像分割部210、予測部220、レジデュアル処理部230、エントロピーエンコーディング部240、加算部250、及びフィルタリング部260は、実施例によって一つ以上のハードウェアコンポーネント (例えば、エンコーダチップセットまたはプロセッサ) により構成されることができる。また、メモリ270は、DPB (d e c o d e d p i c t u r e b u f f e r) を含むことができ、デジタル格納媒体により構成されることもできる。前記ハードウェアコンポーネントは、メモリ270を内／外部コンポーネントとしてさらに含むこともできる。

30

40

【 0 0 6 2 】

映像分割部210は、エンコーディング装置200に入力された入力映像 (または、ピクチャ、フレーム) を一つ以上の処理ユニット (p r o c e s s i n g u n i t) に分割できる。一例として、前記処理ユニットは、コーディングユニット (c o d i n g u n i t 、 C U) と呼ばれることができる。この場合、コーディングユニットは、コーディングツリーユニット (c o d i n g t r e e u n i t 、 C T U) または最大コーディングユニット (l a r g e s t c o d i n g u n i t 、 L C U) からQTBT (Q u a d - t r e e b i n a r y - t r e e t e r n a r y - t r e e) 構造によって再帰

50

的に (r e c u r s i v e l y) 分割されることができる。例えば、一つのコーディングユニットは、クアドツリー構造、バイナリツリー構造、及び/またはターナリ構造に基づいて下位 (d e e p e r) デプスの複数のコーディングユニットに分割されることができる。この場合、例えば、クアドツリー構造が先に適用され、バイナリツリー構造及び/またはターナリ構造がその後に適用されることができる。または、バイナリツリー構造が先に適用されることもできる。それ以上分割されない最終コーディングユニットに基づいて本文書によるコーディング手順が実行されることができる。この場合、映像特性によるコーディング効率などに基づいて、最大コーディングユニットが最終コーディングユニットとして使われることができ、または、必要によって、コーディングユニットは、再帰的に (r e c u r s i v e l y) もっと下位デプスのコーディングユニットに分割されて最適のサイズのコーディングユニットが最終コーディングユニットとして使われることができる。ここで、コーディング手順とは、後述する予測、変換、及び復元などの手順を含むことができる。他の例として、前記処理ユニットは、予測ユニット (P U : P r e d i c t i o n U n i t) または変換ユニット (T U : T r a n s f o r m U n i t) をさらに含むことができる。この場合、前記予測ユニット及び前記変換ユニットは、各々、前述した最終コーディングユニットから分割またはパーティショニングされることができる。前記予測ユニットは、サンプル予測の単位であり、前記変換ユニットは、変換係数を誘導する単位及び/または変換係数からレジデュアル信号 (r e s i d u a l s i g n a l) を誘導する単位である。

10

【 0 0 6 3 】

20

ユニットは、場合によって、ブロック (b l o c k) または領域 (a r e a) などの用語と混用して使われることができる。一般的な場合、 $M \times N$ ブロックは、 M 個の列と N 個の行からなるサンプルまたは変換係数 (t r a n s f o r m c o e f f i c i e n t) の集合を示すことができる。サンプルは、一般的にピクセルまたはピクセルの値を示すことができ、輝度 (l u m a) 成分のピクセル/ピクセル値のみを示すこともでき、彩度 (c h r o m a) 成分のピクセル/ピクセル値のみを示すこともできる。サンプルは、一つのピクチャ (または、映像) をピクセル (p i x e l) またはペル (p e l) に対応する用語として使われることができる。

【 0 0 6 4 】

減算部 2 3 1 は、入力映像信号 (原本ブロック、原本サンプルまたは原本サンプルアレイ) から、予測部 2 2 0 から出力された予測信号 (予測されたブロック、予測サンプルまたは予測サンプルアレイ) を減算してレジデュアル信号 (レジデュアルブロック、レジデュアルサンプルまたはレジデュアルサンプルアレイ) を生成することができ、生成されたレジデュアル信号は、変換部 2 3 2 に送信される。予測部 2 2 0 は、処理対象ブロック (以下、現在ブロックという) に対する予測を実行し、前記現在ブロックに対する予測サンプルを含む予測されたブロック (p r e d i c t e d b l o c k) を生成することができる。予測部 2 2 0 は、現在ブロックまたは $C U$ 単位でイントラ予測が適用されるか、または、インター予測が適用されるかを決定することができる。予測部は、各予測モードに対する説明で後述するように、予測モード情報など、予測に関する多様な情報を生成してエントロピーエンコーディング部 2 4 0 に伝達できる。予測に関する情報は、エントロピーエンコーディング部 2 4 0 でエンコーディングされてビットストリーム形態で出力されることができる。

30

40

【 0 0 6 5 】

イントラ予測部 2 2 2 は、現在ピクチャ内のサンプルを参照して現在ブロックを予測することができる。前記参照されるサンプルは、予測モードによって、前記現在ブロックの隣接 (n e i g h b o r) に位置することもでき、または離れて位置することもできる。イントラ予測で、予測モードは、複数の非方向性モードと複数の方向性モードを含むことができる。非方向性モードは、例えば、 $D C$ モード及びプラナーモード (P l a n a r モード) を含むことができる。方向性モードは、予測方向の細かい程度によって、例えば、3 3 個の方向性予測モードまたは 6 5 個の方向性予測モードを含むことができる。ただし

50

、これは例示に過ぎず、設定によってそれ以上またはその以下の個数の方向性予測モードが使われることができる。イントラ予測部 2 2 2 は、隣接ブロックに適用された予測モードを利用して、現在ブロックに適用される予測モードを決定することもできる。

【0066】

インター予測部 2 2 1 は、参照ピクチャ上で動きベクトルにより特定される参照ブロック（参照サンプルアレイ）に基づいて、現在ブロックに対する予測されたブロックを誘導することができる。このとき、インター予測モードで送信される動き情報の量を減らすために、隣接ブロックと現在ブロックとの間の動き情報の相関性に基づいて、動き情報をブロック、サブブロックまたはサンプル単位で予測できる。前記動き情報は、動きベクトル及び参照ピクチャインデックスを含むことができる。前記動き情報は、インター予測方向（L0 予測、L1 予測、Bi 予測等）情報をさらに含むことができる。インター予測の場合、隣接ブロックは、現在ピクチャ内に存在する空間的隣接ブロック（*spatial neighboring block*）と、参照ピクチャ内に存在する時間的隣接ブロック（*temporal neighboring block*）と、を含むことができる。前記参照ブロックを含む参照ピクチャと前記時間的隣接ブロックを含む参照ピクチャは、同じこともあり、異なることもある。前記時間的隣接ブロックは、同一位置参照ブロック（*collocated reference block*）、同一位置 CU（*colCU*）などの名称で呼ばれることができ、前記時間的隣接ブロックを含む参照ピクチャは、同一位置ピクチャ（*collocated picture*、*colPic*）と呼ばれることもできる。例えば、インター予測部 2 2 1 は、隣接ブロックに基づいて動き情報候補リストを構成し、前記現在ブロックの動きベクトル及び/または参照ピクチャインデックスを導出するために、どの候補が使われるかを指示する情報を生成することができる。多様な予測モードに基づいてインター予測が実行されることができ、例えば、スキップモードとマージモードの場合、インター予測部 2 2 1 は、隣接ブロックの動き情報を現在ブロックの動き情報として利用できる。スキップモードの場合、マージモードとは違って、レジデュアル信号が送信されない。動き情報予測（*motion vector prediction*、*MVP*）モードの場合、隣接ブロックの動きベクトルを動きベクトル予測子（*motion vector predictor*）として利用し、動きベクトル差分（*motion vector difference*）をシグナリングすることで、現在ブロックの動きベクトルを指示することができる。

【0067】

予測部 2 2 0 は、後述する多様な予測方法に基づいて予測信号を生成することができる。例えば、予測部は、一つのブロックに対する予測のために、イントラ予測またはインター予測を適用することができるだけでなく、イントラ予測とインター予測を同時に適用できる。これは *combined inter and intra prediction*（*CIIIP*）と呼ばれることができる。また、予測部は、ブロックに対する予測のために、イントラブロックコピー（*intra block copy*、*IBC*）を実行することもできる。前記イントラブロックコピーは、例えば、*SCC*（*screen content coding*）などのように、ゲームなどのコンテンツ映像/動映像コーディングのために使われることができる。*IBC* は、基本的に現在ピクチャ内で予測を実行するが、現在ピクチャ内で参照ブロックを導出する点でインター予測と類似するように実行されることができる。即ち、*IBC* は、本文書で説明されるインター予測技法のうち少なくとも一つを利用することができる。

【0068】

インター予測部 2 2 1 及び/またはイントラ予測部 2 2 2 を介して生成された予測信号は、復元信号を生成するために利用され、またはレジデュアル信号を生成するために利用されることができる。変換部 2 3 2 は、レジデュアル信号に変換技法を適用して変換係数（*transform coefficients*）を生成することができる。例えば、変換技法は、*DCT*（*Discrete Cosine Transform*）、*DST*（*Discrete Sine Transform*）、*GBT*（*Graph-Based*

10

20

30

40

50

Transform)、またはCNT(Conditionally Non-linear Transform)などを含むことができる。ここで、GBTは、ピクセル間の関係情報をグラフで表現するとする時、このグラフから得られた変換を意味する。CNTは、以前に復元された全てのピクセル(all previously reconstructed pixel)を利用して予測信号を生成し、それに基づいて取得される変換を意味する。また、変換過程は、正方形の同じ大きさを有するピクセルブロックに適用されることもでき、正方形でない可変大きさのブロックに適用されることもできる。

【0069】

量子化部233は、変換係数を量子化してエントロピーエンコーディング部240に送信し、エントロピーエンコーディング部240は、量子化された信号(量子化された変換係数に関する情報)をエンコーディングしてビットストリームで出力できる。前記量子化された変換係数に関する情報は、レジデュアル情報と呼ばれることができる。量子化部233は、係数スキャン順序(scan order)に基づいて、ブロック形態の量子化された変換係数を1次元ベクトル形態で再整列でき、前記1次元ベクトル形態の量子化された変換係数に基づいて、前記量子化された変換係数に関する情報を生成することもできる。エントロピーエンコーディング部240は、例えば、指数ゴロム(exponential Golomb)、CAVLC(context-adaptive variable length coding)、CABAC(context-adaptive binary arithmetic coding)などのような多様なエンコーディング方法を実行することができる。エントロピーエンコーディング部240は、量子化された変換係数外にビデオ/イメージ復元に必要な情報(例えば、シンタックス要素(syntax elements)の値等)を共にまたは別途にエンコーディングすることもできる。エンコーディングされた情報(例えば、エンコーディングされたビデオ/映像情報)は、ビットストリーム形態でNAL(network abstraction layer)ユニット単位で送信または格納されることができる。前記ビデオ/映像情報は、アダプテーションパラメータセット(APS)、ピクチャパラメータセット(PPS)、シーケンスパラメータセット(SPS)またはビデオパラメータセット(VPS)等、多様なパラメータセットに関する情報をさらに含むことができる。また、前記ビデオ/映像情報は、一般制限情報(general constraint information)をさらに含むことができる。本文書で後述されるシグナリング/送信される情報及び/またはシンタックス要素は、前述したエンコーディング手順を介してエンコーディングされて前記ビットストリームに含まれることができる。前記ビットストリームは、ネットワークを介して送信されることができ、またはデジタル格納媒体に格納されることができる。ここで、ネットワークは、放送網及び/または通信網などを含むことができ、デジタル格納媒体は、USB、SD、CD、DVD、ブルーレイ、HDD、SSDなど、多様な格納媒体を含むことができる。エントロピーエンコーディング部240から出力された信号は、送信する送信部(図示せず)及び/または格納する格納部(図示せず)がエンコーディング装置200の内/外部エレメントとして構成されることができ、または、送信部は、エントロピーエンコーディング部240に含まれることもできる。

【0070】

量子化部233から出力された量子化された変換係数は、予測信号を生成するために利用されることができる。例えば、量子化された変換係数に逆量子化部234及び逆変換部235を介して逆量子化及び逆変換を適用することによって、レジデュアル信号(レジデュアルブロックまたはレジデュアルサンプル)を復元することができる。加算部250は、復元されたレジデュアル信号を予測部220から出力された予測信号に加えることによって復元(reconstructed)信号(復元ピクチャ、復元ブロック、復元サンプルまたは復元サンプルアレイ)が生成されることができる。スキップモードが適用された場合のように処理対象ブロックに対するレジデュアルがない場合、予測されたブロックが復元ブロックとして使われることができる。生成された復元信号は、現在ピクチャ内の次の処理対象ブロックのイントラ予測のために使われることができ、後述するようにフィ

10

20

30

40

50

ルタリングを経て次のピクチャのインター予測のために使われることもできる。

【0071】

一方、ピクチャエンコーディング及び/または復元過程でLMCS (luma mapping with chroma scaling) が適用されることもできる。

【0072】

フィルタリング部260は、復元信号にフィルタリングを適用して主観的/客観的画質を向上させることができる。例えば、フィルタリング部260は、復元ピクチャに多様なフィルタリング方法を適用して修正された(modified)復元ピクチャを生成することができ、前記修正された復元ピクチャをメモリ270、具体的に、メモリ270のDPBに格納することができる。前記多様なフィルタリング方法は、例えば、デブロッキング
10
フィルタリング、サンプル適応的オフセット(sample adaptive offset、SAO)、適応的ループフィルタ(adaptive loop filter)、両方向フィルタ(bilateral filter)などを含むことができる。フィルタリング部260は、各フィルタリング方法に対する説明で後述するように、フィルタリングに関する多様な情報を生成してエントロピーエンコーディング部240に伝達できる。フィルタリングに関する情報は、エントロピーエンコーディング部240でエンコーディングされてビットストリーム形態で出力されることができる。

【0073】

メモリ270に送信された修正された復元ピクチャは、インター予測部221で参照ピクチャとして使われることができる。エンコーディング装置は、これを介してインター予測
20
が適用される場合、エンコーディング装置200とデコーディング装置での予測ミスマッチを避けることができ、符号化効率も向上させることができる。

【0074】

メモリ270のDPBは、修正された復元ピクチャをインター予測部221での参照ピクチャとして使用するために格納することができる。メモリ270は、現在ピクチャ内の動き情報が導出された(または、エンコーディングされた)ブロックの動き情報及び/または既に復元されたピクチャ内のブロックの動き情報を格納することができる。前記格納された動き情報は、空間的隣接ブロックの動き情報または時間的隣接ブロックの動き情報として活用するためにインター予測部221に伝達できる。メモリ270は、現在ピクチャ内の復元されたブロックの復元サンプルを格納することができ、イントラ予測部222
30
に伝達できる。

【0075】

図3は、本文書が適用されることができるビデオ/映像デコーディング装置の構成を概略的に説明する図である。

【0076】

図3を参照すると、デコーディング装置300は、エントロピーデコーディング部(entropy decoder)310、レジデュアル処理部(residual processor)320、予測部(predictor)330、加算部(adder)340、フィルタリング部(filter)350、及びメモリ(memoery)360を含んで構成されることができる。予測部330は、インター予測部332及びイントラ
40
予測部331を含むことができる。レジデュアル処理部320は、逆量子化部(dequantizer)321及び逆変換部(inverse transformer)322を含むことができる。前述したエントロピーデコーディング部310、レジデュアル処理部320、予測部330、加算部340、及びフィルタリング部350は、実施例によって一つのハードウェアコンポーネント(例えば、デコーダチップセットまたはプロセッサ)により構成されることができる。また、メモリ360は、DPB(decoded picture buffer)を含むことができ、デジタル格納媒体により構成されることができる。前記ハードウェアコンポーネントは、メモリ360を内/外部コンポーネントとしてさらに含むこともできる。

【0077】

10

20

30

40

50

ビデオ/映像情報を含むビットストリームが入力される場合、デコーディング装置300は、図2のエンコーディング装置でビデオ/映像情報が処理されたプロセスに対応して映像を復元することができる。例えば、デコーディング装置300は、前記ビットストリームから取得したブロック分割関連情報に基づいてユニット/ブロックを導出することができる。デコーディング装置300は、エンコーディング装置で適用された処理ユニットを利用してデコーディングを実行することができる。したがって、デコーディングの処理ユニットは、例えば、コーディングユニットであり、コーディングユニットは、コーディングツリーユニットまたは最大コーディングユニットから、クアドツリー構造、バイナリツリー構造及び/またはターナリツリー構造によって分割されることができる。コーディングユニットから一つ以上の変換ユニットが導出されることができる。そして、デコーディング装置300を介してデコーディング及び出力された復元映像信号は、再生装置を介して再生されることができる。

【0078】

デコーディング装置300は、図2のエンコーディング装置から出力された信号をビットストリーム形態で受信することができ、受信された信号は、エントロピーデコーディング部310を介してデコーディングされることができる。例えば、エントロピーデコーディング部310は、前記ビットストリームをパーシングして映像復元(または、ピクチャ復元)に必要な情報(例えば、ビデオ/映像情報)を導出することができる。前記ビデオ/映像情報は、アダプテーションパラメータセット(APS)、ピクチャパラメータセット(PPS)、シーケンスパラメータセット(PPS)、またはビデオパラメータセット(VPS)等、多様なパラメータセットに関する情報をさらに含むことができる。また、前記ビデオ/映像情報は、一般制限情報(general constraint information)をさらに含むことができる。デコーディング装置は、前記パラメータセットに関する情報及び/または前記一般制限情報にさらに基づいてピクチャをデコーディングすることができる。本文書で後述されるシグナリング/受信される情報及び/またはシンタックス要素は、前記デコーディング手順を介してデコーディングされて前記ビットストリームから取得されることができる。例えば、エントロピーデコーディング部310は、指数ゴロム符号化、CAVLCまたはCABACなどのコーディング方法に基づいてビットストリーム内の情報をデコーディングし、映像復元に必要なシンタックスエレメントの値、レジデュアルに関する変換係数の量子化された値を出力することができる。より詳しくは、CABACエントロピーデコーディング方法は、ビットストリームで各構文要素に該当するbinを受信し、デコーディング対象構文要素情報と隣接及びデコーディング対象ブロックのデコーディング情報または以前ステップでデコーディングされたシンボル/binの情報を利用してコンテキスト(context)モデルを決定し、決定されたコンテキストモデルによってbinの発生確率を予測してbinの算術デコーディング(arithmetic decoding)を実行することで各構文要素の値に該当するシンボルを生成することができる。このとき、CABACエントロピーデコーディング方法は、コンテキストモデル決定後、次のシンボル/binのコンテキストモデルのためにデコーディングされたシンボル/binの情報を利用してコンテキストモデルをアップデートすることができる。エントロピーデコーディング部310でデコーディングされた情報のうち、予測に関する情報は、予測部330に提供され、エントロピーデコーディング部310でエントロピーデコーディングが実行されたレジデュアルに対する情報、即ち、量子化された変換係数及び関連パラメータ情報は、逆量子化部321に入力されることができる。また、エントロピーデコーディング部310でデコーディングされた情報のうち、フィルタリングに関する情報は、フィルタリング部350に提供されることができる。一方、エンコーディング装置から出力された信号を受信する受信部(図示せず)がデコーディング装置300の内/外部エレメントとしてさらに構成されてもよく、または、受信部は、エントロピーデコーディング部310の構成要素であってもよい。一方、本文書によるデコーディング装置は、ビデオ/映像/ピクチャデコーディング装置と呼ばれることができ、前記デコーディング装置は、情報デコーダ(ビデオ/映像/ピクチャ情報デ

10

20

30

40

50

コーダ)及びサンプルデコーダ(ビデオ/映像/ピクチャサンプルデコーダ)に区分することもできる。前記情報デコーダは、前記エンロピーデコーディング部310を含むことができ、前記サンプルデコーダは、前記逆量子化部321、逆変換部322、予測部330、加算部340、フィルタリング部350、及びメモリ360のうち少なくとも一つを含むことができる。

【0079】

逆量子化部321では量子化された変換係数を逆量子化して変換係数を出力することができる。逆量子化部321は、量子化された変換係数を2次元のブロック形態で再整列できる。この場合、前記再整列は、エンコーディング装置で実行された係数スキャン順序に基づいて再整列を実行することができる。逆量子化部321は、量子化パラメータ(例えば、量子化ステップサイズ情報)を利用して量子化された変換係数に対する逆量子化を実行し、変換係数(transform coefficient)を取得することができる。

10

【0080】

逆変換部322では変換係数を逆変換してレジデュアル信号(レジデュアルブロック、レジデュアルサンプルアレイ)を取得するようになる。

【0081】

予測部は、現在ブロックに対する予測を実行し、前記現在ブロックに対する予測サンプルを含む予測されたブロック(predicted block)を生成することができる。予測部は、エンロピーデコーディング部310から出力された前記予測に関する情報に基づいて、前記現在ブロックにイントラ予測が適用されるか、または、インター予測が適用されるかを決定することができる、具体的なイントラ/インター予測モードを決定することができる。

20

【0082】

予測部は、後述する多様な予測方法に基づいて予測信号を生成することができる。例えば、予測部は、一つのブロックに対する予測のためにイントラ予測またはインター予測を適用することができるだけでなく、イントラ予測とインター予測を同時に適用することもできる。これはcombined inter and intra prediction(CIIP)と呼ばれることができる。また、予測部は、ブロックに対する予測のためにイントラブロックコピー(intra block copy、IBC)を実行することもできる。前記イントラブロックコピーは、例えば、SCC(screen content coding)などのように、ゲームなどのコンテンツ映像/動映像コーディングのために使われることができる。IBCは、基本的に現在ピクチャ内で予測を実行するが、現在ピクチャ内で参照ブロックを導出する点でインター予測と類似するように実行される。即ち、IBCは、本文書で説明されるインター予測技法のうち少なくとも一つを利用することができる。

30

【0083】

イントラ予測部331は、現在ピクチャ内のサンプルを参照して現在ブロックを予測することができる。前記参照されるサンプルは、予測モードによって、前記現在ブロックの隣接(neighbor)に位置することもでき、または離れて位置することもできる。イントラ予測で、予測モードは、複数の非方向性モードと複数の方向性モードを含むことができる。イントラ予測部331は、隣接ブロックに適用された予測モードを利用し、現在ブロックに適用される予測モードを決定することもできる。

40

【0084】

インター予測部332は、参照ピクチャ上で動きベクトルにより特定される参照ブロック(参照サンプルアレイ)に基づいて、現在ブロックに対する予測されたブロックを誘導することができる。このとき、インター予測モードで送信される動き情報の量を減らすために、隣接ブロックと現在ブロックとの間の動き情報の相関性に基づいて動き情報をブロック、サブブロック、またはサンプル単位で予測できる。前記動き情報は、動きベクトル及び参照ピクチャインデックスを含むことができる。前記動き情報は、インター予測方向

50

(L0予測、L1予測、Bi予測等)情報をさらに含むことができる。インター予測の場合、隣接ブロックは、現在ピクチャ内に存在する空間的隣接ブロック(spatial neighboring block)と、参照ピクチャ内に存在する時間的隣接ブロック(temporal neighboring block)と、を含むことができる。例えば、インター予測部332は、隣接ブロックに基づいて動き情報候補リストを構成し、受信した候補選択情報に基づいて、前記現在ブロックの動きベクトル及び/または参照ピクチャインデックスを導出することができる。多様な予測モードに基づいてインター予測が実行されることができ、前記予測に関する情報は、前記現在ブロックに対するインター予測のモードを指示する情報を含むことができる。

【0085】

加算部340は、取得されたレジデュアル信号を、予測部330から出力された予測信号(予測されたブロック、予測サンプルアレイ)に加えることによって復元信号(復元ピクチャ、復元ブロック、復元サンプルアレイ)を生成することができる。スキップモードが適用された場合のように処理対象ブロックに対するレジデュアルがない場合、予測されたブロックが復元ブロックとして使われることができる。

【0086】

加算部340は、復元部または復元ブロック生成部と呼ばれることができる。生成された復元信号は、現在ピクチャ内の次の処理対象ブロックのイントラ予測のために使われることができ、後述するように、フィルタリングを経て出力されることもでき、または、次のピクチャのインター予測のために使われることもできる。

【0087】

一方、ピクチャデコーディング過程でLMCS(luma mapping with chroma scaling)が適用されることもできる。

【0088】

フィルタリング部350は、復元信号にフィルタリングを適用して主観的/客観的画質を向上させることができる。例えば、フィルタリング部350は、復元ピクチャに多様なフィルタリング方法を適用することで、修正された(modified)復元ピクチャを生成することができ、前記修正された復元ピクチャをメモリ360、具体的に、メモリ360のDPBに送信できる。前記多様なフィルタリング方法は、例えば、デブロッキングフィルタリング、サンプル適応的オフセット(sample adaptive offset)、適応的ループフィルタ(adaptive loop filter)、両方向フィルタ(bilateral filter)などを含むことができる。

【0089】

メモリ360のDPBに格納された(修正された)復元ピクチャは、インター予測部332で参照ピクチャとして使われることができる。メモリ360は、現在ピクチャ内の動き情報が導出された(または、デコーディングされた)ブロックの動き情報及び/または既に復元されたピクチャ内のブロックの動き情報を格納することができる。前記格納された動き情報は、空間的隣接ブロックの動き情報または時間的隣接ブロックの動き情報として活用するために、インター予測部332に伝達できる。メモリ360は、現在ピクチャ内の復元されたブロックの復元サンプルを格納することができ、イントラ予測部331に伝達できる。

【0090】

本明細書において、デコーディング装置300の予測部330、逆量子化部321、逆変換部322、及びフィルタリング部350などで説明された実施例は、各々、エンコーディング装置200の予測部220、逆量子化部234、逆変換部235、及びフィルタリング部260などにも同一または対応されるように適用されることができる。

【0091】

前述したように、ビデオコーディングを実行するにあたって、圧縮効率を上げるために予測を実行する。これを介してコーディング対象ブロックである現在ブロックに対する予測サンプルを含む予測されたブロックを生成することができる。ここで、前記予測された

10

20

30

40

50

ブロックは、空間ドメイン（または、ピクセルドメイン）での予測サンプルを含む。前記予測されたブロックは、エンコーディング装置及びデコーディング装置で同じく導出され、前記エンコーディング装置は、原本ブロックの原本サンプル値自体でない前記原本ブロックと前記予測されたブロックとの間のレジデュアルに対する情報（レジデュアル情報）をデコーディング装置にシグナリングすることで映像コーディング効率を上げることができる。デコーディング装置は、前記レジデュアル情報に基づいてレジデュアルサンプルを含むレジデュアルブロックを導出し、前記レジデュアルブロックと前記予測されたブロックとを合わせて復元サンプルを含む復元ブロックを生成することができ、復元ブロックを含む復元ピクチャを生成することができる。

【 0 0 9 2 】

前記レジデュアル情報は、変換及び量子化手順を介して生成されることができる。例えば、エンコーディング装置は、前記原本ブロックと前記予測されたブロックとの間のレジデュアルブロックを導出し、前記レジデュアルブロックに含まれているレジデュアルサンプル（レジデュアルサンプルアレイ）に変換手順を実行して変換係数を導出し、前記変換係数に量子化手順を実行して量子化された変換係数を導出して関連したレジデュアル情報を（ビットストリームを介して）デコーディング装置にシグナリングできる。ここで、前記レジデュアル情報は、前記量子化された変換係数の値情報、位置情報、変換技法、変換カーネル、量子化パラメータなどの情報を含むことができる。デコーディング装置は、前記レジデュアル情報に基づいて逆量子化／逆変換手順を実行してレジデュアルサンプル（または、レジデュアルブロック）を導出することができる。デコーディング装置は、予測されたブロックと前記レジデュアルブロックとに基づいて復元ピクチャを生成することができる。また、エンコーディング装置は、以後ピクチャのインター予測のための参照のために量子化された変換係数を逆量子化／逆変換してレジデュアルブロックを導出し、これに基づいて復元ピクチャを生成することができる。

【 0 0 9 3 】

図 4 は、本文書が適用されるコンテンツストリーミングシステム構造図を例示的に示す。

【 0 0 9 4 】

また、本文書が適用されるコンテンツストリーミングシステムは、大いに、エンコーディングサーバ、ストリーミングサーバ、ウェブサーバ、メディア格納所、ユーザ装置、及びマルチメディア入力装置を含むことができる。

【 0 0 9 5 】

前記エンコーディングサーバは、スマートフォン、カメラ、カムコーダなどのようなマルチメディア入力装置から入力されたコンテンツをデジタルデータで圧縮してビットストリームを生成し、これを前記ストリーミングサーバに送信する役割をする。他の例として、スマートフォン、カメラ、カムコーダなどのようなマルチメディア入力装置がビットストリームを直接生成する場合、前記エンコーディングサーバは省略されることができる。前記ビットストリームは、本文書が適用されるエンコーディング方法またはビットストリーム生成方法により生成されることができ、前記ストリーミングサーバは、前記ビットストリームを送信または受信する過程で一時的に前記ビットストリームを格納することができる。

【 0 0 9 6 】

前記ストリーミングサーバは、ウェブサーバを介したユーザ要請に基づいてマルチメディアデータをユーザ装置に送信し、前記ウェブサーバは、どのようなサービスがあるかをユーザに知らせる媒体役割をする。ユーザが前記ウェブサーバに所望のサービスを要請すると、前記ウェブサーバは、これをストリーミングサーバに伝達し、前記ストリーミングサーバは、ユーザにマルチメディアデータを送信する。このとき、前記コンテンツストリーミングシステムは、別途の制御サーバを含むことができ、この場合、前記制御サーバは、前記コンテンツストリーミングシステム内の各装置間の命令／応答を制御する役割をする。

【 0 0 9 7 】

10

20

30

40

50

前記ストリーミングサーバは、メディア格納所及び/またはエンコーディングサーバからコンテンツを受信することができる。例えば、前記エンコーディングサーバからコンテンツを受信するようになる場合、前記コンテンツをリアルタイムで受信することができる。この場合、円滑なストリーミングサービスを提供するために、前記ストリーミングサーバは、前記ビットストリームを一定時間の間に格納することができる。

【0098】

前記ユーザ装置の例として、携帯電話、スマートフォン (smart phone)、ノートパソコン (laptop computer)、デジタル放送用端末機、PDA (personal digital assistants)、PMP (portable multimedia player)、ナビゲーション、スレートPC (slate PC)、タブレットPC (tablet PC)、ウルトラブック (ultrabook)、ウェアラブルデバイス (wearable device、例えば、ウォッチ型端末機 (smart watch)、グラス型端末機 (smart glass)、HMD (head mounted display)、デジタルTV、デスクトップコンピュータ、デジタルサイニジなどがある。前記コンテンツストリーミングシステム内の各サーバは、分散サーバで運営されることができ、この場合、各サーバで受信するデータは、分散処理されることができる。

【0099】

図5は、本文書による多重変換技法を概略的に示す。

【0100】

図5を参照すると、変換部は、前述した図2のエンコーディング装置内の変換部に対応されることができ、逆変換部は、前述した図2のエンコーディング装置内の逆変換部または図3のデコーディング装置内の逆変換部に対応されることができる。

【0101】

変換部は、レジデュアルブロック内のレジデュアルサンプル (レジデュアルサンプルアレイ) に基づいて1次変換を実行して (1次) 変換係数を導出することができる (S510)。このような1次変換 (primary transform) は、核心変換 (core transform) と呼ばれることができる。ここで、前記1次変換は、多重変換選択 (Multiple Transform Selection、MTS) に基づくことができ、1次変換として多重変換が適用される場合、多重核心変換と呼ばれることができる。

【0102】

多重核心変換は、DCT (Discrete Cosine Transform) タイプ2とDST (Discrete Sine Transform) タイプ7、DCTタイプ8、及び/またはDSTタイプ1を追加的に使用して変換する方式を示すことができる。即ち、前記多重核心変換は、前記DCTタイプ2、前記DSTタイプ7、前記DCTタイプ8、及び前記DSTタイプ1の中から選択された複数の変換カーネルに基づいて、空間ドメインのレジデュアル信号 (または、レジデュアルブロック) を周波数ドメインの変換係数 (または、1次変換係数) に変換する変換方法を示すことができる。ここで、前記1次変換係数は、変換部立場で臨時変換係数と呼ばれることができる。

【0103】

即ち、既存の変換方法が適用される場合、DCTタイプ2に基づいて、レジデュアル信号 (または、レジデュアルブロック) に対する空間ドメインから周波数ドメインへの変換が適用されて変換係数が生成されることができた。これと違って、前記多重核心変換が適用される場合、DCTタイプ2、DSTタイプ7、DCTタイプ8、及び/またはDSTタイプ1などに基づいて、レジデュアル信号 (または、レジデュアルブロック) に対する空間ドメインから周波数ドメインへの変換が適用されて変換係数 (または、1次変換係数) が生成されることができる。ここで、DCTタイプ2、DSTタイプ7、DCTタイプ8、及びDSTタイプ1等は、変換タイプ、変換カーネル (kernel) または変換コア (core) と呼ばれることができる。このようなDCT/DST変換タイプは、基底

10

20

30

40

50

関数に基づいて定義されることができる。

【0104】

前記多重核心変換が実行される場合、前記変換カーネルの中から対象ブロックに対する垂直変換カーネル及び水平変換カーネルが選択されることができ、前記垂直変換カーネルに基づいて前記対象ブロックに対する垂直変換が実行され、前記水平変換カーネルに基づいて前記対象ブロックに対する水平変換が実行されることができ、ここで、前記水平変換は、前記対象ブロックの水平成分に対する変換を示すことができ、前記垂直変換は、前記対象ブロックの垂直成分に対する変換を示すことができる。前記垂直変換カーネル/水平変換カーネルは、レジデュアルブロックを含む対象ブロック(CUまたはサブブロック)の予測モード及び/または変換インデックスに基づいて適応的に決定されることができる。

10

【0105】

また、一例によると、MTSを適用して1次変換を実行する場合、特定基底関数を所定値に設定し、垂直変換または水平変換である時、どのような基底関数が適用されるかを組み合わせ変換カーネルに対するマッピング関係を設定することができる。例えば、水平方向変換カーネルをtrTypeHorで表し、垂直方向変換カーネルをtrTypeVerで表す場合、trTypeHorまたはtrTypeVer値0はDCT2に設定され、trTypeHorまたはtrTypeVer値1はDST7に設定され、trTypeHorまたはtrTypeVer値2はDCT8に設定されることができる。

20

【0106】

この場合、多数の変換カーネルセットのうちいずれか一つを指示するために、MTSインデックス情報がエンコーディングされてデコーディング装置にシグナリングされることができる。例えば、MTSインデックスが0である場合、trTypeHor及びtrTypeVer値が両方とも0であることを指示し、MTSインデックスが1である場合、trTypeHor及びtrTypeVer値が両方とも1であることを指示し、MTSインデックスが2である場合、trTypeHor値は2であり、trTypeVer値は1であることを指示し、MTSインデックスが3である場合、trTypeHor値は1であり、trTypeVer値は2であることを指示し、MTSインデックスが4である場合、trTypeHor及びtrTypeVer値が両方とも2であることを指示することができる。

30

【0107】

一例によって、MTSインデックス情報による変換カーネルセットを表で示すと、以下の通りである。

【0108】

【表1】

tu_mts_idx[x0][y0]	0	1	2	3	4
trTypeHor	0	1	2	1	2
trTypeVer	0	1	1	2	2

40

【0109】

変換部は、前記(1次)変換係数に基づいて2次変換を実行して修正された(2次)変換係数を導出することができる(S520)。前記1次変換は、空間ドメインから周波数ドメインへの変換であり、前記2次変換は、(1次)変換係数間に存在する相関関係(correlation)を利用してもっと圧縮的な表現で変換することを意味する。前記2次変換は、非分離変換(non-separable transform)を含むことができる。この場合、前記2次変換は、非分離2次変換(non-separable

50

secondary transform、NSST)またはMDNSST(mode-dependent non-separable secondary transform)と呼ばれることができる。前記非分離2次変換は、前記1次変換を介して導出された(1次)変換係数を非分離変換マトリクス(non-separable transform matrix)に基づいて2次変換してレジデュアル信号に対する修正された変換係数(または、2次変換係数)を生成する変換を示すことができる。ここで、前記非分離変換マトリクスに基づいて、前記(1次)変換係数に対して垂直変換及び水平変換を分離して(または、水平垂直変換を独立的に)適用せずに、一度に変換を適用することができる。即ち、前記非分離2次変換は、前記(1次)変換係数に対して垂直方向と水平方向に別に適用されずに、例えば、2次元信号(変換係数)を特定決められた方向(例えば、行優先(row-first)方向または列優先(column-first)方向)を介して1次元信号に再整理した後、前記非分離変換マトリクスに基づいて修正された変換係数(または、2次変換係数)を生成する変換方法を示すことができる。例えば、行優先順序は、M×Nブロックに対して1番目の行、2番目の行、...、N番目の行の順序に一行に配置することであり、列優先順序は、M×Nブロックに対して1番目の列、2番目の列、...、M番目の列の順序に一行に配置することである。前記非分離2次変換は、(1次)変換係数で構成されたブロック(以下、変換係数ブロックと呼ばれることができる)の左上端(top-left)領域に対して適用されることができる。例えば、前記変換係数ブロックの幅(W)及び高さ(H)が両方とも8以上であると、8×8非分離2次変換が前記変換係数ブロックの左上端8×8領域に対して適用されることができる。また、前記変換係数ブロックの幅(W)及び高さ(H)が両方とも4以上であり、前記変換係数ブロックの幅(W)または高さ(H)が8より小さいと、4×4非分離2次変換が前記変換係数ブロックの左上端min(8、W)×min(8、H)領域に対して適用されることができる。ただし、実施例は、これに限定されるものではなく、例えば、前記変換係数ブロックの幅(W)または高さ(H)が両方とも4以上である条件のみを満たしても、4×4非分離2次変換が前記変換係数ブロックの左上端min(8、W)×min(8、H)領域に対して適用されることもできる。

10

20

【0110】

具体的に、例えば、4×4入力ブロックが使われる場合、非分離2次変換は、下記のように実行されることができる。

30

【0111】

前記4×4入力ブロックXは、以下のように示される。

【0112】

【数1】

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

40

【0113】

前記Xをベクトル形態で表す場合、ベクトル

\vec{X}

は、以下のように示される。

50

【 0 1 1 4 】

【数 2】

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

【 0 1 1 5 】

数式 2 のように、ベクトル

$$\vec{X}$$

10

は、行優先 (row - first) 順序によって、数式 1 の X の 2 次元ブロックを 1 次元ベクトルで再配列する。

【 0 1 1 6 】

この場合、前記 2 次非分離変換は、下記のように計算されることができる。

【 0 1 1 7 】

【数 3】

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

20

【 0 1 1 8 】

ここで、

$$\vec{F}$$

は、変換係数ベクトルを示し、T は、16 × 16 (非分離) 変換マトリクスを示す。

【 0 1 1 9 】

前記数式 3 を介して 16 × 1 変換係数ベクトル

30

$$\vec{F}$$

が導出されることができ、前記

$$\vec{F}$$

は、スキャン順序 (水平、垂直、対角 (diagonal) 等) を介して 4 × 4 ブロックで再構成 (re - organized) されることができる。ただし、前述した計算は、例示に過ぎず、非分離 2 次変換の計算複雑度を減らすために、HyGT (Hypercube - Givens Transform) などが非分離 2 次変換の計算のために使われることもできる。

40

【 0 1 2 0 】

一方、前記非分離 2 次変換は、モードベースの (mode dependent) 変換カーネル (または、変換コア、変換タイプ) が選択されることができる。ここで、モードは、イントラ予測モード及び/またはインター予測モードを含むことができる。

【 0 1 2 1 】

前述したように、前記非分離 2 次変換は、前記変換係数ブロックの幅 (W) 及び高さ (H) に基づいて決定された 8 × 8 変換または 4 × 4 変換に基づいて実行されることができ

50

る。8 × 8 変換は、W と H が両方とも 8 より大きいまたは同じ時、該当変換係数ブロック内部に含まれている 8 × 8 領域に適用されることができる変換を指し、該当 8 × 8 領域は、該当変換係数ブロック内部の左上端 8 × 8 領域である。同様に、4 × 4 変換は、W と H が両方とも 4 より大きいまたは同じ時、該当変換係数ブロック内部に含まれている 4 × 4 領域に適用されることができる変換を指し、該当 4 × 4 領域は、該当変換係数ブロック内部の左上端 4 × 4 領域である。例えば、8 × 8 変換カーネルマトリクスは、64 × 64 / 16 × 64 行列、4 × 4 変換カーネルマトリクスは、16 × 16 / 8 × 16 行列になることができる。

【0122】

このとき、モードベースの変換カーネル選択のために、8 × 8 変換及び 4 × 4 変換の両方ともに対して非分離 2 次変換のための変換セット当たり 2 個ずつの非分離 2 次変換カーネルが構成されることができ、変換セットは、4 個である。即ち、8 × 8 変換に対して 4 個の変換セットが構成され、4 × 4 変換に対して 4 個の変換セットが構成されることができる。この場合、8 × 8 変換に対する 4 個の変換セットには、各々、2 個ずつの 8 × 8 変換カーネルが含まれることができ、この場合、4 × 4 変換に対する 4 個の変換セットには、各々、2 個ずつの 4 × 4 変換カーネルが含まれることができる。

【0123】

ただし、前記変換のサイズ、即ち、変換が適用される領域のサイズは、例示に過ぎず、8 × 8 または 4 × 4 以外のサイズが使われることができ、前記セットの数は n 個、各セット内の変換カーネルの数は k 個である。

【0124】

前記変換セットは、NSST セットまたは LFNST セットと呼ばれることができる。前記変換セットの中からの特定セットの選択は、例えば、現在ブロック (CU またはサブブロック) のイントラ予測モードに基づいて実行されることができる。LFNST (Low-Frequency Non-Separable Transform) は、後述される減少された非分離変換の一例であり、低周波成分に対する非分離変換を示す。

【0125】

参考までに、例えば、イントラ予測モードは、2 個の非方向性 (non-directional、または非角度性 (non-angular)) イントラ予測モードと、65 個の方向性 (directional、または角度性 (angular)) イントラ予測モードと、を含むことができる。前記非方向性イントラ予測モードは、0 番であるプランナー (planar) イントラ予測モード及び 1 番である DC イントラ予測モードを含むことができ、前記方向性イントラ予測モードは、2 番乃至 66 番の 65 個のイントラ予測モードを含むことができる。ただし、これは例示に過ぎず、本文書は、イントラ予測モードの数が異なる場合にも適用されることができる。一方、場合によって、67 番イントラ予測モードがさらに使用されることができ、前記 67 番イントラ予測モードは、LM (linear model) モードを示すことができる。

【0126】

図 6 は、65 個の予測方向のイントラ方向性モードを例示的に示す。

【0127】

図 6 を参照すると、右下向対角予測方向を有する 34 番イントラ予測モードを中心にして、水平方向性 (horizontal directionality) を有するイントラ予測モードと、垂直方向性 (vertical directionality) を有するイントラ予測モードと、を区分することができる。図 6 の H と V は、各々、水平方向性と垂直方向性を意味し、-32 ~ 32 の数字は、サンプルグリッドポジション (sample grid position) 上で 1 / 32 単位の変位を示す。これはモードインデックス値に対するオフセットを示すことができる。2 番乃至 33 番イントラ予測モードは水平方向性を有し、34 番乃至 66 番イントラ予測モードは垂直方向性を有する。一方、34 番イントラ予測モードは、厳密には、水平方向性でも垂直方向性でもないとみることができるが、2 次変換の変換セットを決定する観点で水平方向性に属すると分類され

10

20

30

40

50

ることができる。これは、34番イントラ予測モードを中心にして対称である垂直方向モードに対しては入力データをトランスポーズ (t r a n s p o s e) して使用し、34番イントラ予測モードに対しては水平方向モードに対する入力データ整列方式を使用するためである。入力データをトランスポーズすることは、2次元ブロックデータM×Nに対して行が列になり、列が行になってN×Mデータを構成することを意味する。18番イントラ予測モードと50番イントラ予測モードは、各々、水平イントラ予測モード (h o r i z o n t a l i n t r a p r e d i c t i o n m o d e) 、垂直イントラ予測モード (v e r t i c a l i n t r a p r e d i c t i o n m o d e) を示し、2番イントラ予測モードは、左側参照ピクセルを有して右上向方向に予測するため、右上向対角イントラ予測モードと呼ばれることができ、同じ脈絡で、34番イントラ予測モードは右下向対角イントラ予測モードと呼ばれ、66番イントラ予測モードは左下向対角イントラ予測モードと呼ばれることができる。

10

【0128】

一例によって、イントラ予測モードによって4個の変換セットのマッピング (m a p p i n g) は、例えば、以下の表のように示される。

【0129】

【表2】

lfnstPredModeIntra	lfnstTrSetIdx
lfnstPredModeIntra < 0	1
0 <= lfnstPredModeIntra <= 1	0
2 <= lfnstPredModeIntra <= 12	1
13 <= lfnstPredModeIntra <= 23	2
24 <= lfnstPredModeIntra <= 44	3
45 <= lfnstPredModeIntra <= 55	2
56 <= lfnstPredModeIntra <= 80	1
81 <= lfnstPredModeIntra <= 83	0

20

【0130】

表2のように、イントラ予測モードによって、4個の変換セットのうちいずれか一つ、即ち、lfnstTrSetIdxが0から3、即ち、4個のうちいずれか一つにマッピングされることができる。

30

【0131】

一方、非分離変換に特定セットが使われると決定される場合、非分離2次変換インデックスを介して前記特定セット内のk個の変換カーネルの中から一つが選択されることができる。エンコーディング装置は、RD (r a t e - d i s t o r t i o n) チェックに基づいて特定変換カーネルを指す非分離2次変換インデックスを導出することができ、前記非分離2次変換インデックスをデコーディング装置にシグナリングできる。デコーディング装置は、前記非分離2次変換インデックスに基づいて特定セット内のk個の変換カーネルの中から一つを選択することができる。例えば、lfnstインデックス値0は、1番目の非分離2次変換カーネルを指すことができ、lfnstインデックス値1は、2番目の非分離2次変換カーネルを指すことができ、lfnstインデックス値2は、3番目の非分離2次変換カーネルを指すことができる。または、lfnstインデックス値0は、対象ブロックに対して1番目の非分離2次変換が適用されないことを指すことができ、lfnstインデックス値1乃至3は、前記3個の変換カーネルを指すことができる。

40

【0132】

変換部は、選択された変換カーネルに基づいて前記非分離2次変換を実行して修正された(2次)変換係数を取得することができる。前記修正された変換係数は、前述したように、量子化部を介して量子化された変換係数として導出されることができ、エンコーディングされてデコーディング装置にシグナリング及びエンコーディング装置内の逆量子化/逆変換部に伝達されることができる。

50

【0133】

一方、前述したように、2次変換が省略される場合、前記1次（分離）変換の出力である（1次）変換係数が、前述したように、量子化部を介して量子化された変換係数として導出されることができ、エンコーディングされてデコーディング装置にシグナリング及びエンコーディング装置内の逆量子化／逆変換部に伝達されることができる。

【0134】

逆変換部は、前述した変換部で実行された手順の逆順に一連の手順を実行することができる。逆変換部は、（逆量子化された）変換係数を受信し、2次（逆）変換を実行して（1次）変換係数を導出し（S550）、前記（1次）変換係数に対して1次（逆）変換を実行してレジデュアルブロック（レジデュアルサンプルら）を取得することができる（S560）。ここで、前記1次変換係数は、逆変換部立場で修正された（modified）変換係数と呼ばれることができる。エンコーディング装置及びデコーディング装置は、前記レジデュアルブロックと予測されたブロックとに基づいて復元ブロックを生成し、これに基づいて復元ピクチャを生成することができることは、前述の通りである。

10

【0135】

一方、デコーディング装置は、2次逆変換適用可否決定部（または、2次逆変換の適用可否を決定する要素）と、2次逆変換決定部（または、2次逆変換を決定する要素）をさらに含むことができる。2次逆変換適用可否決定部は、2次逆変換の適用可否を決定することができる。例えば、2次逆変換は、NST、RSTまたはLFNSTであり、2次逆変換適用可否決定部は、ビットストリームからパーシングした2次変換フラグに基づいて2次逆変換の適用可否を決定することができる。他の一例として、2次逆変換適用可否決定部は、レジデュアルブロックの変換係数に基づいて2次逆変換の適用可否を決定することもできる。

20

【0136】

2次逆変換決定部は、2次逆変換を決定することができる。このとき、2次逆変換決定部は、イントラ予測モードによって指定されたLFNST（NSTまたはRST）変換セットに基づいて現在ブロックに適用される2次逆変換を決定することができる。また、一実施例として、1次変換決定方法に依存的に（depend on）2次変換決定方法が決定されることができる。イントラ予測モードによって、1次変換と2次変換の多様な組み合わせが決定されることができる。また、一例として、2次逆変換決定部は、現在ブロックの大きさに基づいて2次逆変換が適用される領域を決定することもできる。

30

【0137】

一方、前述したように、2次（逆）変換が省略される場合、（逆量子化された）変換係数を受信して前記1次（分離）逆変換を実行することでレジデュアルブロック（レジデュアルサンプル）を取得することができる。エンコーディング装置及びデコーディング装置は、前記レジデュアルブロックと予測されたブロックに基づいて復元ブロックを生成し、これに基づいて復元ピクチャを生成することができることは、前述の通りである。

【0138】

一方、本文書では非分離2次変換による計算量とメモリ要求量の低減のために、NSTの概念で変換マトリクス（カーネル）の大きさが減少されたRST（reduced secondary transform）を適用することができる。

40

【0139】

一方、本文書で説明された変換カーネル、変換マトリクス、変換カーネルマトリクスを構成する係数、即ち、カーネル係数またはマトリクス係数は8ビットで表現されることができる。これはデコーディング装置及びエンコーディング装置で具現されるための一つの条件であり、既存の9ビットまたは10ビットと比較して合理的に受け入れることができる性能低下と共に、変換カーネルを格納するためのメモリ要求量を減らすことができる。また、カーネルマトリクスを8ビットで表現することによって小さい掛け算器を使用することができ、最適のソフトウェア具現のために使われるSIMD（Single Instruction Multiple Data）命令にさらに適合できる。

50

【0140】

本明細書において、RSTは、簡素化ファクタ (factor) によって大きさが減少された変換マトリクス (transform matrix) に基づいて、対象ブロックに対するレジデュアルサンプルに対して実行される変換を意味することができる。簡素化変換を実行する場合、変換マトリクスの大きさ減少によって変換時に要求される演算量が減少されることができる。即ち、RSTは、大きさが大きいブロックの変換または非分離変換時に発生する演算複雑度 (complexity) 問題を解消するために利用されることができる。

【0141】

RSTは、減少された変換、減少変換、reduced transform、reduced secondary transform、reduction transform、simplified transform、simple transform など、多様な用語で呼ばれることができ、RSTと呼ばれる名称は、羅列された例示に限定されるものではない。または、RSTは、主に変換ブロックで0でない係数を含む低周波領域で行われるため、LFNST (Low-Frequency Non-Separable Transform) と呼ばれることもできる。前記変換インデックスは、LFNSTインデックスと命名されることができる。

10

【0142】

一方、2次逆変換がRSTに基づいて行われる場合、エンコーディング装置200の逆変換部235とデコーディング装置300の逆変換部322は、変換係数に対する逆RSTに基づいて修正された変換係数を導出する逆RST部と、修正された変換係数に対する逆1次変換に基づいて前記対象ブロックに対するレジデュアルサンプルを導出する逆1次変換部と、を含むことができる。逆1次変換は、レジデュアルに適用された1次変換の逆変換を意味する。本文書で変換に基づいて変換係数を導出することは、該当変換を適用して変換係数を導出するのを意味することができる。

20

【0143】

図7は、本文書の一実施例に係るRSTを説明するための図である。

【0144】

本明細書において、「対象ブロック」は、コーディングが実行される現在ブロックまたはレジデュアルブロックまたは変換ブロックを意味することができる。

30

【0145】

一実施例に係るRSTで、N次元ベクトル (N dimensional vector) が異なる空間に位置したR次元ベクトル (R dimensional vector) にマッピングされて減少された変換マトリクスが決定されることができ、ここで、Rは、Nより小さい。Nは、変換が適用されるブロックの一辺の長さ (length) の自乗または変換が適用されるブロックと対応される変換係数の総個数を意味することができ、簡素化ファクタは、R/N値を意味することができる。簡素化ファクタは、減少されたファクタ、減少ファクタ、reduced factor、reduction factor、simplified factor、simple factorなど、多様な用語で呼ばれることができる。一方、Rは、簡素化係数 (reduced coefficient) と呼ばれることができるが、場合によっては、簡素化ファクタがRを意味することもできる。また、場合によって、簡素化ファクタは、N/R値を意味することもできる。

40

【0146】

一実施例において、簡素化ファクタまたは簡素化係数は、ビットストリームを介してシグナリングされることができるが、実施例がこれに限定されるものではない。例えば、簡素化ファクタまたは簡素化係数に対する既定義された値が各エンコーディング装置200及びデコーディング装置300に格納されている場合があり、この場合、簡素化ファクタまたは簡素化係数は、別途にシグナリングされない。

【0147】

一実施例に係る簡素化変換マトリクスのサイズは、通常の変換マトリクスのサイズ $N \times$

50

Nより小さいR × Nであり、以下の数式4のように定義されることができる。

【0148】

【数4】

$$T_{R \times N} = \begin{bmatrix} t_{11} & t_{12} & t_{13} & \dots & t_{1N} \\ t_{21} & t_{22} & t_{23} & \dots & t_{2N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{R1} & t_{R2} & t_{R3} & \dots & t_{RN} \end{bmatrix}$$

【0149】

図7の(a)に示すReduced Transformブロック内のマトリクスTは、数式4のマトリクスT_{R × N}を意味することができる。図7の(a)のように、対象ブロックに対するレジデュアルサンプルに対して簡素化変換マトリクスT_{R × N}が掛けられる場合、対象ブロックに対する変換係数が導出されることができる。

【0150】

一実施例において、変換が適用されるブロックのサイズが8 × 8であり、R = 16（即ち、R / N = 16 / 64 = 1 / 4）である場合、図7の(a)によるRSTは、以下の数式5のような行列演算で表現されることができる。この場合、メモリと掛け演算が簡素化ファクタにより概略1 / 4に減少できる。

【0151】

本文書において、行列演算とは、行列を列ベクトルの左側に置いて行列と列ベクトルをかけて列ベクトルを得る演算と理解されることができる。

【0152】

【数5】

$$\begin{bmatrix} t_{1,1} & t_{1,2} & t_{1,3} & \dots & t_{1,64} \\ t_{2,1} & t_{2,2} & t_{2,3} & \dots & t_{2,64} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{16,1} & t_{16,2} & t_{16,3} & \dots & t_{16,64} \end{bmatrix} \times \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_{64} \end{bmatrix}$$

【0153】

数式5において、r₁乃至r₆₄は、対象ブロックに対するレジデュアルサンプルを示すことができ、より具体的に、1次変換を適用して生成された変換係数である。数式5の演算結果、対象ブロックに対する変換係数c_iが導出されることができる。c_iの導出過程は、数式6の通りである。

【0154】

【数6】

```

for i from 1 to R:
    ci = 0
    for j from 1 to N:
        ci += ti,j * rj

```

【0155】

数式6の演算結果、対象ブロックに対する変換係数c₁乃至c_Rが導出されることがで

10

20

30

40

50

きる。即ち、 $R = 16$ である場合、対象ブロックに対する変換係数 c_1 乃至 c_{16} が導出されることができる。もし、 RST ではなく通常の (regular) 変換が適用されてサイズが 64×64 ($N \times N$) である変換マトリクスが、サイズが 64×1 ($N \times 1$) であるレジデュアルサンプルに掛けられた場合、対象ブロックに対する変換係数が 64 個 (N 個) が導出されるが、 RST が適用されたため、対象ブロックに対する変換係数が 16 個 (R 個) のみ導出される。対象ブロックに対する変換係数の総個数が N 個から R 個に減少してエンコーディング装置 200 がデコーディング装置 300 に送信するデータの量が減少するため、エンコーディング装置 200 - デコーディング装置 300 間の送信効率が増加できる。

【0156】

変換マトリクスのサイズ観点で検討すると、通常の変換マトリクスのサイズは 64×64 ($N \times N$) であり、簡素化変換マトリクスのサイズは 16×64 ($R \times N$) に減少するため、通常の変換を実行する時と比較すると、 RST を実行する時にメモリ使用を R/N 割合に減少させることができる。また、通常の変換マトリクスを利用する時の掛け算演算数 $N \times N$ と比較すると、簡素化変換マトリクスを利用する場合、掛け算演算数を R/N 割合に減少 ($R \times N$) させることができる。

【0157】

一実施例において、エンコーディング装置 200 の変換部 232 は、対象ブロックに対するレジデュアルサンプルを 1 次変換及び RST ベースの 2 次変換を実行することによって対象ブロックに対する変換係数を導出することができる。このような変換係数は、デコーディング装置 300 の逆変換部に伝達されることができ、デコーディング装置 300 の逆変換部 322 は、変換係数に対する逆 RST (reduced secondary transform) に基づいて修正された変換係数を導出し、修正された変換係数に対する逆 1 次変換に基づいて対象ブロックに対するレジデュアルサンプルを導出することができる。

【0158】

一実施例に係る逆 RST マトリクス $T_{N \times R}$ のサイズは、通常の変換マトリクスのサイズ $N \times N$ より小さい $N \times R$ であり、数式 4 に示す簡素化変換マトリクス $T_{R \times N}$ とトランスポーズ (transpose) 関係にある。

【0159】

図 7 の (b) に示す Reduced Inv. Transform ブロック内のマトリクス T^T は、逆 RST マトリクス $T_{R \times N}^T$ を意味することができる (上付き文字 T は、トランスポーズを意味する)。図 7 の (b) のように、対象ブロックに対する変換係数に対して逆 RST マトリクス $T_{R \times N}^T$ が掛けられる場合、対象ブロックに対する修正された変換係数または対象ブロックに対するレジデュアルサンプルが導出されることができる。逆 RST マトリクス $T_{R \times N}^T$ は、 $(T_{R \times N})^T_{N \times R}$ で表現することもできる。

【0160】

より具体的に、2 次逆変換として逆 RST が適用される場合には、対象ブロックに対する変換係数に対して逆 RST マトリクス $T_{R \times N}^T$ が掛けられると、対象ブロックに対する修正された変換係数が導出されることができる。一方、逆 1 次変換として逆 RST が適用されることができ、この場合、対象ブロックに対する変換係数に対して逆 RST マトリクス $T_{R \times N}^T$ が掛けられると、対象ブロックに対するレジデュアルサンプルが導出されることができる。

【0161】

一実施例において、逆変換が適用されるブロックのサイズが 8×8 であり、 $R = 16$ (即ち、 $R/N = 16/64 = 1/4$ である場合) である場合、図 7 の (b) による RST は、以下の数式 7 のような行列演算で表現されることができる。

【0162】

【数 7】

10

20

30

40

50

$$\begin{bmatrix} t_{1,1} & t_{2,1} & & t_{16,1} \\ t_{1,2} & t_{2,2} & \dots & t_{16,2} \\ t_{1,3} & t_{2,3} & & t_{16,3} \\ \vdots & & \ddots & \vdots \\ t_{1,64} & t_{2,64} & \dots & t_{16,64} \end{bmatrix} \times \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_{16} \end{bmatrix}$$

【 0 1 6 3 】

数式 7 において、 c_1 乃至 c_{16} は、対象ブロックに対する変換係数を示すことができる。数式 7 の演算結果、対象ブロックに対する修正された変換係数または対象ブロックに対するレジデュアルサンプルを示す r_i が導出されることができ、 r_i の導出過程は、数式 8 の通りである。

10

【 0 1 6 4 】

【 数 8 】

```

For i from 1 to N
  ri=0
  for j from 1 to R
    ri +=tji * cj

```

20

【 0 1 6 5 】

数式 8 の演算結果、対象ブロックに対する修正された変換係数または対象ブロックに対するレジデュアルサンプルを示す r_1 乃至 r_N が導出されることができ。逆変換マトリクスのサイズ観点で検討すると、通常の変換マトリクスのサイズは 64×64 ($N \times N$) であり、簡素化逆変換マトリクスのサイズは 64×16 ($N \times R$) に減少するため、通常の変換を実行する時と比較すると、逆 RST を実行する時にメモリ使用を R/N 割合に減少させることができる。また、通常の変換マトリクスを利用する時の掛け算演算数 $N \times N$ と比較すると、簡素化逆変換マトリクスを利用する場合、掛け算演算数を R/N 割合に減少 ($N \times R$) させることができる。

30

【 0 1 6 6 】

一方、 8×8 RST に対しても、表 2 のような変換セット構成を適用することができる。即ち、表 2 での変換セットによって該当 8×8 RST が適用されることができる。一つの変換セットは、画面内の予測モードによって 2 個または 3 個の変換 (カーネル) で構成されているため、2 次変換を適用しないとまで含んで最大 4 個の変換の中から一つを選択するように構成されることができる。2 次変換を適用しない時の変換は、恒等行列が適用されたと見なされることができる。4 個の変換に対して各々 0、1、2、3 のインデックスを付与するとした時 (例えば、0 番インデックスを恒等行列、即ち、2 次変換を適用しない場合で割り当てることができる)、変換インデックスまたは $lfnst$ インデックスというシンタックス要素 (syntax element) を変換係数ブロック毎にシグナリングして適用される変換を指定することができる。即ち、変換インデックスを介して 8×8 左上端ブロックに対して、RST 構成では 8×8 RST を指定することができる、または LFNST が適用される場合、 8×8 $lfnst$ を指定することができる。 8×8 $lfnst$ 及び 8×8 RST は、変換の対象になる対象ブロックの W と H が両方とも 8 より大きいまたは同じ時、該当変換係数ブロック内部に含まれている 8×8 領域に適用されることができる変換を指し、該当 8×8 領域は、該当変換係数ブロック内部の左上端 8×8 領域である。同様に、 4×4 $lfnst$ 及び 4×4 RST は、対象ブロックの W と H が両方とも 4 より大きいまたは同じ時、該当変換係数ブロック内部に含まれてい

40

50

る 4×4 領域に適用されることができる変換を指し、該当 4×4 領域は該当変換係数ブロック内部の左上端 4×4 領域である。

【0167】

一方、本文書の一実施例によって、エンコーディング過程の変換で、 8×8 領域を構成する64個のデータに対して 16×64 変換カーネルマトリクスでない、48個のデータのみを選択して最大 16×48 変換カーネルマトリクスを適用することができる。ここで、「最大」とは、 m 個の係数を生成することができる $m \times 48$ 変換カーネルマトリクスに対して m の最大値が16であることを意味する。即ち、 8×8 領域に $m \times 48$ 変換カーネルマトリクス ($m = 16$) を適用して RST を実行する場合、48個のデータの入力を受けて m 個の係数を生成することができる。 m が16である場合、48個のデータの入力を受けて16個の係数を生成する。即ち、48個のデータが 48×1 ベクトルをなすとした時、 16×48 行列と 48×1 ベクトルをじゅんにかけて 16×1 ベクトルが生成されることができる。このとき、 8×8 領域をなす48個のデータを適切に配列して 48×1 ベクトルを構成することができる。例えば、 8×8 領域のうち右下端 4×4 領域を除外した領域を構成する48個のデータに基づいて 48×1 ベクトルを構成することができる。このとき、最大 16×48 変換カーネルマトリクスを適用して行列演算を実行すると、16個の修正された変換係数が生成され、16個の修正された変換係数は、スキヤニング順序によって左上端 4×4 領域に配置されることができ、右上端 4×4 領域と左下端 4×4 領域は、0で満たされることができる。

10

【0168】

デコーディング過程の逆変換には前記叙述された変換カーネルマトリクスのトランスポーズされたマトリクスが使われることができる。即ち、デコーディング装置で実行される逆変換過程として逆 RST または LFNST が実行される場合、逆 RST を適用する入力係数データは、所定の配列順序によって1次元ベクトルで構成され、1次元ベクトルに該当逆 RST 行列を左側でかけて得られた修正された係数ベクトルを所定の配列順序によって2次元ブロックに配列されることができる。

20

【0169】

整理すると、変換過程で、 8×8 領域に RST または LFNST が適用される場合、 8×8 領域の変換係数のうち、 8×8 領域の右下端領域を除外した左上端、右上端、左下端領域の48個変換係数と 16×48 の変換カーネルマトリクスとの行列演算が実行される。行列演算のために、48個の変換係数は、1次元配列で入力される。このような行列演算が実行されると、16個の修正された変換係数が導出され、修正された変換係数は、 8×8 領域の左上端領域に配列されることができる。

30

【0170】

逆に、逆変換過程で、 8×8 領域に逆 RST または LFNST が適用される場合、 8×8 領域の変換係数のうち 8×8 領域の左上端に対応する16個の変換係数は、スキヤニング順序によって1次元配列形態で入力されて 48×16 の変換カーネルマトリクスと行列演算されることができる。即ち、このような場合の行列演算は、 $(48 \times 16 \text{ 行列}) * (16 \times 1 \text{ 変換係数ベクトル}) = (48 \times 1 \text{ 修正された変換係数ベクトル})$ で表すことができる。ここで、 $n \times 1$ ベクトルは、 $n \times 1$ 行列のような意味で解釈されることができるため、 $n \times 1$ 列ベクトルで表記されることもできる。また、 $*$ は、行列掛け算演算を意味する。このような行列演算が実行される場合、48個の修正された変換係数が導出されることができ、48個の修正された変換係数は、 8×8 領域の右下端領域を除外した左上端、右上端、左下端領域に配列されることができる。

40

【0171】

一方、2次逆変換が RST に基づいて行われる場合、エンコーディング装置 200 の逆変換部 235 とデコーディング装置 300 の逆変換部 322 は、変換係数に対する逆 RST に基づいて修正された変換係数を導出する逆 RST 部と、修正された変換係数に対する逆1次変換に基づいて前記対象ブロックに対するレジデュアルサンプルを導出する逆1次変換部と、を含むことができる。逆1次変換は、レジデュアルに適用された1次変換の逆

50

変換を意味する。本文書において、変換に基づいて変換係数を導出することは、該当変換を適用して変換係数を導出することを意味することができる。

【0172】

詳述された非分離変換、LFNSTに対して具体的にみると、次の通りである。LFNSTは、エンコーディング装置による順方向(forward)変換とデコーディング装置による逆方向(inverse)変換とを含むことができる。

【0173】

エンコーディング装置は、順方向1次変換(primary (core) transform)を適用した後に導出された結果(または、結果の一部)を入力とし、順方向2次変換(secondary transform)を適用する。

10

【0174】

【数9】

$$y = G^T x$$

【0175】

前記数式9において、 x と y は、各々、2次変換の入力と出力であり、 G は、2次変換を示す行列であって、変換基底ベクトル(transform basis vector)は、列ベクトルで構成される。逆方向LFNSTの場合、変換行列 G の次元(dimension)を[row数 × column数]で表記した時、順方向LFNSTの場合、行列 G のトランスポーズを行ったものが G^T の次元になる。

20

【0176】

逆方向LFNSTの場合、行列 G の次元は、[48 × 16]、[48 × 8]、[16 × 16]、[16 × 8]になり、[48 × 8]行列と[16 × 8]行列は、各々、[48 × 16]行列と[16 × 16]行列の左側から8個の変換基底ベクトルをサンプリングした部分行列である。

【0177】

それに対して、順方向LFNSTの場合、行列 G^T の次元は[16 × 48]、[8 × 48]、[16 × 16]、[8 × 16]になり、[8 × 48]行列と[8 × 16]行列は、各々、[16 × 48]行列と[16 × 16]行列の上側から8個の変換基底ベクトルをサンプリングした部分行列である。

30

【0178】

したがって、順方向LFNSTの場合、入力 x としては[48 × 1]ベクトルまたは[16 × 1]ベクトルが可能であり、出力 y としては[16 × 1]ベクトルまたは[8 × 1]ベクトルが可能である。ビデオコーディング及びデコーディングにおける順方向1次変換の出力は、二次元(2D)データであるため、入力 x として[48 × 1]ベクトルまたは[16 × 1]ベクトルを構成するために、順方向変換の出力である2Dデータを適切に配列して1次元ベクトルを構成しなければならない。

【0179】

図8は、一例によって順方向1次変換の出力データを1次元ベクトルに配列する順序を示す図である。図8の(a)及び(b)の左側図は、[48 × 1]ベクトルを作成するための順序を示し、図8の(a)及び(b)の右側図は、[16 × 1]ベクトルを作成するための順序を示す。LFNSTの場合、図8の(a)及び(b)のような順序に、2Dデータを順次に配列して1次元ベクトル x を得ることができる。

40

【0180】

このような順方向1次変換の出力データの配列方向は、現在ブロックのイントラ予測モードによって決定されることができる。例えば、現在ブロックのイントラ予測モードが対角線方向を基準にして水平方向である場合、順方向1次変換の出力データは、図8の(a)の順序に配列されることができ、現在ブロックのイントラ予測モードが対角線方向を基

50

準にして垂直方向である場合、順方向 1 次変換の出力データは、図 8 の (b) の順序に配列されることができる。

【 0 1 8 1 】

一例によって、図 8 の (a) 及び (b) の配列順序 (o r d e r i n g) と異なる配列順序を適用することができ、図 8 の (a) 及び (b) の配列順序を適用した時と同じ結果 (y ベクトル) を導出しようとする場合は、行列 G の列ベクトルを該当配列順序に合わせて再配列すればよい。即ち、x ベクトルを構成する各要素に対して常に同じ変換基底ベクトルと掛けられるように G の列ベクトルを再配置することができる。

【 0 1 8 2 】

数式 9 を介して導出される出力 y は、一次元ベクトルであるため、もし、順方向 2 次変換の結果を入力として処理する構成、例えば、量子化またはレジデュアルコーディングを実行する構成が入力データとして 2 次元データが必要である場合、数式 9 の出力 y ベクトルは、再び 2 D データで適切に配置されなければならない。

10

【 0 1 8 3 】

図 9 は、一例によって順方向 2 次変換の出力データを 2 次元ブロックに配列する順序を示す図である。

【 0 1 8 4 】

L F N S T の場合、決められたスキャン順序によって 2 D ブロックに配置されることができる。図 9 の (a) は、出力 y が [1 6 × 1] ベクトルである場合、2 次元ブロックの 1 6 個の位置に対角スキャン (d i a g o n a l s c a n) 順序によって出力値が配置されることを示す。図 9 の (b) は、出力 y が [8 × 1] ベクトルである場合、2 次元ブロックの 8 個の位置に対角スキャン順序によって出力値が配置され、残り 8 個の位置には 0 で満たされることを示す。図 9 の (b) の X は、0 で満たされることを示す。

20

【 0 1 8 5 】

他の例によって、量子化またはレジデュアルコーディングを実行する構成により出力ベクトル y が処理される順序は、既設定された順序によって実行できるため、図 9 のように出力ベクトル y が 2 D ブロックに配置されないことがある。ただし、レジデュアルコーディングの場合、C G (C o e f f i c i e n t G r o u p) のような 2 D ブロック (例えば、4 × 4) 単位でデータコーディングが実行されることができ、この場合、図 9 の対角スキャン順序のように特定順序によってデータが配列されることができる。

30

【 0 1 8 6 】

一方、デコーディング装置は、逆方向変換のために逆量子化過程などを介して出力された 2 次元データを既設定されたスキャン順序によって羅列して 1 次元入力ベクトルである y を構成することができる。入力ベクトル y は、下記数式により入力ベクトル x として出力されることができる。

【 0 1 8 7 】

【 数 1 0 】

$$x = G y$$

40

【 0 1 8 8 】

逆方向 L F N S T の場合、[1 6 × 1] ベクトルまたは [8 × 1] ベクトルである入力ベクトル y に G 行列を掛けることによって、出力ベクトル x を導出することができる。逆方向 L F N S T の場合、出力ベクトル x は、[4 8 × 1] ベクトルまたは [1 6 × 1] ベクトルである。

【 0 1 8 9 】

出力ベクトル x は、図 8 に示す順序によって 2 次元ブロックに配置されて 2 次元データで配列され、このような 2 次元データは、逆方向 1 次変換の入力データ (または、入力データの一部分) になる。

50

【0190】

したがって、逆方向2次変換は、全体的に順方向2次変換過程と反対であり、逆変換の場合、順方向と違って、逆方向2次変換を先に適用した後に逆方向1次変換を適用するようになる。

【0191】

逆方向LFNSTでは変換行列Gとして[48×16]行列8個と[16×16]行列8個の中から一つが選択されることができる。[48×16]行列と[16×16]行列のうちどの行列を適用するかは、ブロックの大きさともようによって決定される。

【0192】

また、8個の行列は、前述された表2のように4個の変換セットから導出されることができ、各変換セットは、2個の行列で構成されることができる。4個の変換セットのうちどの変換セットを使用するかは、イントラ予測モードによって決定され、より具体的に、広角イントラ予測モード(Wide Angle Intra Prediction、WAIP)まで考慮して拡張されたイントラ予測モード値に基づいて変換セットが決定される。選択された変換セットを構成する2個の行列の中からどの行列を選択するかは、インデックスシグナリング(index signaling)を介して導出される。より具体的に、送信されるインデックス値では0、1、2が可能であり、0は、LFNSTを適用しないことを指示し、1と2は、イントラ予測モード値に基づいて選択された変換セットを構成する2個の変換行列のうちいずれか一つを指示することができる。

10

【0193】

図10は、本文書の一実施例に係る広角イントラ予測モードを示す図である。

20

【0194】

一般的なイントラ予測モード値は、0～66と81～83までの値を有することができ、図示されたように、WAIPによって拡張されたイントラ予測モード値は、-14～83までの値を有することができる。81～83までの値は、CCLM(Cross Component Linear Model)モードを指し、-14～-1までの値と67～80までの値は、WAIP適用によって拡張されたイントラ予測モード値を指す。

【0195】

予測現在ブロックの幅が高さより大きい場合、一般的に上側参照ピクセルが予測しようとするブロック内部の位置ともっと近い。したがって、右上端(top-right)方向に予測することより左下端(bottom-left)方向に予測することがより正確である。それに対して、ブロックの高さが幅より大きい場合は、左側参照ピクセルが予測しようとするブロック内部の位置と一般的に近い。したがって、左下端(bottom-left)方向に予測することより右上端(top-right)方向に予測することがより正確である。したがって、広角イントラ予測モードのインデックスでリマッピング、即ち、モードインデックス変換を適用することが有利である。

30

【0196】

広角イントラ予測が適用される場合、既存のイントラ予測に対する情報がシグナリングされることができ、前記情報がパースングされた以後、前記情報が前記広角イントラ予測モードのインデックスでリマッピングされることができる。したがって、特定ブロック(例えば、特定サイズの非正方形ブロック)に対する総イントラ予測モードの数は変更されなく、即ち、総イントラ予測モードの数は67個であり、前記特定ブロックに対するイントラ予測モードコーディングは変更されない。

40

【0197】

以下の表3は、イントラ予測モードを広角イントラ予測モードにリマッピングして修正されたイントラモードを導出する過程を示している。

【0198】

50

【表 3】

<p>Inputs to this process are:</p> <ul style="list-style-type: none"> - a variable <code>predModeIntra</code> specifying the intra prediction mode, - a variable <code>nTbW</code> specifying the transform block width, - a variable <code>nTbH</code> specifying the transform block height, - a variable <code>cIdx</code> specifying the colour component of the current block. <p>Output of this process is the modified intra prediction mode <code>predModeIntra</code>. The variables <code>nW</code> and <code>nH</code> are derived as follows:</p> <ul style="list-style-type: none"> - If <code>IntraSubPartitionsSplitType</code> is equal to <code>ISP_NO_SPLIT</code> or <code>cIdx</code> is not equal to 0, the following applies: <ul style="list-style-type: none"> $nW = nTbW$ (8-97) $nH = nTbH$ (8-98) - Otherwise (<code>IntraSubPartitionsSplitType</code> is not equal to <code>ISP_NO_SPLIT</code> and <code>cIdx</code> is equal to 0), the following applies: <ul style="list-style-type: none"> $nW = nCbW$ (8-99) $nH = nCbH$ (8-100) <p>The variable <code>whRatio</code> is set equal to $Abs(\text{Log}_2(nW / nH))$.</p> <p>For non-square blocks (<code>nW</code> is not equal to <code>nH</code>), the intra prediction mode <code>predModeIntra</code> is modified as follows:</p> <ul style="list-style-type: none"> - If all of the following conditions are true, <code>predModeIntra</code> is set equal to (<code>predModeIntra + 65</code>). <ul style="list-style-type: none"> - <code>nW</code> is greater than <code>nH</code> - <code>predModeIntra</code> is greater than or equal to 2 - <code>predModeIntra</code> is less than ($whRatio > 1$) ? ($8 + 2 * whRatio$) : 8 - Otherwise, if all of the following conditions are true, <code>predModeIntra</code> is set equal to (<code>predModeIntra - 67</code>). <ul style="list-style-type: none"> - <code>nH</code> is greater than <code>nW</code> - <code>predModeIntra</code> is less than or equal to 66 - <code>predModeIntra</code> is greater than ($whRatio > 1$) ? ($60 - 2 * whRatio$) : 60 	<p>10</p> <p>20</p> <p>30</p>
---	-------------------------------

【 0 1 9 9 】

表 3 において、最終的に `predModeIntra` 変数に拡張されたイントラ予測モード値が格納され、`ISP_NO_SPLIT` は、現在 VVC 標準に採択された `IntraSubPartitions (ISP)` 技術により CU ブロックがサブパーティションに分割されないことを示し、`cIdx` 変数値が 0、1、2 であることは、各々、ルマ、Cb、Cr コンポーネントである場合を指す。表 3 に示す Log_2 関数は、ベース (base) が 2 であるログ値をリターンし、`Abs` 関数は、絶対値をリターンする。

【 0 2 0 0 】

広角イントラ予測モードのマッピング過程 (`Wide angle intra prediction mode mapping process`) の入力値として、イントラ予測モードを指示する変数 `predModeIntra`、変換ブロックの高さ及び幅などが使われ、出力値は、修正されたイントラ予測モード (`the modified intra prediction mode predModeIntra`) になる。変換ブロックまたはコーディングブロックの高さ及び幅が、イントラ予測モードのリマッピングのための現在ブロックの高さ及び幅になることができる。このとき、幅と高の比率を反映する変数 `whRatio` は、 $Abs(\text{Log}_2(nW / nH))$ に設定されることができる。

10

20

30

40

50

【0201】

正方形でないブロックに対して、イントラ予測モードは、二つの場合に区分されて修正されることができる。

【0202】

まず、(1) 現在ブロックの幅が高さより大きい、(2) 修正前のイントラ予測モードが2より大きいまたは同じ、(3) イントラ予測モードが、変数 $whRatio$ が1より大きい場合は $(8 + 2 * whRatio)$ であり、変数 $whRatio$ が1より小さいまたは同じ場合は8であって、導出される値より小さい $[predModeIntra \leq (whRatio - 1) ? (8 + 2 * whRatio) : 8]$ という全ての条件を満たす場合、イントラ予測モードは、イントラ予測モードより65大きい値に設定される $[predModeIntra \leq (predModeIntra + 65)]$ 。

10

【0203】

前記と異なる場合、(1) 現在ブロックの高さが幅より大きい、(2) 修正前のイントラ予測モードが66より小さいまたは同じ、(3) イントラ予測モードが、変数 $whRatio$ が1より大きい場合は $(60 - 2 * whRatio)$ であり、変数 $whRatio$ が1より小さいまたは同じ場合は60であって、導出される値より大きい $[predModeIntra \geq (whRatio - 1) ? (60 - 2 * whRatio) : 60]$ という全ての条件を満たす場合、イントラ予測モードは、イントラ予測モードより67小さい値に設定される $[predModeIntra \leq (predModeIntra - 67)]$ 。

20

【0204】

前述された表2は、LFNSTでWAIPにより拡張されたイントラ予測モード値に基づいて変換セットがどのように選択されるかを示している。図10のように、14~33までのモードと35~80までのモードは、モード34を中心にして予測方向観点で互いに対称である。例えば、モード14とモード54は、モード34に該当する方向を中心にして対称である。したがって、互いに対称である方向に位置するモードどうしは同じ変換セットを適用するようになり、表2でもこのような対称性が反映されている。

【0205】

ただし、モード54に対する順方向LFNST入力データは、モード14に対する順方向LFNST入力データと対称を成すことを仮定する。例えば、モード14とモード54に対しては、各々、図8の(a)と図8の(b)に示す配列順序によって2次元データを1次元データで再配列するようになり、図8の(a)と図8の(b)に示す順序のパターンは、モード34が指す方向(対角線方項)を中心にして対称であることを知ることができる。

30

【0206】

一方、前述したように、 $[48 \times 16]$ 行列と $[16 \times 16]$ 行列のうちどの変換行列をLFNSTに適用するかは、変換対象ブロックの大きさともようにより決定される。

【0207】

図11は、LFNSTが適用されるブロックもようを示す図である。図11の(a)は 4×4 ブロックを、(b)は 4×8 及び 8×4 ブロックを、(c)は N が16以上である $4 \times N$ または $N \times 4$ ブロックを、(d)は 8×8 ブロックを、(e)は $M = 8$ 、 $N = 8$ であり、 $N = 8$ または $M = 8$ である $M \times N$ ブロックを示している。

40

【0208】

図11で太い枠のブロックがLFNSTが適用される領域を指す。図11の(a)及び(b)のブロックに対しては左上端(top-left) 4×4 領域に対してLFNSTが適用され、図11の(c)のブロックに対しては連続配置された2個の左上端 4×4 領域に対して各々LFNSTが適用される。図11の(a)、(b)、(c)では 4×4 領域単位でLFNSTが適用されるため、このようなLFNSTを以下「 4×4 LFNST」と命名するようにし、該当変換行列では数式9及び数式10のGに対する行列次元を基

50

準 $[16 \times 16]$ または $[16 \times 8]$ 行列が適用されることができる。

【0209】

より具体的に、図11の(a)の 4×4 ブロック(4×4 TUまたは 4×4 CU)に対しては $[16 \times 8]$ 行列が適用され、図11の(b)及び(c)のブロックに対しては $[16 \times 16]$ 行列が適用される。これは最悪の場合(worst case)に対する計算複雑度をサンプル当たり8乗算(8 multiplications per sample)に合わせるためである。

【0210】

図11の(d)及び(e)に対しては左上端 8×8 領域に対してLFNSTが適用され、このようなLFNSTを以下「 8×8 LFNST」と命名するようにする。該当変換行列では $[48 \times 16]$ または $[48 \times 8]$ 行列が適用されることができる。順方向LFNSTの場合、入力データとして $[48 \times 1]$ ベクトル(数式9のxベクトル)が入力されるため、左上端 8×8 領域の全てのサンプル値が順方向LFNSTの入力値として使われない。即ち、図8の(a)の左側順序または図8の(b)の左側順序で見ることができるよう、右下端(bottom-right)の 4×4 ブロックはそのまま置き、残り3個の 4×4 ブロックに属するサンプルに基づいて $[48 \times 1]$ ベクトルを構成することができる。

10

【0211】

図11の(d)での 8×8 ブロック(8×8 TUまたは 8×8 CU)に $[48 \times 8]$ 行列が適用され、図11の(e)での 8×8 ブロックに $[48 \times 16]$ 行列が適用されることができる。これも最悪の場合(worst case)に対する計算複雑度をサンプル当たり8乗算(8 multiplications per sample)に合わせたためである。

20

【0212】

ブロックもようによって、これに対応する順方向LFNST(4×4 LFNSTまたは 8×8 LFNST)が適用されると、8個または16個の出力データ(数式9でのyベクトル、 $[8 \times 1]$ または $[16 \times 1]$ ベクトル)が生成され、順方向LFNSTでは行列GTの特性上、出力データの数が入力データの数と同じまたは少なくなる。

【0213】

図12は、一例によって順方向LFNSTの出力データの配列を示し、ブロックもようによって順方向LFNSTの出力データが配置されるブロックを示す図である。

30

【0214】

図12に示すブロックの左上端に陰影で処理された領域が順方向LFNSTの出力データが位置する領域に該当し、0で表記された位置は0値で満たされるサンプルを示し、残り領域は、順方向LFNSTにより変更されない領域を示す。LFNSTにより変更されない領域には順方向1次変換の出力データが変更されずにそのまま存在する。

【0215】

前述したように、ブロックもようによって適用される変換行列の次元が変わるため、出力データの数も変わる。図12のように、順方向LFNSTの出力データが左上端 4×4 ブロックを全部満たすことができない場合もある。図12の(a)及び(d)の場合、太線で表示されたブロックまたはブロック内部の一部領域には、各々、 $[16 \times 8]$ 行列と $[48 \times 8]$ 行列が適用されて順方向LFNSTの出力として $[8 \times 1]$ ベクトルが生成される。即ち、図9の(b)に示すスキャン順序によって8個の出力データのみが図12の(a)及び(d)のように満たされ、残り8個の位置に対しては0で満たされることができる。図11の(d)のLFNST適用ブロックの場合、図12の(d)のように左上端 4×4 ブロックに隣接した右上端及び左下端の二つの 4×4 ブロックも0値で満たされる。

40

【0216】

前記のように、基本的にLFNSTインデックスをシグナリングしてLFNST適用可否及び適用する変換行列を指定するようになる。図12に示すように、LFNSTが適用

50

される場合、順方向 L F N S T の出力データ数が入力データ数と同じまたは少ないことがあるため、0 値で満たされる領域が下記のように発生する。

【 0 2 1 7 】

1) 図 1 2 の (a) のように左上端 4×4 ブロック内にスキャン順序上 8 番目以後の位置、即ち、9 番目から 1 6 番目までのサンプル

【 0 2 1 8 】

2) 図 1 2 の (d) 及び (e) のように、 $[1 6 \times 4 8]$ 行列または $[8 \times 4 8]$ 行列が適用されて左上端 4×4 ブロックに隣接した二つの 4×4 ブロックまたはスキャン順序上 2 番目と 3 番目の 4×4 ブロック

【 0 2 1 9 】

したがって、前記 1) と 2) の領域をチェックして 0 でない (n o n - z e r o) データが存在するようになると、L F N S T が適用されないことが確実であるため、該当 L F N S T インデックスのシグナリングを省略することができるようになる。

【 0 2 2 0 】

一例によって、例えば、V V C 標準に採択された L F N S T の場合、L F N S T インデックスのシグナリングは、レジデュアルコーディング以後に実行されるため、エンコーディング装置は、レジデュアルコーディングを介して T U または C U ブロック内部の全ての位置に対する 0 でないデータ (有効係数) の存在可否を知ることができるようになる。したがって、エンコーディング装置は、0 でないデータ存在可否を介して L F N S T インデックスに対するシグナリングを実行するかどうかを判断することができ、デコーディング装置は、L F N S T インデックスのパーシング可否を判断することができる。もし、前記 1) と 2) で指定された領域に 0 でないデータが存在しない場合、L F N S T インデックスのシグナリングを実行するようになる。

【 0 2 2 1 】

一方、採択された L F N S T に対して、下記のような単純化方法が適用されることができる。

【 0 2 2 2 】

(i) 一例によって、順方向 L F N S T に対する出力データ数を最大 1 6 個に限定できる。

【 0 2 2 3 】

図 1 1 の (c) の場合、左上端に隣接した 2 個の 4×4 領域に各々 4×4 L F N S T が適用されることができ、このとき、最大 3 2 個の L F N S T 出力データが生成されることができる。もし、順方向 L F N S T に対する出力データ数を最大 1 6 に限定する場合、 $4 \times N / N \times 4$ ($N = 1 6$) ブロック (T U または C U) に対しても左上端に存在する 1 個の 4×4 領域に対してのみ 4×4 L F N S T を適用し、図 1 1 の全てのブロックに対して L F N S T を 1 回のみ適用できる。これを介して映像コーディングに対する具現が単純になる。

【 0 2 2 4 】

(ii) 一例によって、L F N S T が適用されない領域に対して追加的にゼロアウト (z e r o - o u t) を適用することができる。本文書において、ゼロアウトは、特定領域に属する全ての位置の値を 0 値で満たすことを意味することができる。即ち、L F N S T によって変更されずに、順方向 1 次変換の結果を維持している領域に対してもゼロアウトを適用することができる。前述したように、L F N S T は、 4×4 L F N S T と 8×8 L F N S T に区分されるため、下記のように 2 種類 ((ii) - (A) 及び (ii) - (B)) にゼロアウトを区分することができる。

【 0 2 2 5 】

(ii) - (A) 4×4 L F N S T が適用される時、 4×4 L F N S T が適用されない領域をゼロアウトすることができる。図 1 3 は、一例によって 4×4 L F N S T が適用されるブロックでのゼロアウトを示す図である。

【 0 2 2 6 】

10

20

30

40

50

図 1 3 のように、 4×4 L F N S T が適用されるブロックに対して、即ち、図 1 2 の (a)、(b)、及び (c) のブロックに対して L F N S T が適用されない領域まで全て 0 で満たされることができる。

【 0 2 2 7 】

一方、図 1 3 の (d) は、一例によって順方向 L F N S T の出力データ個数の最大値を 1 6 に限定した場合、 4×4 L F N S T が適用されない残りブロックに対してゼロアウトを実行したことを示す。

【 0 2 2 8 】

(ii) - (B) 8×8 L F N S T が適用される時、 8×8 L F N S T が適用されない領域をゼロアウトすることができる。図 1 4 は、一例によって 8×8 L F N S T が適用されるブロックでのゼロアウトを示す図である。

10

【 0 2 2 9 】

図 1 4 のように、 8×8 L F N S T が適用されるブロックに対して、即ち、図 1 2 の (d) 及び (e) のブロックに対して L F N S T が適用されない領域まで全て 0 で満たされることができる。

【 0 2 3 0 】

(iii) 前記 (ii) で提示したゼロアウトによって L F N S T が適用される時、0 で満たされる領域が変わることができる。したがって、前記 (ii) で提案されたゼロアウトによって 0 でないデータが存在するかどうかを図 1 2 の L F N S T の場合より広い領域に対してチェックできる。

20

【 0 2 3 1 】

例えば、(ii) - (B) を適用する場合、図 1 2 の (d) 及び (e) で 0 値で満たされる領域に追加して図 1 4 で追加的に 0 で満たされた領域まで 0 でないデータが存在するかどうかをチェックした後、0 でないデータが存在しない場合にのみ L F N S T インデックスに対するシグナリングを実行することができる。

【 0 2 3 2 】

もちろん、前記 (ii) で提案されたゼロアウトを適用しても、既存 L F N S T インデックスシグナリングと同じく 0 でないデータが存在するかどうかをチェックすることができる。即ち、図 1 2 に 0 で満たされたブロックに対して 0 でないデータが存在するかどうかをチェックして L F N S T インデックスシグナリングを適用することができる。このような場合、エンコーディング装置にのみゼロアウトを実行し、デコーディング装置では該当ゼロアウトを仮定することなく、即ち、図 1 2 で明示的に 0 で表記された領域に対してのみ 0 でないデータが存在するかどうかのみをチェックして L F N S T インデックスパーシングを実行することができる。

30

【 0 2 3 3 】

前記 L F N S T に対する単純化方法 ((i)、(ii) - (A)、(ii) - (B)、(iii)) の組み合わせを適用した多様な実施例が導出されることができる。もちろん、前記単純化方法に対する組み合わせは、下記の実施例に限定されるものではなく、任意の組み合わせを L F N S T に適用できる。

【 0 2 3 4 】

実施例

40

【 0 2 3 5 】

- 順方向 L F N S T に対する出力データ数を最大 1 6 個に限定 (i)

【 0 2 3 6 】

- 4×4 L F N S T が適用される時、 4×4 L F N S T が適用されない領域を全てゼロアウト (ii) - (A)

【 0 2 3 7 】

- 8×8 L F N S T が適用される時、 8×8 L F N S T が適用されない領域を全てゼロアウト (ii) - (B)

【 0 2 3 8 】

50

- 既存 0 値で満たされる領域と追加的なゼロアウト ((ii) - (A)、(ii) - (B)) によって 0 で満たされる領域に対しても、0 でないデータが存在するかどうかをチェックした後、0 でないデータが存在しない場合にのみ L F N S T インデクシングシグナリング (iii)

【 0 2 3 9 】

前記実施例の場合、L F N S T が適用される時、0 でない出力データが存在できる領域が左上端 4 × 4 領域内部に制限される。より詳細に、図 1 3 の (a) と図 1 4 の (a) の場合、スキャン順序上、8 番目の位置が 0 でないデータが存在できる最後の位置になり、図 1 3 の (b) 及び (d) と図 1 4 の (b) の場合、スキャン順序上、1 6 番目の位置 (即ち、左上端 4 × 4 ブロックの右下端の最端位置) が 0 でないデータが存在できる最後の位置になる。

10

【 0 2 4 0 】

したがって、L F N S T が適用された時、レジデュアルコーディング過程が許容されない位置 (最後の位置を超えた位置) で 0 でないデータが存在するかどうかをチェックした後、L F N S T インデックスシグナリング可否が決定されることができ。

【 0 2 4 1 】

(ii) で提案されたゼロアウト方式の場合、1 次変換と L F N S T を全て適用した時に最終的に発生するようになるデータの数減らすため、全体変換過程を実行する時に要求される計算量を減らすことができる。即ち、L F N S T が適用される場合、L F N S T が適用されない領域に存在する順方向 1 次変換出力データに対してもゼロアウトを適用するため、順方向 1 次変換を実行する時からゼロアウトになる領域に対するデータを生成する必要がない。したがって、該当データ生成に要求される演算量を節約することができる。(ii) で提案されたゼロアウト方式の追加的な効果を整理すると、下記の通りである。

20

【 0 2 4 2 】

第一、前記のように全体変換過程の実行に必要な計算量が低減される。

【 0 2 4 3 】

特に、(ii) - (B) を適用する場合、最悪の場合に対する計算量が減少して変換過程を軽量化することができる。説明を敷衍すると、一般的に大きいサイズの 1 次変換実行に多い量の演算が要求され、(ii) - (B) を適用するようになると、順方向 L F N S T 実行結果として導出されるデータ数を 1 6 個以下に減らすことができ、全体ブロック (T U または C U) 大きさが大きくなるほど、変換演算量低減効果は一層増加される。

30

【 0 2 4 4 】

第二、変換過程全体に必要な演算量が減少して変換実行に必要な電力消費を減らすことができる。

【 0 2 4 5 】

第三、変換過程に伴われる遅延時間 (l a t e n c y) を減少させる。

【 0 2 4 6 】

L F N S T のような 2 次変換は、既存 1 次変換に計算量を追加するようになるため、変換実行に伴われる全体遅延時間を増加させる。特にイントラ予測の場合、予測過程で隣接ブロックの復元データが使われるため、エンコーディング時に 2 次変換による遅延時間増加が復元 (r e c o n s t r u c t i o n) までの遅延時間増加につながるようになって、イントラ予測エンコーディングの全体的な遅延時間増加につながる事ができる。

40

【 0 2 4 7 】

しかし、(ii) で提示したゼロアウトを適用するようになると、L F N S T 適用時に 1 次変換実行の遅延時間を大幅減らすことができるため、変換実行全体に対する遅延時間は、そのまま維持され、または減るようになって、エンコーディング装置をより簡単に具現できる。

【 0 2 4 8 】

一方、従来のイントラ予測は、現在符号化しようとするブロックを一つの符号化単位と見なして分割無しで符号化を実行した。しかし、I S P (I n t r a S u b - P a r i

50

tions) コーディングは、現在符号化しようとするブロックを水平方向または垂直方向に分割してイントラ予測符号化を実行することを意味する。このとき、分割されたブロック単位で符号化/復号化を実行して復元されたブロックを生成し、復元されたブロックは次に分割されたブロックの参照ブロックとして使われることができる。一例によって、ISP コーディング時に一つのコーディングブロックが2個または4個のサブブロックに分割されてコーディングされることができ、ISP で一つのサブブロックは、隣接した左側または隣接した上側に位置したサブブロックの復元されたピクセル値を参照してイントラ予測が実行される。以下、使われる「コーディング」は、エンコーディング装置で実行されるコーディングとデコーディング装置で実行されるデコーディングを全て含む概念で使われることができる。

10

【0249】

ISP は、ブロックのサイズによってルマイントラで予測されたブロックを垂直方向または水平方向に2または4個のサブパーティショニングに分割することである。例えば、ISP が適用されることができるとする最小ブロックサイズは、 4×8 または 8×4 である。もし、ブロックサイズが 4×8 または 8×4 より大きい場合、ブロックは、4個のサブパーティショニングに分割される。

【0250】

ISP 適用時、サブブロックは、分割形態によって、例えば、水平 (Horizontal) または垂直 (Vertical)、左側から右側または上側から下側に順次にコーディングされ、一つのサブブロックに対する逆変換とイントラ予測を経て復元過程まで実行された後、次のサブブロックに対するコーディングが進行されることができ、最も左側または最も上側サブブロックに対しては通常的なイントラ予測方式のように、既にコーディングされたコーディングブロックの復元ピクセルを参照するようになる。また、後に続いた内部のサブブロックの各辺に対して以前サブブロックと隣接しない場合には該当辺に隣接した参照ピクセルを導出するために、通常的なイントラ予測方式のように、既にコーディングされた隣接したコーディングブロックの復元ピクセルを参照する。

20

【0251】

ISP コーディングモードでは全てのサブブロックが同じイントラ予測モードでコーディングされることができ、ISP コーディングを使用するかどうかを示すフラグとどの方向に (水平または垂直) 分割するかを示すフラグなどがシグナリングされることができ、このとき、ブロックもよによってサブブロックの個数を2個または4個に調節でき、一つのサブブロックの大きさ (幅 \times 高さ) が16未満である場合、該当サブブロックへの分割を許容しない、またはISP コーディング自体を適用しないように制限できる。

30

【0252】

一方、ISP 予測モードの場合、一つのコーディングユニットが2個または4個のパーティションブロック、即ち、サブブロックに分割されて予測され、該当分割された2個または4個のパーティションブロックには同じ画面内の予測モードが適用される。

【0253】

前述したように、分割方向には水平方向 (横長さと縦長さが各々 M 、 N である $M \times N$ コーディングユニットが水平方向に分割されると、2個に分割される場合は $M \times (N / 2)$ ブロックに分割され、4個に分割される場合は $M \times (N / 4)$ ブロックに分割されると)、垂直方向 ($M \times N$ コーディングユニットが垂直方向に分割されると、2個に分割される場合は $(M / 2) \times N$ ブロックに分割され、4個に分割される場合は $(M / 4) \times N$ ブロックに分割されると)、が両方とも可能である。水平方向に分割される場合は、上側から下側方向の順序にパーティションブロックがコーディングされ、垂直方向に分割される場合は、左側から右側方向の順序にパーティションブロックがコーディングされる。現在コーディングされるパーティションブロックは、水平 (垂直) 方向分割である場合、上側 (左側) パーティションブロックの復元されたピクセル値を参照して予測されることができ、

40

【0254】

50

ISP 予測方法で生成されたレジデュアル信号にパーティションブロック単位で変換が適用されることができる。順方向 (forward) を基準にして1次変換 (core transform または primary transform) に既存 DCT - 2 だけでなく、DST - 7 / DCT - 8 組み合わせベースの MTS (Multiple Transform Selection) 技術が適用されることができ、1次変換によって生成された変換係数に順方向 LFNST (Low Frequency Non - Separable Transform) が適用されて最終的な修正された変換係数が生成されることができる。

【0255】

即ち、ISP 予測モードが適用されて分割されたパーティションブロックにも LFNST が適用されることができ、前述したように、分割されたパーティションブロックには同じイントラ予測モードが適用される。したがって、イントラ予測モードに基づいて導出される LFNST セット選択時に全てのパーティションブロックに導出された LFNST セットを適用することができる。即ち、全てのパーティションブロックに同じイントラ予測モードが適用されるため、これによって、全てのパーティションブロックには同じ LFNST セットが適用されることができる。

10

【0256】

一方、一例によって、LFNST は、横と縦の長さが全て4以上である変換ブロックに対してのみ適用されることができる。したがって、ISP 予測方式によって分割されたパーティションブロックの横または縦の長さが4未満である場合、LFNST が適用されずに LFNST インデックスもシグナリングされない。また、各パーティションブロックに LFNST を適用する場合、該当パーティションブロックを一つの変換ブロックと見なすことができる。もちろん、ISP 予測方式が適用されない場合、コーディングブロックに LFNST が適用されることができる。

20

【0257】

各パーティションブロックに LFNST を適用することを具体的にみると、下記の通りである。

【0258】

一例によって、個別的なパーティションブロックに対して順方向 LFNST を適用した後、左上端 4×4 領域に変換係数スキミング順序によって、最大16個 (8個または16個) の係数のみを残した後、残り位置及び領域は、全て0値で満たすゼロアウトが適用されることができる。

30

【0259】

または、一例によって、パーティションブロック一辺の長さが4である場合、左上端 4×4 領域に対してのみ LFNST を適用し、パーティションブロックの全ての辺、即ち、幅及び高さの長さが8以上である場合、左上端 8×8 領域内部の右下端 4×4 領域を除外した残り48個の係数に対して LFNST を適用することができる。

【0260】

または、一例によって、最悪の場合の計算複雑度を8乗算/サンプル (multiplications per sample) で合わせるために、各パーティションブロックが 4×4 または 8×8 である場合には順方向 LFNST 適用後に8個の変換係数のみを出力することができる。即ち、パーティションブロックが 4×4 である場合、変換マトリクスで 8×16 行列が適用され、パーティションブロックが 8×8 である場合、変換マトリクスで 8×48 行列が適用されることができる。

40

【0261】

一方、現在 VVC 標準で、LFNST インデックスシグナリングは、コーディングユニット単位で実行される。したがって、ISP 予測モードであり、全てのパーティションブロックに対して LFNST を適用する場合、該当パーティションブロックに対しては同じ LFNST インデックス値が適用されることができる。即ち、コーディングユニットレベルで LFNST インデックス値が一回送信されると、コーディングユニット内部の全ての

50

パーティションブロックに対しては該当 L F N S T インデックスが適用されることができ
る。前述したように、L F N S T インデックス値は 0、1、2 値を有することができ、0
は L F N S T が適用されない場合を示し、1 と 2 は L F N S T が適用される時に一つの L
F N S T セット内に存在する二つの変換マトリクスを指す。

【0262】

前記のように、L F N S T セットは、イントラ予測モードにより決定され、I S P 予測
モードである場合、コーディングユニット内の全てのパーティションブロックが同じイン
トラ予測モードで予測されるため、パーティションブロックは、同じ L F N S T セットを
参照することができる。

【0263】

他の一例として、L F N S T インデックスシグナリングは、依然としてコーディングユ
ニット単位で実行されるが、I S P 予測モードの場合に全てのパーティションブロックに
対して一律に L F N S T 適用可否を決定せずに、別途の条件を介して各々のパーティショ
ンブロックに対してコーディングユニットレベルでシグナリングされた L F N S T インデ
ックス値を適用するか、または L F N S T を適用しないかを決定することができる。ここ
で、別途の条件は、ビットストリームを介して各パーティションブロック別にフラグ形態
でシグナリングされることができ、フラグ値が 1 である場合は、コーディングユニットレ
ベルでシグナリングされた L F N S T インデックス値を適用し、フラグ値が 0 である場合
は、L F N S T を適用しない。

【0264】

以下では、I S P モードに L F N S T 適用時、最悪の場合に対する計算複雑度を維持す
る方法に対して記述する。

【0265】

I S P モードである場合、L F N S T 適用時、サンプル当たり（または、係数当たり、
位置当たり）乗算数を一定値以下に維持するために L F N S T 適用を制限することができ
る。パーティションブロックの大きさによって、下記のように L F N S T を適用してサン
プル当たり（または、係数当たり、位置当たり）乗算数を 8 個以下に維持できる。

【0266】

1. パーティションブロックの横長さと縦長さが全て 4 以上である場合は、現在 V V C
標準での L F N S T に対する最悪の場合に対する計算複雑度調節方式と同じ方式を適用す
ることができる。

【0267】

即ち、パーティションブロックが 4×4 ブロックである場合には 16×16 行列の代わり
に、順方向では 16×16 行列から上側 8 個の行をサンプリングした 8×16 行列を適
用し、逆方向では 16×16 行列から左側 8 個の列をサンプリングした 16×8 行列を適
用することができる。また、パーティションブロックが 8×8 ブロックである時は、順方
向の場合、 16×48 行列の代わりに、 16×48 行列から上側 8 個の行をサンプリング
した 8×48 行列を適用し、逆方向の場合、 48×16 行列の代わりに、 48×16 行列
から左側 8 個の列をサンプリングした 48×8 行列を適用することができる。

【0268】

$4 \times N$ または $N \times 4$ ($N \geq 4$) ブロックの場合、順方向変換を実行する時、左上端 4×4
ブロックに対してのみ 16×16 行列を適用した後に生成された 16 個の係数は、左上
端 4×4 領域に配置され、以外の領域は 0 値で満たされることができる。また、逆方向変
換を実行する時は、左上端 4×4 ブロックに位置した 16 個の係数をスキミング順序に
よって配置して入力ベクトルを構成した後に 16×16 行列を掛けて 16 個の出力デー
タを生成することができる。生成された出力データは、左上端 4×4 領域に配置され、左
上端 4×4 領域を除外した残り領域は 0 で満たされることができる。

【0269】

$8 \times N$ または $N \times 8$ ($N \geq 8$) ブロックの場合、順方向変換を実行する時、左上端 8×8
ブロック内部の R O I 領域（左上端 8×8 ブロックで右下端 4×4 ブロックを除外した

10

20

30

40

50

残り領域) に対してのみ 16×48 行列を適用した後に生成された 16 個の係数は、左上端 4×4 領域に配置され、以外領域は全て 0 値で満たされることができる。また、逆方向変換を実行する時は、左上端 4×4 ブロックに位置した 16 個の係数をスキニング順序によって配置して入力ベクトルを構成した後に 48×16 行列を掛けて 48 個の出力データを生成することができる。生成された出力データは、前記 ROI 領域に満たされ、残り領域は全て 0 値で満たされることができる。

【0270】

他の一例として、サンプル当たり(または、係数当たり、位置当たり)乗算数を一定値以下に維持するために、ISPパーティションブロックの大きさでないISPコーディングユニット大きさを基準にしてサンプル当たり(または、係数当たり、位置当たり)乗算数を8個以下に維持できる。もし、ISPパーティションブロックのうちLFNSTが適用される条件を満たすブロックが一つのみ存在する場合、パーティションブロックの大きさでない該当コーディングユニット大きさに基づいてLFNST最悪の場合に対する複雑度演算が適用されることができる。例えば、あるコーディングユニットに対するルマコーディングブロックが 4×4 大きさの4個のパーティションブロックに分割されてISPでコーディングされ、そのうち2個のパーティションブロックに対しては0でない変換係数が存在しない場合、他の2個のパーティションブロックには(エンコーダ基準に)各々8個でない16個の変換係数が生成されるように設定できる。

10

【0271】

以下では、ISPモードである場合、LFNSTインデックスをシグナリングする方法に対して考察する。

20

【0272】

前述のように、LFNSTインデックスは0、1、2値を有することができる、0はLFNSTを適用しないということを指示し、1と2は選択されたLFNSTセットに含まれている二つのLFNSTカーネルマトリクスのうちいずれか一つずつを指示する。LFNSTインデックスにより選択されたLFNSTカーネルマトリクスに基づいてLFNSTが適用される。現在VVC標準でLFNSTインデックスが送信される方法を説明すると、下記の通りである。

【0273】

1. コーディングユニット(CU)毎に一回ずつLFNSTインデックスを送信することができ、デュアルツリー(dual-tree)である時は、ルマブロックとクロマブロックに対して各々個別的なLFNSTインデックスがシグナリングされることができる。

30

【0274】

2. LFNSTインデックスがシグナリングされない場合、LFNSTインデックス値は、デフォルト値である0に定められる(infer)。LFNSTインデックス値が0に類推される場合は、下記の通りである。

【0275】

A. 変換が適用されないモードである場合(例えば、変換スキップ(transform skip)、BDPCM、無損失(lossless)コーディング等)

【0276】

B. 1次変換がDCT-2でない場合(DST7やDCT8)、即ち、水平方向の変換または垂直方向の変換がDCT-2でない場合

40

【0277】

C. コーディングユニットのルマブロックに対する横長さまたは縦長さが、変換が可能な最大ルマ変換の大きさを超過する場合、例えば、変換が可能な最大ルマ変換の大きさが64である場合、コーディングブロックのルマブロックに対する大きさが 128×16 のような場合にはLFNSTが適用されることができない。

【0278】

デュアルツリーの場合、ルマ成分に対するコーディングユニットとクロマ成分に対するコーディングユニットの各々に対して最大ルマ変換の大きさを超過するかどうか判断さ

50

れる。即ち、ルマブロックに対して変換が可能な最大ルマ変換の大きさを超過するかどうかチェックされ、クロマブロックに対してカラーフォーマットに対する対応ルマブロックの横／縦長さと最大変換が可能な最大ルマ変換の大きさを超過するかどうかチェックされる。例えば、カラーフォーマットが4：2：0である場合、対応ルマブロックの横／縦長さは、各々、該当クロマブロックの2倍になり、対応ルマブロックの変換大きさは、該当クロマブロックの2倍になる。他の例として、カラーフォーマットが4：4：4である場合、対応ルマブロックの横／縦長さと変換大きさは、対応するクロマブロックと同じである。

【0279】

64 - 長さ変換または32 - 長さ変換は、各々、64または32長さを有する横または縦に適用される変換を意味し、「変換大きさ」は該当長さである64または32を意味することができる。

10

【0280】

シングルツリーである場合、ルマブロックに対して横長さまたは縦長さが、変換が可能な最大ルマ変換ブロック大きさを超過するかどうかをチェックした後、超過する場合、LFNSTインデックスシグナリングを省略することができる。

【0281】

D . コーディングユニットの横長さと縦長さが全て4以上である場合にのみLFNSTインデックスを送信することができる。

【0282】

デュアルツリーである場合、該当成分（即ち、ルマまたはクロマ成分）に対する横長さと縦長さが全て4以上である場合に対してのみLFNSTインデックスをシグナリングすることができる。

20

【0283】

シングルツリーである場合は、ルマ成分に対する横長さと縦長さが全て4以上である場合に対してLFNSTインデックスをシグナリングすることができる。

【0284】

E . 最後の0でない係数の位置 (last non-zero coefficient position) がDC位置（ブロックの左上端位置）でない場合、デュアルツリータイプのルマブロックである場合は、最後の0でない係数の位置がDC位置でない時にLFNSTインデックスを送信する。デュアルツリータイプのクロマブロックである場合はCbに対する最後の0でない係数の位置とCrに対する最後の0でない係数の位置のうち一つでもDC位置でない時に該当LFNSTインデックスを送信する。

30

【0285】

シングルツリータイプの場合、ルマ成分、Cb成分、Cr成分のうち一つでも該当最後の0でない係数の位置がDC位置でない時にLFNSTインデックスを送信する。

【0286】

ここで、一つの変換ブロックに対する変換係数存在可否を指すCBF (coded block flag) 値が0である場合、LFNSTインデックスシグナリング可否を判断するために該当変換ブロックに対する最後の0でない係数の位置をチェックしない。即ち、該当CBF値が0である場合、該当ブロックに変換が適用されないため、LFNSTインデックスシグナリングに対する条件をチェックする時、最後の0でない係数の位置を考慮しない。

40

【0287】

例えば、1) デュアルツリータイプであり、ルマ成分である場合、該当CBF値が0である時は、LFNSTインデックスをシグナリングしなく、2) デュアルツリータイプであり、クロマ成分である場合、Cbに対するCBF値が0であり、Crに対するCBF値が1である時は、Crに対する最後の0でない係数の位置のみチェックして該当LFNSTインデックスを送信し、3) シングルツリータイプである場合は、ルマ、Cb、Crの全てに対して各CBF値が1である成分に対してのみ最後の0でない係数の位置をチェッ

50

クするようになる。

【0288】

F L F N S T 変換係数が存在できる位置でない位置に変換係数が存在することが確認された場合、L F N S T インデックスシグナリングを省略することができる。4 × 4 変換ブロックと 8 × 8 変換ブロックの場合は、V V C 標準での変換係数スキミング順序によって D C 位置から 8 個の位置に L F N S T 変換係数が存在でき、残り位置は全て 0 で満たされるようになる。また、4 × 4 変換ブロックと 8 × 8 変換ブロックでない場合は、V V C 標準での変換係数スキミング順序によって D C 位置から 16 個の位置に L F N S T 変換係数が存在でき、残り位置は全て 0 で満たされるようになる。

【0289】

したがって、レジデュアルコーディング (r e s i d u a l c o d i n g) を進行した後、前記 0 値が満たされるべき領域に 0 でない変換係数が存在すると、L F N S T インデックスシグナリングを省略することができる。

【0290】

一方、I S P モードは、ルマブロックである場合にのみ適用され、またはルマブロックとクロマブロックの両方ともに適用されることもできる。前述したように、I S P 予測が適用される場合、該当コーディングユニットは、2 個または 4 個のパーティションブロックに分割されて予測され、変換も該当パーティションブロックに各々適用されることができる。したがって、コーディングユニット単位で L F N S T インデックスをシグナリングする条件を決定する時も、該当パーティションブロックに各々 L F N S T が適用されることができるという事実を考慮しなければならない。また、I S P 予測モードが特定成分 (例えば、ルマブロック) に対してのみ適用される場合には、該当成分に対してのみパーティションブロックに分割されるという事実を考慮して L F N S T インデックスをシグナリングしなければならない。I S P モードである時に可能な L F N S T インデックスシグナリング方式を整理してみると、下記の通りである。

【0291】

1 . コーディングユニット (C U) 毎に一回ずつ L F N S T インデックスを送信ことができ、デュアルツリー (d u a l - t r e e) である時は、ルマブロックとクロマブロックに対して各々個別的な L F N S T インデックスがシグナリングされることができる。

【0292】

2 . L F N S T インデックスがシグナリングされない場合、L F N S T インデックス値は、デフォルト値である 0 に定められる (i n f e r) 。 L F N S T インデックス値が 0 に類推される場合は、下記の通りである。

【0293】

A . 変換が適用されないモードである場合 (例えば、変換スキップ (t r a n s f o r m s k i p) 、 B D P C M 、 無損失 (l o s s l e s s) コーディング等)

【0294】

B . コーディングユニットのルマブロックに対する横長さまたは縦長さが、変換が可能な最大ルマ変換の大きさを超過する場合、例えば、変換が可能な最大ルマ変換の大きさが 64 である場合、コーディングブロックのルマブロックに対する大きさが 128 × 16 のような場合には L F N S T が適用されることができない。

【0295】

コーディングユニットの代わりにパーティションブロックの大きさを基準にして L F N S T インデックスのシグナリング可否を決定することもできる。即ち、該当ルマブロックに対するパーティションブロックの横長さまたは縦長さが、変換が可能な最大ルマ変換の大きさを超過する場合、L F N S T インデックスシグナリングを省略して L F N S T インデックス値を 0 に類推できる。

【0296】

デュアルツリーの場合、ルマ成分に対するコーディングユニットまたはパーティションブロックとクロマ成分に対するコーディングユニットまたはパーティションブロックの各

10

20

30

40

50

々に対して最大変換ブロック大きさを超過するかどうか判断される。即ち、ルマに対するコーディングユニットまたはパーティションブロックの横と縦長さを各々最大ルマ変換大きさと比較して、一つでも最大ルマ変換大きさより大きい場合はLFNSTを適用しなく、クロマに対するコーディングユニットまたはパーティションブロックの場合にはカラーフォーマットに対する対応ルマブロックの横/縦長さと最大変換が可能な最大ルマ変換の大きさが比較される。例えば、カラーフォーマットが4:2:0である場合、対応ルマブロックの横/縦長さは、各々、該当クロマブロックの2倍になり、対応ルマブロックの変換大きさは、該当クロマブロックの2倍になる。他の例として、カラーフォーマットが4:4:4である場合、対応ルマブロックの横/縦長さと変換大きさは、対応するクロマブロックと同じである。

10

【0297】

シングルツリーである場合、ルマブロック(コーディングユニットまたはパーティションブロック)に対して横長さまたは縦長さが、変換が可能な最大ルマ変換ブロック大きさを超過するかどうかをチェックした後、超過する場合、LFNSTインデックスシグナリングを省略することができる。

【0298】

C. もし、現在のVVC標準に含まれているLFNSTを適用する場合、パーティションブロックの横長さと縦長さが全てが4以上である場合に対してのみLFNSTインデックスを送信することができる。

【0299】

もし、現在VVC標準に含まれているLFNST以外に $2 \times M (1 \times M)$ または $M \times 2 (M \times 1)$ ブロックに対するLFNSTまで適用する場合、パーティションブロックの大きさが $2 \times M (1 \times M)$ または $M \times 2 (M \times 1)$ ブロックより大きいまたは同じ場合に対してのみLFNSTインデックスを送信することができる。ここで、 $P \times Q$ ブロックが $R \times S$ ブロックより大きいまたは同じという意味は、 $P \geq R$ であり、 $Q \geq S$ ということの意味する。

20

【0300】

整理すると、パーティションブロックが、LFNSTが適用可能な最小限の大きさより大きいまたは同じ場合に対してのみLFNSTインデックスを送信することができる。デュアルツリーの場合、ルマまたはクロマ成分に対するパーティションブロックが、LFNSTが適用可能な最小限の大きさより大きいまたは同じ場合にのみLFNSTインデックスをシグナリングすることができる。シングルツリーの場合、ルマ成分に対するパーティションブロックが、LFNSTが適用可能な最小限の大きさより大きいまたは同じ場合にのみLFNSTインデックスをシグナリングすることができる。

30

【0301】

本文書において、 $M \times N$ ブロックが $K \times L$ ブロックより大きいまたは同じということは、 M が K より大きいまたは同じであり、 N が L より大きいまたは同じということの意味する。 $M \times N$ ブロックが $K \times L$ ブロックより大きいということは、 M が K より大きいまたは同じであり、 N が L より大きいまたは同じであり、かつ M が K より大きいまたは N が L より大きいということの意味する。 $M \times N$ ブロックが $K \times L$ ブロックより小さいまたは同じということは、 M が K より小さいまたは同じであり、 N が L より小さいまたは同じということの意味し、 $M \times N$ ブロックが $K \times L$ ブロックより小さいということは、 M が K より小さいまたは同じであり、 N が L より小さいまたは同じであり、かつ M が K より小さいまたは N が L より小さいということの意味する。

40

【0302】

D. 最後の0でない係数の位置(last non-zero coefficient position)がDC位置(ブロックの左上端位置)でない場合、デュアルツリータイプのルマブロックである場合は、全てのパーティションブロックのうち一つでも該当最後の0でない係数の位置がDC位置でない時にLFNSTインデックスを送信することができる。デュアルツリータイプでクロマブロックである場合は、Cbに対する全てのパー

50

パーティションブロックの (I S P モードがクロマ成分に適用されない場合にはパーティションブロックの数が一個と見なす) 最後の 0 でない係数の位置と、C r に対する全てのパーティションブロックの (I S P モードがクロマ成分に適用されない場合にはパーティションブロックの数が一個と見なす) 最後の 0 でない係数の位置と、のうち一つでも D C 位置でない時に該当 L N F S T インデックスを送信することができる。

【 0 3 0 3 】

シングルツリータイプ場合、ルマ成分、C b 成分、C r 成分に対する全てのパーティションブロックのうち一つでも最後の 0 でない係数の位置が D C 位置でない時に該当 L F N S T インデックスを送信することができる。

【 0 3 0 4 】

ここで、各パーティションブロックに対して変換係数存在可否を指す C B F (c o d e d b l o c k f l a g) 値が 0 である場合、L F N S T インデックスシグナリング可否を判断するために該当パーティションブロックに対する最後の 0 でない係数の位置をチェックしない。即ち、該当 C B F 値が 0 である場合、該当ブロックに変換が適用されないため、L F N S T インデックスシグナリングに対する条件をチェックする時、該当パーティションブロックに対する最後の 0 でない係数の位置を考慮しない。

【 0 3 0 5 】

例えば、1) デュアルツリータイプであり、ルマ成分である場合、各パーティションブロックに対して該当 C B F 値が 0 である時は、L F N S T インデックスシグナリング可否を決定する時、該当パーティションブロックを除外させ、2) デュアルツリータイプであり、クロマ成分である場合、各パーティションブロックに対して C b に対する C B F 値が 0 であり、C r に対する C B F 値が 1 である時は、C r に対する最後の 0 でない係数の位置のみをチェックして該当 L F N S T インデックスシグナリング可否を決定し、3) シングルツリータイプである場合、ルマ成分、C b 成分、C r 成分の全てのパーティションブロックに対して C B F 値が 1 であるブロックに対してのみ最後の 0 でない係数の位置をチェックして L F N S T インデックスシグナリング可否を決定することができる。

【 0 3 0 6 】

I S P モードである場合には最後の 0 でない係数の位置をチェックしないように映像情報を構成することもでき、これに対する実施例は、下記の通りである。

【 0 3 0 7 】

i . I S P モードである場合は、ルマブロックとクロマブロックの両方ともに対して最後の 0 でない係数の位置に対するチェックを省略して L F N S T インデックスシグナリングを許容することができる。即ち、全てのパーティションブロックに対して最後の 0 でない係数の位置が D C 位置であり、または該当 C B F 値が 0 であるとしても、該当 L F N S T インデックスシグナリングを許容することができる。

【 0 3 0 8 】

ii . I S P モードである場合は、ルマブロックに対してのみ最後の 0 でない係数の位置に対するチェックを省略し、クロマブロックの場合は、前述した方式の最後の 0 でない係数の位置に対するチェックを実行することができる。例えば、デュアルツリータイプであり、ルマブロックである場合は、最後の 0 でない係数の位置に対するチェックをしなくて L F N S T インデックスシグナリングを許容し、デュアルツリータイプであり、クロマブロックである場合は、前述した方式に最後の 0 でない係数の位置に対する D C 位置存在可否をチェックして該当 L F N S T インデックスのシグナリング可否を決定することができる。

【 0 3 0 9 】

iii . I S P モードであり、シングルツリータイプである場合は、前記 i 方式または ii 方式を適用することができる。即ち、I S P モードであり、シングルツリータイプに i 方式を適用する場合、ルマブロックとクロマブロックの両方ともに対して最後の 0 でない係数の位置に対するチェックを省略して L F N S T インデックスシグナリングを許容することができる。または、ii 方式を適用してルマ成分に対するパーティションブロックに対して

10

20

30

40

50

は最後の0でない係数の位置に対するチェックを省略し、クロマ成分に対するパーティションブロック（クロマ成分に対してISPを適用しない場合にはパーティションブロックの数が1と見なすことができる）に対しては、前述した方式に最後の0でない係数の位置に対するチェックを実行して該当LFNSTインデックスシグナリング可否を決定することができる。

【0310】

E. 全てのパーティションブロックのうち一つのパーティションブロックに対してでもLFNST変換係数が存在できる位置でない位置に変換係数が存在することが確認されると、LFNSTインデックスシグナリングを省略することができる。

【0311】

例えば、4×4パーティションブロックと8×8パーティションブロックの場合は、VVC標準での変換係数スキャン順序によって、DC位置から8個の位置にLFNST変換係数が存在でき、残り位置は全て0で満たされるようになる。また、4×4より大きいまたは同じであり、かつ4×4パーティションブロック及び8×8パーティションブロックでない場合には、VVC標準での変換係数スキャン順序によって、DC位置から16個の位置にLFNST変換係数が存在でき、残り位置は全て0で満たされるようになる。

【0312】

したがって、レジデュアルコーディング(residual coding)を進行した後前記0値が満たされるべき領域に0でない変換係数が存在すると、LFNSTインデックスシグナリングを省略することができる。

【0313】

一方、ISPモードである場合、現在VVC標準では水平方向と垂直方向に対して各々独立的に長さ条件を見てMTSインデックスに対するシグナリング無しでDC T - 2の代わりにD S T - 7を適用するようになる。横または縦の長さが4より大きいまたは同じであり、16より小さいまたは同じであるかが判断され、判断結果によって1次変換カーネルが決定される。したがって、ISPモードであり、かつLFNSTを適用することができる場合に対しては、下記のような変換組み合わせ構成が可能である。

【0314】

1. LFNSTインデックスが0である場合（LFNSTインデックスが0に類推される場合も含む）に対しては現在VVC標準に含まれているISPである時の1次変換決定条件に従うことができる。即ち、水平方向と垂直方向に対して各々独立的に長さ条件（4より大きいまたは同じであり、16より小さいまたは同じ条件）を満たすかどうかをチェックし、満たす場合は1次変換のためにDC T - 2の代わりにD S T - 7を適用し、満たさない場合はDC T - 2を適用することができる。

【0315】

2. LFNSTインデックスが0より大きい場合に対しては1次変換で下記のような二つの構成が可能である。

【0316】

A. 水平方向と垂直方向に対して全てDC T - 2を適用することができる。

【0317】

B. 現在VVC標準に含まれているISPである時の1次変換決定条件に従うことができる。即ち、水平方向と垂直方向に対して各々独立的に長さ条件（4より大きいまたは同じであり、16より小さいまたは同じ条件）を満たすかどうかをチェックし、満たす場合はDC T - 2の代わりにD S T - 7を適用し、満たさない場合はDC T - 2を適用することができる。

【0318】

ISPモードである時、LFNSTインデックスは、コーディングユニット毎に送信されるものではなく、パーティションブロック毎に送信されるように映像情報を構成することができる。このような場合、前述したLFNSTインデックスシグナリング方式でLF

10

20

30

40

50

N S Tインデックスが送信される単位内にパーティションブロックが1個のみが存在すると見なしてL F N S Tインデックスシグナリング可否を決定することができる。

【0319】

一方、以下ではL F N S TインデックスとM T Sインデックスのシグナリング順序に対して考察する。

【0320】

一例によって、レジデュアルコーディングでシグナリングされるL F N S Tインデックスは、最後の0でない係数位置に対するコーディング位置の次にコーディングされることができ、M T Sインデックスは、L F N S Tインデックスの直後にコーディングされることができ、このような構成の場合、変換ユニット毎にL F N S Tインデックスがシグナリングされることができ、または、レジデュアルコーディングでシグナリングされなくても、L F N S Tインデックスは、最後の有効係数位置に対するコーディングの次にコーディングされることができ、M T Sインデックスは、L F N S Tインデックスの次にコーディングされることができ、

10

【0321】

一例に係るレジデュアルコーディングのシンタックスは、下記の通りである。

【0322】

20

30

40

50

【表 4 - 1】

residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {	
if((sps_mts_enabled_flag && cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6) && cIdx == 0 && log2TbWidth > 4)	
log2ZoTbWidth = 4	
else	
log2ZoTbWidth = Min(log2TbWidth, 5)	
MaxCbs = 2 * (1 << log2TbWidth) * (1 << log2TbHeight)	
if((sps_mts_enabled_flag && cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6) && cIdx == 0 && log2TbHeight > 4)	10
log2ZoTbHeight = 4	
else	
log2ZoTbHeight = Min(log2TbHeight, 5)	
if(log2TbWidth > 0)	
last_sig_coeff_x_prefix	
if(log2TbHeight > 0)	
last_sig_coeff_y_prefix	
if(last_sig_coeff_x_prefix > 3)	20
last_sig_coeff_x_suffix	
if(last_sig_coeff_y_prefix > 3)	
last_sig_coeff_y_suffix	
remBinsPass1 = ((1 << (log2TbWidth + log2TbHeight)) * 7) >> 2	
log2SbW = (Min(log2TbWidth, log2TbHeight) < 2 ? 1 : 2)	
log2SbH = log2SbW	
if(log2TbWidth + log2TbHeight > 3) {	
if(log2TbWidth < 2) {	
log2SbW = log2TbWidth	
log2SbH = 4 - log2SbW	30
} else if(log2TbHeight < 2) {	
log2SbH = log2TbHeight	
log2SbW = 4 - log2SbH	
}	
}	
numSbCoeff = 1 << (log2SbW + log2SbH)	
lastScanPos = numSbCoeff	
lastSubBlock = (1 << (log2TbWidth + log2TbHeight - (log2SbW + log2SbH))) - 1	
do {	40

【 0 3 2 3 】

【表 4 - 2】

if(lastScanPos == 0) {
lastScanPos = numSbCoeff
lastSubBlock--
}
lastScanPos--
xS = DiagScanOrder[log2TbWidth - log2SbW][log2TbHeight - log2SbH]
[lastSubBlock][0]
yS = DiagScanOrder[log2TbWidth - log2SbW][log2TbHeight - log2SbH]
[lastSubBlock][1]
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][lastScanPos][0]
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][lastScanPos][1]
} while((xC != LastSignificantCoeffX) (yC != LastSignificantCoeffY))
cbWidth = CbWidth[0][x0][y0]
cbHeight = CbHeight[0][x0][y0]
if(Min(log2TbWidth, log2TbHeight) >= 2 && sps_lfnst_enabled_flag == 1
&&
CuPredMode[chType][x0][y0] == MODE_INTRA &&
IntraSubPartitionsSplit[x0][y0] == ISP_NO_SPLIT &&
(!intra_mip_flag[x0][y0] Min(log2TbWidth, log2TbHeight) >= 4) &&
Max(cbWidth, cbHeight) <= MaxTbSizeY &&
(cIdx == 0 (treeType == DUAL_TREE_CHROMA &&
(cIdx == 1 tu_cbf_cb[x0][y0] == 0)))) {
if(lastSubBlock == 0 && lastScanPos > 0 &&
!(lastScanPos > 7 && (log2TbWidth == 2 log2TbWidth == 3)
&& log2TbWidth == log2TbHeight)
lfnst_idx[x0][y0]
}
if(cIdx == 0 && lfnst_idx[x0][y0] == 0 &&
(log2TbWidth <= 5) && (log2TbHeight <= 5) &&
(LastSignificantCoeffX < 16) && (LastSignificantCoeffY < 16) &&
(IntraSubPartitionsSplit[x0][y0] == ISP_NO_SPLIT) && (!cu_sbt_flag)) {
if(((CuPredMode[chType][x0][y0] == MODE_INTER &&
sps_explicit_mts_inter_enabled_flag
(CuPredMode[chType][x0][y0] == MODE_INTRA &&
sps_explicit_mts_intra_enabled_flag))
tu_mts_idx[x0][y0]
}
if(tu_mts_idx[x0][y0] > 0 && cIdx == 0 && log2TbWidth > 4)
log2ZoTbWidth = 4
if(tu_mts_idx[x0][y0] > 0 && cIdx == 0 && log2TbHeight > 4)
log2ZoTbHeight = 4
log2TbWidth = log2ZoTbWidth
log2TbHeight = log2ZoTbHeight
...

10

20

30

40

【 0 3 2 4 】

表 4 に示す主要変数の意味は、下記の通りである。

【 0 3 2 5 】

1 . c b W i d t h 、 c b H e i g h t : 現在コーディングブロック (C o d i n g B l o c k) の幅と高さ

【 0 3 2 6 】

2 . l o g 2 T b W i d t h 、 l o g 2 T b H e i g h t : 現在変換ブロック (T r a n s f o r m B l o c k) の幅と高さに対するベース - 2 のログ値、ゼロアウトが反映されて 0 でない係数 (n o n - z e r o c o e f f i c i e n t) が存在できる左上端

50

領域に縮小されることができる。

【0327】

3. `sps_lfnst_enabled_flag`: LFNSTの適用可能 (`enable`) 可否を示すフラグであって、フラグ値が0である場合は、LFNSTが適用不可であることを示し、フラグ値が1である場合は、LFNSTが適用可能であることを示す。シーケンスパラメータセット (Sequence Parameter Set; SPS) に定義されている。

【0328】

4. `CuPredMode[chType][x0][y0]`: 変数 `chType` と (`x0`, `y0`) 位置に対応される予測モード、`chType` は0と1値を有することができ、0はルマ成分を示し、1はクロマ成分を示す。(`x0`, `y0`) 位置は、ピクチャ上での位置を示し、`CuPredMode[chType][x0][y0]` 値では `MODE_INTRA` (イントラ予測) と `MODE_INTER` (インター予測) が可能である。

10

【0329】

5. `IntraSubPartitionsSplit[x0][y0]`: (`x0`, `y0`) 位置に対する内容は、前記4と同じである。(`x0`, `y0`) 位置でのどのようなISP分割が適用されたかを示し、`ISP_NO_SPLIT` は、(`x0`, `y0`) 位置に該当するコーディングユニットがパーティションブロックに分割されないことを示す。

【0330】

6. `intra_mip_flag[x0][y0]`: (`x0`, `y0`) 位置に対する内容は、前記4と同じである。`intra_mip_flag` は、MIP (Matrix-based Intra Prediction) 予測モードが適用されたかどうかを示すフラグである。フラグ値が0である場合は、MIPが適用不可であることを示し、フラグ値が1である場合は、MIPが適用されることを示す。

20

【0331】

7. `ctx_idx`: 0値は、ルマを示し、1値と2値は、各々、クロマ成分であるCb、Crを示す。

【0332】

8. `treeType`: シングルツリー (`single-tree`) とデュアルツリー (`dual-tree`) などを指す (`SINGLE_TREE`: シングルツリー、`DUAL_TREE_LUMA`: ルマ成分に対するデュアルツリー、`DUAL_TREE_CHROMA`: クロマ成分に対するデュアルツリー)

30

【0333】

9. `tu_cbf_cbf[x0][y0]`: (`x0`, `y0`) 位置に対する内容は、前記4と同じである。Cb成分に対するCBF (`Coded Block Flag`) を示す、その値が0である場合は、0でない係数がCb成分に対する該当変換ユニットに存在しないということを意味し、1である場合は、0でない係数がCb成分に対する該当変換ユニットに存在するということを示す。

【0334】

10. `lastSubBlock`: 最後の有効係数 (`lastnon-zero coefficient`) が位置するサブブロック (`sub-block`, `Coefficient Group (CG)`) のスキャン順序上の位置を示す。0は、DC成分が含まれているサブブロックを指し、0より大きい場合は、DC成分が含まれているサブブロックでない。

40

【0335】

11. `lastScanPos`: 最後の有効係数がサブブロック内部でスキャン順序上どの位置にあるかを示す。一つのサブブロックが16個の位置で構成されている場合、0から15までの値が可能である。

【0336】

12. `lfnst_idx[x0][y0]`: パーシングしようとするLFNSTイン

50

デックスシンタックスエレメントである。パーシングされない場合、0値に類推される。即ち、デフォルト値が0に設定され、LFNSTを適用しないことを示す。

【0337】

13. LastSignificantCoeffX、LastSignificantCoeffY：最後の有効係数が変換ブロック内に位置するx座標とy座標を示す。x座標は、0から始めて左側から右側へ増加し、y座標は0から始めて上側から下側に増加する。二つの変数の値が全て0である場合は、最後の有効係数がDCに位置することを意味する。

【0338】

14. cu_sbt_flag：現在VVC標準に含まれているサブブロック変換(SubBlock Transform、SBT)が適用可能かどうかを示すフラグであって、フラグ値が0である場合は、SBTが適用不可であることを示し、フラグ値が1である場合は、SBTが適用されることを示す。

10

【0339】

15. sps_explicit_mts_inter_enabled_flag、sps_explicit_mts_intra_enabled_flag：各々インターCUとイントラCUに対して明示的なMTSが適用されたかどうかを示すフラグであって、該当フラグ値が0である場合は、インターCUまたはイントラCUに対してMTSが適用不可であることを示し、1である場合は、適用可能であることを示す。

【0340】

16. tu_mts_idx[x0][y0]：パーシングしようとするMTSインデックスシンタックスエレメントである。パーシングされない場合、0値に類推される。即ち、デフォルト値が0に設定され、水平方向と垂直方向に対して全てDC-2が適用されることを示す。

20

【0341】

表4のように、シングルツリーである場合は、ルマに対する最後の有効係数位置条件のみを有してLFNSTインデックスのシグナリング可否を決定することができる。即ち、最後の有効係数位置がDCでない、かつ最後の有効係数が左上端サブブロック(CG)、例えば、4x4ブロック、内部に存在する場合、LFNSTインデックスがシグナリングされる。このとき、4x4変換ブロックと8x8変換ブロックの場合は、左上端サブブロック内部の0から7までの位置に最後の有効係数が存在してこそLFNSTインデックスがシグナリングされる。

30

【0342】

デュアルツリーの場合は、ルマとクロマは、各々、独立的にLFNSTインデックスがシグナリングされ、クロマの場合は、Cb成分に対してのみ最後の有効係数位置条件を適用してLFNSTインデックスをシグナリングすることができる。Cr成分に対しては該当条件をチェックしなく、もし、Cbに対するCBF値が0である場合は、Cr成分に対して最後の有効係数位置条件を適用してLFNSTインデックスをシグナリングすることができる。

【0343】

表4の「 $\text{Min}(\log_2 \text{TbWidth}, \log_2 \text{TbHeight}) = 2$ 」は、「 $\text{Min}(\text{tbWidth}, \text{tbHeight}) = 4$ 」で表現されることができ、「 $\text{Min}(\log_2 \text{TbWidth}, \log_2 \text{TbHeight}) = 4$ 」は、「 $\text{Min}(\text{tbWidth}, \text{tbHeight}) = 16$ 」で表現されることができ、

40

【0344】

表4において、 $\log_2 \text{ZoTbWidth}$ と $\log_2 \text{ZoTbHeight}$ は、各々、ゼロアウトにより最後の有効係数が存在できる左上端領域に対する幅と高さのベースが2である($\text{base} - 2$)ログ値を意味する。

【0345】

表4のように、 $\log_2 \text{ZoTbWidth}$ と $\log_2 \text{ZoTbHeight}$ 値は、二

50

箇所アップデートされることができる。1番目は、M T S インデックスまたは L F N S T インデックス値がパーシングされる前であり、2番目は、M T S インデックスのパーシング後である。

【0346】

1番目のアップデートは、M T S インデックス (t u _ m t s _ i d x [x 0] [y 0]) 値がパーシングされる前であるため、M T S インデックス値にかかわらず l o g 2 Z o T b W i d t h と l o g 2 Z o T b H e i g h t を設定することができる。

【0347】

M T S インデックスがパーシングされた後にはM T S インデックス値が0より大きい場合 (D S T - 7 / D C T - 8 組み合わせである場合) に対して l o g 2 Z o T b W i d t h と l o g 2 Z o T b H e i g h t を設定するようになる。1次変換で水平方向と垂直方向に対して各々独立的にD S T - 7 / D C T - 8を適用する場合、各方向に対して行または列毎に最大16個までの有効係数が存在できる。即ち、32長さ以上のD S T - 7 / D C T - 8を適用した後、左側または上側から行または列毎に最大16個の変換係数が導出されることができる。したがって、2次元ブロックに対しては水平方向と垂直方向の両方ともに対してD S T - 7 / D C T - 8が適用される時、最大左上端16 x 16領域までのみ有効係数が存在できる。

10

【0348】

また、現在1次変換で水平方向と垂直方向に対して各々独立的にD C T - 2が適用される場合、各方向に対して行または列毎に最大32個までの有効係数が存在できる。即ち、64長さ以上のD C T - 2を適用する時は、左側または上側から行または列毎に最大32個の変換係数が導出されることができる。したがって、2次元ブロックに対しては水平方向と垂直方向の両方ともに対してD C T - 2が適用される時、最大左上端32 x 32領域までのみ有効係数が存在できる。

20

【0349】

また、水平方向と垂直方向に対して、一方ではD S T - 7 / D C T - 8が適用され、他方ではD C T - 2が適用される時、前者の方向では16個の有効係数が存在でき、後者の方向では32個の有効係数が存在できる。例えば、64 x 8変換ブロックであり、かつ水平方向にはD C T - 2が適用され、垂直方向にはD S T - 7が適用される場合 (暗黙的M T S が適用される状況で発生できる) 、最大左上端32 x 8領域で有効係数が存在できる。

30

【0350】

もし、表4のように l o g 2 Z o T b W i d t h と l o g 2 Z o T b H e i g h t が二箇所アップデートされる場合、即ち、M T S インデックスパーシング前にアップデートされる場合、以下の表のように l a s t _ s i g _ c o e f f _ x _ p r e f i x と l a s t _ s i g _ c o e f f _ y _ p r e f i x の範囲が l o g 2 Z o T b W i d t h と l o g 2 Z o T b H e i g h t により決められることができる。

【0351】

【表5】

<p>7.4.9.11 Residual coding semantics</p> <p>.....</p> <p>last_sig_coeff_x_prefix specifies the prefix of the column position of the last significant coefficient in scanning order within a transform block. The values of last_sig_coeff_x_prefix shall be in the range of 0 to (log2ZoTbWidth << 1) - 1, inclusive.</p> <p>When last_sig_coeff_x_prefix is not present, it is inferred to be 0.</p> <p>last_sig_coeff_y_prefix specifies the prefix of the row position of the last significant coefficient in scanning order within a transform block. The values of last_sig_coeff_y_prefix shall be in the range of 0 to (log2ZoTbHeight << 1) - 1, inclusive.</p> <p>When last_sig_coeff_y_prefix is not present, it is inferred to be 0.</p> <p>.....</p>	40
--	----

50

【0352】

また、このような場合、last_sig_coeff_x_prefixとlast_sig_coeff_y_prefixに対する二進化過程でlog2ZoTbWidthとlog2ZoTbHeight値を反映してlast_sig_coeff_x_prefixとlast_sig_coeff_y_prefixの最大値を設定することができる。

【0353】

【表6】

Table 9-77 – Syntax elements and associated binarizations

residual_coding()	last_sig_coeff_x_prefix	TR	$cMax = (\log2ZoTbWidth \ll 1) - 1, cRiceParam = 0$
	last_sig_coeff_y_prefix	TR	$cMax = (\log2ZoTbHeight \ll 1) - 1, cRiceParam = 0$
	last_sig_coeff_x_suffix	FL	$cMax = (1 \ll ((last_sig_coeff_x_prefix \gg 1) - 1)) - 1$
	last_sig_coeff_y_suffix	FL	$cMax = (1 \ll ((last_sig_coeff_y_prefix \gg 1) - 1)) - 1$

10

【0354】

一方、一例によって、ISPモードであり、かつLFNSTが適用される場合、表4のシグナリングを適用した時、表7のようにスペックテキストが構成されることができる。表4と比較した時、ISPモードでない場合に対してのみLFNSTインデックスをシグナリングした条件(表4のIntraSubPartitionsSplit[x0][y0] == ISP_NO_SPLIT)が削除された。

20

【0355】

シングルツリーである場合、ルマである時(cIdx = 0である時)に送信したLFNSTインデックスをクロマである時に再使用する場合は、有効係数が存在する1番目のISPパーティションブロックに対して送信されたLFNSTインデックスをクロマ変換ブロックに適用できる。または、シングルツリーである場合であるとしても、クロマ成分である場合に対してルマ成分と別途にLFNSTインデックスをシグナリングすることができる。表7に記載された変数に対する説明は、表4の通りである。

30

【0356】

40

50

【表 7】

residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {
...
if(log2TbWidth > 0)
last_sig_coeff_x_prefix
if(log2TbHeight > 0)
last_sig_coeff_y_prefix
if(last_sig_coeff_x_prefix > 3)
last_sig_coeff_x_suffix
if(last_sig_coeff_y_prefix > 3)
last_sig_coeff_y_suffix
...
cbWidth = CbWidth[0][x0][y0]
cbHeight = CbHeight[0][x0][y0]
if(Min(log2TbWidth, log2TbHeight) >= 2 && sps_lfnst_enabled_flag == 1 && && CuPredMode[chType][x0][y0] == MODE_INTRA && (!intra_mip_flag[x0][y0] Min(log2TbWidth, log2TbHeight) >= 4) && Max(cbWidth, cbHeight) <= MaxTbSizeY && (cIdx == 0 (treeType == DUAL_TREE_CHROMA && (cIdx == 1 tu_cbf_cb[x0][y0] == 0)))) {
if(lastSubBlock == 0 && lastScanPos > 0 && !(lastScanPos > 7 && (log2TbWidth == 2 log2TbWidth == 3) && log2TbWidth == log2TbHeight)
lfnst_idx[x0][y0]
}
if(cIdx == 0 && lfnst_idx[x0][y0] == 0 && (log2TbWidth <= 5) && (log2TbHeight <= 5) && (LastSignificantCoeffX < 16) && (LastSignificantCoeffY < 16) && (IntraSubPartitionsSplit[x0][y0] == ISP_NO_SPLIT) && (!cu_sbt_flag)) {
if(((CuPredMode[chType][x0][y0] == MODE_INTER && sps_explicit_mts_inter_enabled_flag) (CuPredMode[chType][x0][y0] == MODE_INTRA && sps_explicit_mts_intra_enabled_flag)))
tu_mts_idx[x0][y0]
}
...

10

20

30

40

【0357】

一方、一例によって、LFNSTインデックスまたは/及びMTSインデックスは、コーディングユニットレベルでシグナリングされることができる。LFNSTインデックスは、前述したように、0、1、2の三つの値を有することができる。0はLFNSTを適用しないことを指し、1と2は選択されたLFNSTセットに含まれている2個のLFNSTカーネル候補のうち各々1番目の候補と2番目の候補を指示する。LFNSTインデックスは、トランケイテッドユナリ二進化(truncated unary binarization)を介してコーディングされ、0、1、2値は、各々、binストリング0、10、11でコーディングされることができる。

【0358】

50

一例によると、1次変換で水平方向と垂直方向の両方ともに対してDCT-2が適用される時にのみLFNSTが適用されることができる。したがって、もし、MTSインデックスをLFNSTインデックスシグナリング後にシグナリングする場合、LFNSTインデックス値が0である場合に限ってのみMTSインデックスをシグナリングすることができ、LFNSTインデックスが0でない場合にはMTSインデックスをシグナリングせずに水平方向と垂直方向の両方ともにDCT-2を適用して1次変換を実行することができる。

【0359】

MTSインデックス値は、0、1、2、3、4の値を有することができ、0、1、2、3、4は、各々、水平方向と垂直方向に対してDCT-2/DCT-2、DST-7/DST-7、DCT-8/DST-7、DST-7/DCT-8、DCT-8/DCT-8が適用されることを指示することができる。また、MTSインデックスは、トランケイテッドユナリ二進化を介してコーディングされることができ、前記0、1、2、3、4値は、各々、binストリング0、10、110、1110、1111でコーディングされることができる。

10

【0360】

LFNSTインデックス及びMTSインデックスは、コーディングユニットレベルでシグナリングされることができ、MTSインデックスをコーディングユニットレベルでLFNSTインデックス以後に続いてコーディングできる。これに対するコーディングユニットシンタックステーブルは、以下の通りである。

20

【0361】

30

40

50

【表 8】

coding_unit(x0, y0, cbWidth, cbHeight, cqtDepth, treeType, modeType) {	Descriptor
.....	ae(v)
if(cu_cbf) {	
.....	
LfnstDcOnly = 1	
LfnstZeroOutSigCoeffFlag = 1	
transform_tree(x0, y0, cbWidth, cbHeight, treeType)	
lfnstWidth = (treeType == DUAL_TREE_CHROMA) ? cbWidth / SubWidthC :(IntraSubPartitionsSplitType == ISP_VER_SPLIT) ? cbWidth / NumIntraSubPartitions : cbWidth	10
lfnstHeight = (treeType == DUAL_TREE_CHROMA) ? cbHeight / SubHeightC :(IntraSubPartitionsSplitType == ISP_HOR_SPLIT) ? cbHeight / NumIntraSubPartitions : cbHeight	
if(Min(lfnstWidth, lfnstHeight) >= 4 && sps_lfnst_enabled_flag == 1 && CuPredMode[chType][x0][y0] == MODE_INTRA && (!intra_mip_flag[x0][y0] Min(lfnstWidth, lfnstHeight) >= 16) && !transform_skip_flag[x0][y0] && Max(cbWidth, cbHeight) <= MaxTbSizeY) {	
if((IntraSubPartitionsSplitType != ISP_NO_SPLIT LfnstDcOnly == 0) && LfnstZeroOutSigCoeffFlag == 1)	
lfnst_idx[x0][y0]	ae(v)
}	20
if(tu_cbf_luma[x0][y0] && treeType != DUAL_TREE_CHROMA && lfnst_idx[x0][y0] == 0 && (cbWidth <= 32) && (cbHeight <= 32) && (IntraSubPartitionsSplit[x0][y0] == ISP_NO_SPLIT) && (!cu_sbt_flag)) {	
if(((CuPredMode[chType][x0][y0] == MODE_INTER && sps_explicit_mts_inter_enabled_flag) (CuPredMode[chType][x0][y0] == MODE_INTRA && sps_explicit_mts_intra_enabled_flag)))	
tu_mts_idx[x0][y0]	ae(v)
}	
}	

10

20

30

【0362】

表 8 の変数 LfnstDcOnly と変数 LfnstZeroOutSigCoeffFlag は、以下の表 9 に示すように設定されることができる。

【0363】

変数 LfnstDcOnly は、該当 CBF (Coded Block Flag、該当ブロック内に有効係数が一つでも存在する場合は 1、そうでない場合は 0) 値が 1 である変換ブロックに対して最後の有効係数が全て DC 位置 (左上端位置) に位置する場合は 1 になり、そうでない場合は 0 になる。より具体的に、デュアルツリーマである場合には、最後の有効係数の位置をルマ変換ブロック一つに対してチェックし、デュアルツリークマである場合には、Cb に対する変換ブロックと Cr に対する変換ブロックの両方ともに対して最後の有効係数位置をチェックする。シングルツリーである場合には、ルマ、Cb、Cr に対する変換ブロックに対して最後の有効係数位置をチェックすることができる。

40

【0364】

変数 LfnstZeroOutSigCoeffFlag は、LFNST が適用される時、ゼロアウトになる位置に有効係数が存在する場合は 0 であり、そうでない場合は 1 になる。

【0365】

表 8 及び以下の表に含まれる lfnst_idx[x0][y0] は、該当コーディングユニットに対する LFNST インデックスを示し、tu_mts_idx[x0][y0] は、該当コーディングユニットに対する MTS インデックスを示す。

50

【0366】

表8に示すように、`lfnst_idx[x0][y0]`をシグナリングする条件に`transform_skip_flag[x0][y0]`値が0であるかをチェックする条件(`!transform_skip_flag[x0][y0]`)が含まれることができ、この場合、既存の`tu_mts_idx[x0][y0]`値が0であるかをチェックする条件(即ち、水平方向と垂直方向に対して両方ともDCT-2であるかどうかをチェックすること)は省略されることができる。

【0367】

`transform_skip_flag[x0][y0]`は、コーディングユニットが変換が省略される変換スキップモードにコーディングされたかどうかを示し、前記フラグは、MTSインデックス及びLFNSTインデックスより先にシグナリングされる。即ち、`tu_mts_idx[x0][y0]`値をシグナリングする前に`lfnst_idx[x0][y0]`をシグナリングするため、`transform_skip_flag[x0][y0]`値に対する条件のみをチェックすることができる。

10

【0368】

表8に示すように、`tu_mts_idx[x0][y0]`をコーディングする時、様々な条件がチェックされ、前述したように、`lfnst_idx[x0][y0]`値が0である場合にのみ`tu_mts_idx[x0][y0]`がシグナリングされる。

【0369】

また、`tu_cbf_luma[x0][y0]`は、ルマ成分に対して有効係数が存在するかどうかを示すフラグであり、`cbWidth`と`cbHeight`は、各々、ルマ成分に対するコーディングユニットの幅と高さを示す。

20

【0370】

表8によると、ルマ成分に対するコーディングユニットの幅と高さが両方とも32以下である時、`tu_mts_idx[x0][y0]`がシグナリングされ、即ち、MTS適用可否がルマ成分に対するコーディングユニットの幅と高さにより決定される。

【0371】

他の例によって、変換ブロックタイリング(`TU_tiling`)が発生する場合(例えば、最大変換大きさが32に設定された場合、 64×64 コーディングユニットは、4個の 32×32 変換ブロックに分割されてコーディングされる)、各変換ブロックの大きさを基準にしてMTSインデックスがシグナリングされることができる。例えば、変換ブロックの幅と高さが両方とも32以下である時、コーディングユニット内の全ての変換ブロックに対しては同じMTSインデックス値が適用されて同じ1次変換が適用されることができる。また、変換ブロックタイリングが発生する場合、表8の`tu_cbf_luma[x0][y0]`値は、左上端変換ブロックに対するCBF値であり、または全ての変換ブロックに対して一つの変換ブロックでも該当CBF値が1である場合は、1に設定されることができる。

30

【0372】

また、表8によると、ISPモードである場合にも(`IntraSubPartitionsSplitType != ISP_NO_SPLIT`)`lfnst_idx[x0][y0]`をシグナリングするように構成でき、全てのISPパーティションブロックに対して同じLFNSTインデックス値が適用されることができる。

40

【0373】

一方、`tu_mts_idx[x0][y0]`は、ISPモードでない場合にのみシグナリングされることができる(`IntraSubPartitionsSplit[x0][y0] == ISP_NO_SPLIT`)。

【0374】

表8のように、MTSインデックスをLFNSTインデックスの直後にシグナリングする場合、レジデュアルコーディングを実行する時、1次変換に対する情報を知ることができない。即ち、MTSインデックスがレジデュアルコーディング以後にシグナリングされ

50

る。したがって、レジデュアルコーディングパートで32長さのDST-7またはDCT-8に対して16個の係数のみを残してゼロアウトを実行する部分は、以下の表9のように変更されることができる。

【0375】

【表9-1】

residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {	Descriptor
if((sps_mts_enabled_flag && cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6) && cIdx == 0 && log2TbWidth > 4)	
log2ZoTbWidth = 4	
else	
log2ZoTbWidth = Min(log2TbWidth, 5)	
MaxCbs = 2 * (1 << log2TbWidth) * (1 << log2TbHeight)	
if((sps_mts_enabled_flag && cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6) && cIdx == 0 && log2TbHeight > 4)	
log2ZoTbHeight = 4	
else	
log2ZoTbHeight = Min(log2TbHeight, 5)	

10

20

【0376】

30

40

50

【表 9 - 2】

if(log2TbWidth > 0)	
last_sig_coeff_x_prefix	ac(v)
if(log2TbHeight > 0)	
last_sig_coeff_y_prefix	ac(v)
if(last_sig_coeff_x_prefix > 3)	
last_sig_coeff_x_suffix	ac(v)
if(last_sig_coeff_y_prefix > 3)	
last_sig_coeff_y_suffix	ac(v)
log2TbWidth = log2ZoTbWidth	
log2TbHeight = log2ZoTbHeight	
remBinsPass1 = ((1 << (log2TbWidth + log2TbHeight)) * 7) >> 2	
log2SbW = (Min(log2TbWidth, log2TbHeight) < 2 ? 1 : 2)	
log2SbH = log2SbW	
if(log2TbWidth + log2TbHeight > 3) {	
if(log2TbWidth < 2) {	
log2SbW = log2TbWidth	
log2SbH = 4 - log2SbW	
} else if(log2TbHeight < 2) {	
log2SbH = log2TbHeight	
log2SbW = 4 - log2SbH	
}	
}	
numSbCoeff = 1 << (log2SbW + log2SbH)	
lastScanPos = numSbCoeff	
lastSubBlock = (1 << (log2TbWidth + log2TbHeight - (log2SbW + log2SbH))) - 1	
do {	
if(lastScanPos == 0) {	
lastScanPos = numSbCoeff	
lastSubBlock--	
}	
lastScanPos--	
xS = DiagScanOrder[log2TbWidth - log2SbW][log2TbHeight - log2SbH]	
[lastSubBlock][0]	
yS = DiagScanOrder[log2TbWidth - log2SbW][log2TbHeight - log2SbH]	
[lastSubBlock][1]	
xC = (xS << log2SbW) + DiagScanOrder[log2SbW][log2SbH][lastScanPos][0]	
yC = (yS << log2SbH) + DiagScanOrder[log2SbW][log2SbH][lastScanPos][1]	
} while((xC != LastSignificantCoeffX) (yC != LastSignificantCoeffY))	
if(lastSubBlock == 0 && log2TbWidth >= 2 && log2TbHeight >= 2	
&&	
!transform_skip_flag[x0][y0] && lastScanPos > 0)	
LfstDcOnly = 0	
if((lastSubBlock > 0 && log2TbWidth >= 2 && log2TbHeight >= 2)	
(lastScanPos > 7 && (log2TbWidth == 2 log2TbWidth == 3) &&	
log2TbWidth == log2TbHeight)	
LfstZeroOutSigCoeffFlag = 0	

10

20

30

40

【 0 3 7 7 】

表 9 のように $\log_2 Z o T b W i d t h$ と $\log_2 Z o T b H e i g h t$ を決定する過程で (ここで、 $\log_2 Z o T b W i d t h$ と $\log_2 Z o T b H e i g h t$ は、各々、ゼロアウトが実行された後に残った左上端領域に対する幅と高さのベース - 2 (base - 2) ログ値を示す) $t u_m t s_i d x [x 0] [y 0]$ 値をチェックする部分が省略されることができる。

【 0 3 7 8 】

表 9 の $l a s t_s i g_c o e f f_x_p r e f i x$ と $l a s t_s i g_c o e f f_y_p r e f i x$ に対する二進化は、表 6 のように $\log_2 Z o T b W i d t h$ と

50

log2ZotbHeightに基づいて決定されることができる。

【0379】

また、表9のように、レジデュアルコーディングでlog2ZotbWidthとlog2ZotbHeightを決定する時、sps_mts_enable_flagをチェックする条件が追加されることができる。

【0380】

一例によって、ルマ変換ブロックに対する最後の有効係数位置に対する情報をレジデュアルコーディング過程で記録しておくこと、表10のようにMTSインデックスをシグナリングすることもできる。

【0381】

【表10】

10

coding_unit(x0, y0, cbWidth, cbHeight, cqtDepth, treeType, modeType) {	Descriptor
.....	ae(v)
if(cu_cbf) {	
.....	
LfnstDcOnly = 1	
LfnstZeroOutSigCoeffFlag = 1	
LumaLastSignificantCoeffX = 0	
LumaLastSignificantCoeffY = 0	
transform_tree(x0, y0, cbWidth, cbHeight, treeType)	
lfnstWidth = (treeType == DUAL_TREE_CHROMA) ? cbWidth / SubWidthC :(IntraSubPartitionsSplitType == ISP_VER_SPLIT) ? cbWidth / NumIntraSubPartitions : cbWidth	
lfnstHeight = (treeType == DUAL_TREE_CHROMA) ? cbHeight / SubHeightC :(IntraSubPartitionsSplitType == ISP_HOR_SPLIT) ? cbHeight / NumIntraSubPartitions : cbHeight	
if(Min(lfnstWidth, lfnstHeight) >= 4 && sps_lfnst_enabled_flag == 1 && CuPredMode[chType][x0][y0] == MODE_INTRA && (!intra_mip_flag[x0][y0] Min(lfnstWidth, lfnstHeight) >= 16) && !transform_skip_flag[x0][y0] && Max(cbWidth, cbHeight) <= MaxTbSizeY) {	
if((IntraSubPartitionsSplitType != ISP_NO_SPLIT LfnstDcOnly == 0) && LfnstZeroOutSigCoeffFlag == 1)	
lfnst_idx[x0][y0]	ae(v)
}	
if(tu_cbf_luma[x0][y0] && treeType != DUAL_TREE_CHROMA && lfnst_idx[x0][y0] == 0 && (cbWidth <= 32) && (cbHeight <= 32) && (LumaLastSignificantCoeffX < 16) && (LumaLastSignificantCoeffY < 16) && (IntraSubPartitionsSplit[x0][y0] == ISP_NO_SPLIT) && (!cu_sbt_flag)) {	
if(((CuPredMode[chType][x0][y0] == MODE_INTER && sps_explicit_mts_inter_enabled_flag) (CuPredMode[chType][x0][y0] == MODE_INTRA && sps_explicit_mts_intra_enabled_flag)))	
tu_mts_idx[x0][y0]	ae(v)
}	
}	

20

30

40

【0382】

表10において、LumaLastSignificantCoeffXとLumaLastSignificantCoeffYは、各々、ルマ変換ブロックに対する最後の有効係数位置のX座標とY座標を示す。LumaLastSignificantCoeffXとLumaLastSignificantCoeffYが全て16より小さくなるべき条件が表10に追加されたし、もし、二つのうち一つでも16以上になると、水平方向と垂直方向の両方ともにDC T - 2が適用されることであるため、tu_mts_idx[x0][y0]に対するシグナリングを省略し、水平方向と垂直方向に対して全てDC T - 2が適用されることと類推できる。

50

【0383】

LumaLastSignificantCoeffXとLumaLastSignificantCoeffYが全て16より小さいということは、最後の有効係数が左上端 16×16 領域内に存在するということを意味し、現在VVC標準で32長さのDST-7またはDCT-8が適用される場合、最左側または最上側から16個の変換係数のみを残すゼロアウトが適用された可能性が存在することを示す。したがって、`tu__mts__idx[x0][y0]`をシグナリングして1次変換のために使われた変換カーネルを指示することができる。

【0384】

一方、他の例によって、コーディングユニットシンタックステーブル、変換ユニットシンタックステーブル、及びレジデュアルコーディングシンタックステーブルは、以下の表の通りである。表11によると、MTSインデックスは、変換ユニットレベルでコーディングユニットレベルのシンタックスに移動し、LFNSTインデックスシグナリング以後にシグナリングされる。また、コーディングユニットにISPが適用される場合、LFNSTを許容しない制限条件が除去された。コーディングユニットにISPが適用される場合、LFNSTを許容しない制限条件が除去されるため、LFNSTを全てのイントラ予測ブロックに適用できる。また、MTSインデックス及びLFNSTインデックスは、全てコーディングユニットレベルの最後の部分に条件付きでシグナリングされる。

【0385】

10

20

30

40

50

【表 1 1】

coding_unit(x0, y0, cbWidth, cbHeight, cqtDepth, treeType, modeType) {	
...	
LfnstDcOnly = 1	
LfnstZeroOutSigCoeffFlag = 1	
MtsZeroOutSigCoeffFlag = 1	
transform_tree(x0, y0, cbWidth, cbHeight, treeType)	
LfnstWidth = (treeType == DUAL_TREE_CHROMA) ? cbWidth / SubWidthC : (IntraSubPartitionsSplitType == ISP_VER_SPLIT) ? cbWidth / NumIntraSubPartitions : cbWidth	10
LfnstHeight = (treeType == DUAL_TREE_CHROMA) ? cbHeight / SubHeightC : (IntraSubPartitionsSplitType == ISP_HOR_SPLIT) ? cbHeight / NumIntraSubPartitions : cbHeight	
if(Min(LfnstWidth, LfnstHeight) >= 4 && sps_lfnst_enabled_flag == 1 && CuPredMode[chType][x0][y0] == MODE_INTRA && (!intra_mip_flag[x0][y0] Min(LfnstWidth, LfnstHeight) >= 16) && Max(cbWidth, cbHeight) <= MaxTbSizeY) {	
if((IntraSubPartitionsSplitType != ISP_NO_SPLIT LfnstDcOnly == 0) && LfnstZeroOutSigCoeffFlag == 1)	20
lfnst_idx [x0][y0]	
}	
if(treeType != DUAL_TREE_CHROMA && lfnst_idx[x0][y0] == 0 && transform_skip_flag[x0][y0] == 0 && Max(cbWidth, cbHeight) <= 32 && IntraSubPartitionsSplit[x0][y0] == ISP_NO_SPLIT && (!cu_sbt_flag) && MtsZeroOutSigCoeffFlag == 1 && tu_cbf_luma[x0][y0]) {	
if(((CuPredMode[chType][x0][y0] == MODE_INTER && sps_explicit_mts_inter_enabled_flag) (CuPredMode[chType][x0][y0] == MODE_INTRA && sps_explicit_mts_intra_enabled_flag)))	
mts_idx [x0][y0]	30
}	
...	

【 0 3 8 6 】

【表 1 2】

transform_unit(x0, y0, tbWidth, tbHeight, treeType, subTuIndex, chType) {	
...	
if(tu_cbf_luma[x0][y0] && treeType != DUAL_TREE_CHROMA && (tbWidth <= 32) && (tbHeight <= 32) && (IntraSubPartitionsSplit[x0][y0] == ISP_NO_SPLIT) && (!cu_sbt_flag)) {	40
if(sps_transform_skip_enabled_flag && !BdpcmFlag[x0][y0] && tbWidth <= MaxTsSize && tbHeight <= MaxTsSize)	
transform_skip_flag[x0][y0]	
}	
...	

【 0 3 8 7 】

【表 1 3】

residual_coding(x0, y0, log2TbWidth, log2TbHeight, cIdx) {	
...	
if((sps_mts_enabled_flag && cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6)) && cIdx == 0 && log2TbWidth > 4)	
log2ZoTbWidth = 4	
else	
log2ZoTbWidth = Min(log2TbWidth, 5)	10
MaxCcbcs = 2 * (1 << log2TbWidth) * (1 << log2TbHeight)	
if((sps_mts_enabled_flag && cu_sbt_flag && log2TbWidth < 6 && log2TbHeight < 6)) && cIdx == 0 && log2TbHeight > 4)	
log2ZoTbHeight = 4	
else	
log2ZoTbHeight = Min(log2TbHeight, 5)	
...	
if((lastSubBlock > 0 && log2TbWidth >= 2 && log2TbHeight >= 2) (lastScanPos > 7 && (log2TbWidth == 2 log2TbWidth == 3) && log2TbWidth == log2TbHeight))	20
LfstZeroOutSigCoeffFlag = 0	
if((LastSignificantCoeffX > 15 LastSignificantCoeffY > 15) && cIdx == 0)	
MtsZeroOutSigCoeffFlag = 0	
...	

【0388】

表 1 1 において、MtsZeroOutSigCoeffFlag は、最初に 1 に設定され、この値は、表 1 3 のレジデュアルコーディングで変更されることができる。変数 MtsZeroOutSigCoeffFlag は、ゼロアウトによって 0 で満たされるべき領域 (LastSignificantCoeffX > 15 || LastSignificantCoeffY > 15) に有効係数が存在すると、その値が 1 から 0 に変更され、この場合、表 1 1 のように、MTS インデックスはシグナリングされない。

【0389】

一方、表 1 1 のように、tu_cbf_luma [x 0] [y 0] が 0 である場合には mts_idx [x 0] [y 0] コーディングを省略することができる。即ち、ルマ成分の CBF 値が 0 である場合は、変換を適用しないため、MTS インデックスをシグナリングする必要がなくて、MTS インデックスコーディングを省略することができる。

【0390】

一例によって、前記技術的特徴は、他の条件付き構文で具現されることができる。例えば、MTS が実行された後、現在ブロックの DC 領域を除外した領域に有効係数が存在するかどうかを示す変数を導出することができ、前記変数が DC 領域を除外した領域に有効係数が存在することを示す場合、MTS インデックスをシグナリングすることができる。即ち、現在ブロックの DC 領域を除外した領域に有効係数が存在するということは、tu_cbf_luma [x 0] [y 0] 値が 1 であることを示し、この場合、MTS インデックスをシグナリングすることができる。

【0391】

前記変数は、MtsDcOnly で表すことができ、変数 MtsDcOnly は、コー

10

20

30

40

50

ディングユニットレベルで最初に1に設定された後、レジデュアルコーディングレベルで現在ブロックのDC領域を除外した領域に有効係数が存在することを示す場合、その値が0に変更されることができる。変数 `MtsDcOnly` が0である場合、MTSインデックスがシグナリングされるように映像情報が構成されることができる。

【0392】

もし、`tu_cbf_luma[x0][y0]` が0である場合は、表12の変換ユニットレベルでレジデュアルコーディングシンタックスの呼び出しが行われないため、変数 `MtsDcOnly` は、初期値1を維持するようになる。このような場合、変数 `MtsDcOnly` が0に変更されなかったため、MTSインデックスがシグナリングされないように映像情報が構成されることができる。即ちMTSインデックスは、パーシング及びシグナリングされない。

10

【0393】

一方、デコーディング装置は、表13の変数 `MtsZeroOutSigCoeffFlag` を導出するために変換係数のカラーインデックス (`cIdx`) を判断することができる。カラーインデックス (`cIdx`) が0であることは、ルマ成分を意味する。

【0394】

一例によって、現在ブロックのルマ成分にのみMTSが適用されることができるため、デコーディング装置は、MTSインデックスのパーシング可否を決定する変数 `MtsZeroOutSigCoeffFlag` を導出する時、カラーインデックスがルマであるかどうかを判断することができる (`if cIdx == 0, MtsZeroOutSigCoeffFlag = 0`)。

20

【0395】

変数 `MtsZeroOutSigCoeffFlag` は、MTS適用時にゼロアウトが実行されたかどうかを示す変数であって、MTS実行後にゼロアウトにより最後の有効係数が存在できる左上端領域、即ち、左上端 16×16 領域以外の領域に変換係数が存在するかどうかを示す。変数 `MtsZeroOutSigCoeffFlag` は、表11のようにコーディングユニットレベルで最初に1に設定され (`MtsZeroOutSigCoeffFlag = 1`)、 16×16 領域以外の領域に変換係数が存在する場合、表13のようにレジデュアルコーディングレベルでその値が1から0へ変更されることができる (`MtsZeroOutSigCoeffFlag = 0`)。変数 `MtsZeroOutSigCoeffFlag` の値が0である場合、MTSインデックスはシグナリングされない。

30

【0396】

表13のように、レジデュアルコーディングレベルで、MTSに伴われるゼロアウトが実行されたかどうかによって0でない変換係数が存在できるノンゼロアウト領域が設定されることができ、この場合にも、カラーインデックス (`cIdx`) が0である場合、ノンゼロアウト領域は、現在ブロックの左上端 16×16 領域に設定されることができる。

【0397】

このように、MTSインデックスのパーシング可否を決定する変数を導出する時は、カラー成分がルマであるかまたはクロマであるかを判断するが、現在ブロックのルマ成分またはクロマ成分の両方ともにLFNSTが適用されることができるため、LFNSTインデックスのパーシング可否を決定する変数を導出する時はカラー成分を判断しない。

40

【0398】

例えば、表11にはLFNST適用時にゼロアウトが実行されたことを示すことができる変数 `LfnstZeroOutSigCoeffFlag` が示されている。変数 `LfnstZeroOutSigCoeffFlag` は、現在ブロックの左上端第1の領域を除外した第2の領域に有効係数が存在するかどうかを示し、この値は、最初に1に設定され、第2の領域に有効係数が存在すると、その値は、0に変更されることができる。最初設定された変数 `LfnstZeroOutSigCoeffFlag` 値が1に維持されてこそLFNSTインデックスがパーシングされることができる。変数 `LfnstZeroO`

50

utSigCoeffFlag値が1であるかどうかを判断及び導出する時、現在ブロックのルマ成分またはクロマ成分の両方ともにLFNSTが適用されることができると、現在ブロックのカラーインデックスは判断されない。

【0399】

図15は、一実施例に係るCABACエンコーディングシステムのブロック図を示すであり、単一構文要素を符号化するためのCABAC(context-adaptive binary arithmetic coding)のブロック図を示す。

【0400】

まず、CABACの符号化過程は、入力信号が二進値でない構文要素である場合、二進化を介して入力信号を二進値に変換する。入力信号が既に二進値である場合には二進化を経ずにバイパス、即ち、符号化エンジンに入力される。ここで、二進値を構成する各々の二進数0または1をbinという。例えば、二進化された後の二進ストリングが110である場合、1、1、0の各々を一つのbinという。一つの構文要素に対する前記bin(ら)は、該当構文要素の値を示すことができる。

【0401】

二進化されたbinは、正規(regular)符号化エンジンまたはバイパス符号化エンジンに入力される。

【0402】

正規符号化エンジンは、該当binに対して確率値を反映する文脈モデルを割り当て、割り当てられた文脈モデルに基づいて該当binを符号化する。正規符号化エンジンでは各binに対する符号化を実行した後に該当binに対する確率モデルを更新することができる。このように符号化されるbinを文脈符号化bin(context-coded bin)という。

【0403】

バイパス符号化エンジンは、入力されたbinに対して確率を推定する手順と符号化後に該当binに適用した確率モデルを更新する手順を省略する。文脈を割り当てる代わりに均一な確率分布を適用して入力されるbinを符号化することによって符号化速度を向上させる。このように符号化されるbinをバイパスbin(bypass bin)という。

【0404】

エントロピー符号化は、正規符号化エンジンを介して符号化を実行するか、バイパス符号化エンジンを介して符号化を実行するかを決定し、符号化経路をスイッチングすることができる。エントロピー復号化は、符号化と同じ過程を逆順に実行する。

【0405】

LFNSTインデックスに対する二進化方法でトランケイテッドユナリコード(Truncated unary code)が適用される場合、LFNSTインデックスは、最大2個のbinで構成され、可能なLFNSTインデックス値である0、1、2に対する二進化コード(binary code)では各々0、10、11が割り当てられる。

【0406】

一例によって、LFNSTインデックスの1番目のbinに対してはコンテキストベースのCABACコーディングが適用され(regular coding)、2番目のbinに対してはバイパスコーディング(bypass coding)が適用されることができると、

【0407】

また、他の例によると、LFNSTインデックスの1番目及び2番目のbinの両方ともに対してコンテキストベースのCABACコーディングが適用されることができると、このようなコンテキストコーディングされたbinによるLFNSTインデックスのctxIncの割当てを表で表せると、以下の通りである。

【0408】

10

20

30

40

50

【表 1 4】

Syntax element	binIdx					
	0	1	2	3	4	>= 5
.....
lfnst_idx[][]	(treeType != SINGLE_TREE) ? 1 : 0	2	na	na	na	na
.....

【0409】

10

表 1 4 のように、1 番目の bin (binIdx = 0) に対しては、シングルツリーの場合、0 番コンテキストを適用し、シングルツリーでない場合、1 番コンテキストを適用することができる。また、表 1 4 のように、2 番目の bin (binIdx = 1) に対しては、2 番コンテキストを適用することができる。即ち、1 番目の bin に対しては 2 個のコンテキストが割り当てられることができ、2 番目の bin に対しては 1 個のコンテキストが割り当てることができる。即ち、1 番目の bin に対しては 2 個のコンテキストが割り当てられることができ、2 番目の bin に対しては 1 個のコンテキストが割り当てることができる。即ち、表 1 4 のように 1 番コンテキストを割り当てることができる。

【0410】

ここで、シングルツリーは、ルマ成分とクロマ成分が同じコーディング構造 (coding structure) でコーディングされることを意味する。コーディングユニットが同じコーディング構造で分割された後、コーディングユニットの大きさが特定閾値以下になってルマ成分とクロマ成分が別途のツリー構造でコーディングされる場合、該当コーディングユニットに対してはデュアルツリーと見なして 1 番目の bin のコンテキストを決定することができる。即ち、表 1 4 のように 1 番コンテキストを割り当てることができる。

20

【0411】

または、1 番目の bin に対して変数 treeType の値が SINGLE_TREE で割り当てられる場合は 0 番コンテキストを、そうでない場合は 1 番コンテキストを使用してコーディングできる。

【0412】

30

一例によって、エンコーディング過程に 2 個の LFNST カーネル候補のうち 1 番目 (または、2 番目) の候補のみ試みる時、LFNST インデックスは、該当候補に対して固定された 2 個の bin 値を有するようになる (1 番目の候補のみ試みる場合、10 でコーディングされ、2 番目の候補のみ試みる場合、11 でコーディングされる) 。この場合、2 番目の bin をバイパスでコーディングすると、2 番目の bin が固定された値 (0 または 1) であるにもかかわらず、2 番目の bin に対して固定されたビット量が発生するようになって 2 番目の bin に対してコーディング費用が相当大きくなる。2 番目の bin をバイパスでないコンテキスト方式にコーディングすると、LFNST カーネル候補で固定された 1 個のみ試みる場合、2 番目の bin に対して該当固定値 (0 または 1) が発生する確率が 100% である側にアップデートされるため、コーディング費用が相当減少されることができる。総合すると、LFNST インデックスコーディングのための二つの bin を全てコンテキスト方式にコーディングすることで、エンコーディング過程で LFNST カーネル候補を一つのみ固定して適用しても、二つの候補を全て適用すること対比コーディング効率の損失を最小化しながらエンコーディング複雑度も減らすことができるため、エンコーディング過程で性能 - 複雑度トレードオフ (trade-off) を探すことにあたって実質的な自由度を付与することができる。

40

【0413】

以下の図面は、本明細書の具体的な一例を説明するために作成された。図面に記載された具体的な装置の名称や具体的な信号 / メッセージ / フィールドの名称は、例示的に提示されたものであるため、本明細書の技術的特徴が以下の図面に使われた具体的な名称に制

50

限されない。

【0414】

図16は、本文書の一実施例に係るビデオデコーディング装置の動作を示す流れ図である。

【0415】

図16に開示された各ステップは、図5乃至図15に詳述された内容のうち一部に基づいている。したがって、図3、図5乃至図15に詳述された内容と重複する具体的な内容は、説明を省略し、または簡単にする。

【0416】

一実施例に係るデコーディング装置300は、ビットストリームからイントラ予測モードに対する情報、レジデュアル情報、及びLFNSTインデックスを受信することができる(S1610)。

10

【0417】

このような情報はシンタックス情報で受信され、シンタックス情報は0と1を含む二進化されたbinストリングで受信される。

【0418】

デコーディング装置は、LFNSTインデックスのシンタックス要素に対する二進化情報を導出することができる。これは受信された変換インデックスのシンタックス要素が有することができる二進化値に対する候補セットを生成することであって、本実施例による場合、LFNSTインデックスのシンタックス要素は、トランケイテッドユナリコード方式に二進化されることができる。

20

【0419】

本実施例に係る変換インデックスのシンタックス要素は、LFNSTが適用されるかどうか及びLFNST変換セットに含まれているLFNSTカーネルのうちいずれか一つを指示することができる、LFNST変換セットが二つのLFNSTカーネルを含む場合、LFNSTインデックスのシンタックス要素の値は三つである。

【0420】

即ち、一実施例によって、LFNSTインデックスに対するシンタックス要素値は、現在ブロックにLFNSTが適用されない場合を指示する0、LFNSTカーネルのうち1番目のLFNSTカーネルを指示する1、LFNSTカーネルのうち2番目のLFNSTカーネルを指示する2、を含むことができる。

30

【0421】

この場合、3個のLFNSTインデックスに対するシンタックス要素値は、トランケイテッドユナリコード方式によって0、10、11でコーディングされることができる。即ち、シンタックス要素に対する値0は「0」で、シンタックス要素に対する値1は「10」で、シンタックス要素に対する値2は「11」で、二進化されることができる。

【0422】

また、デコーディング装置は、受信されたレジデュアル情報に基づいてレジデュアルコーディングを実行して変換係数を導出することができる(S1620)。

【0423】

デコーディング装置300は、ビットストリームから現在ブロックに対する量子化された変換係数に関する情報をデコーディングすることができ、現在ブロックに対する量子化された変換係数に関する情報に基づいて対象ブロックに対する量子化された変換係数を導出することができる。対象ブロックに対する量子化された変換係数に関する情報は、SPS(Sequence Parameter Set)またはスライスヘッダ(slice header)に含まれることができ、簡素化変換(RST)が適用されるかどうかに対する情報、簡素化ファクタに関する情報、簡素化変換を適用する最小変換サイズに対する情報、簡素化変換を適用する最大変換サイズに対する情報、簡素化逆変換サイズ、変換セットに含まれている変換カーネルマトリックスのうちいずれか一つを指示する変換インデックスに対する情報のうち少なくとも一つを含むことができる。

40

50

【0424】

デコーディング装置300は、現在ブロックに対するレジデュアル情報、即ち、量子化された変換係数に対して逆量子化を実行して変換係数を導出することができ、導出された変換係数を所定スキミング順序に配列できる。

【0425】

具体的に、導出された変換係数は、 4×4 ブロック単位で逆方向対角スキャン順序によって配列されることができ、 4×4 ブロック内の変換係数も逆方向対角スキャン順序によって配列されることができる。即ち、逆量子化が実行された変換係数は、VVCやHEVCのビデオコーデックで適用されている逆方向スキャン順序によって配置されることができる。

10

【0426】

このようなレジデュアル情報に基づいて導出された変換係数は、前記のように逆量子化された変換係数であってもよく、量子化された変換係数であってもよい。即ち、変換係数は、量子化可否にかかわらず、現在ブロックで0でないデータであるかどうかをチェックすることができるデータであればよい。

【0427】

デコーディング装置は、量子化された変換係数に逆変換を適用してレジデュアルサンプルを導出することができる。

【0428】

前述したように、デコーディング装置は、非分離変換であるLFNSTまたは分離変換であるMTSを適用してレジデュアルサンプルを導出することができ、このような変換は、各々、LFNSTカーネル、即ち、LFNSTマトリクスを指示するLFNSTインデックスとMTSカーネルを指示するMTSインデックスとに基づいて実行されることができる。

20

【0429】

一方、デコーディング装置は、現在ブロックのツリータイプによって、LFNSTインデックスの1番目のbinに対するコンテキスト情報を導出し、2番目に対する既設定されたコンテキスト情報を導出することができる(S1630)。

【0430】

現在ブロックのツリータイプによって、1番目のbinに対するコンテキスト情報は、異なる値として導出されることができる。例えば、現在ブロックのツリータイプがシングルツリーである場合、1番目のbinは、第1のコンテキスト情報として導出され、現在ブロックのツリータイプがシングルツリーでない場合、1番目のbinは、第2のコンテキスト情報として導出されることができる。

30

【0431】

一方、2番目のbinに対するコンテキスト情報は、現在ブロックのツリータイプにかかわらず既設定された一つの値として導出されることができる。即ち、一例によると、LFNSTインデックスのbinストリングは、二つのbinが全てバイパス方式でないコンテキストモデルに基づいてデコーディングされることができる。

【0432】

デコーディング装置は、コンテキスト情報に基づいてシンタックス要素binストリングのbinをデコーディングし、コンテキスト情報ベースのデコーディングによりLFNSTインデックスのシンタックス要素が有することができる二進化値のうち現在ブロックに適用されるLFNSTインデックスに対するシンタックス要素の値を導出することができる(S1640)。

40

【0433】

整理すると、デコーディング装置は、トランケイテッドユナリコード方式に二進化されたbinストリングを受信し、コンテキスト情報に基づいてLFNSTインデックスのシンタックス要素をデコーディングする。

【0434】

50

即ち、L F N S T インデックス 0、1、2、のうちいずれかが一つが現在対象ブロックに適用されるかが導出されることができる。

【0435】

デコーディング装置 300 は、現在ブロックに適用されるイントラ予測モードによるマッピング関係に基づいて L F N S T 変換セットを決定し、L F N S T 変換セット、L F N S T 変換セットに含まれている L F N S T カーネル、及び L F N S T インデックスに基づいて L F N S T を適用して修正された変換係数を導出することができる (S 1 6 5 0)。

【0436】

その後、デコーディング装置は、修正された変換係数に対する逆 1 次変換に基づいて現在ブロックに対するレジデュアルサンプルを導出し (S 1 6 6 0)、レジデュアルサンプルに基づいて復元ピクチャを生成することができる (S 1 6 7 0)。

10

【0437】

以下の図面は、本明細書の具体的な一例を説明するために作成された。図面に記載された具体的な装置の名称や具体的な信号 / メッセージ / フィールドの名称は、例示的に提示されたものであるため、本明細書の技術的特徴が以下の図面に使われた具体的な名称に制限されない。

【0438】

図 17 は、本文書の一実施例に係るビデオエンコーディング装置の動作を示す流れ図である。

【0439】

図 17 に開示された各ステップは、図 5 乃至図 15 に詳述された内容のうち一部に基づいている。したがって、図 2 及び図 5 乃至図 15 に詳述された内容と重複する具体的な内容は、説明を省略し、または簡単にする。

20

【0440】

一実施例に係るエンコーディング装置 200 は、現在ブロックに適用されるイントラ予測モードに基づいて現在ブロックに対する予測サンプルを導出することができる (S 1 7 1 0)。

【0441】

エンコーディング装置 200 は、予測サンプルに基づいて現在ブロックに対するレジデュアルサンプルを導出することができる (S 1 7 2 0)。

30

【0442】

エンコーディング装置 200 は、レジデュアルサンプルに L F N S T または M T S のうち少なくとも一つを適用して現在ブロックに対する変換係数を導出し、変換係数を所定のスキミング順序によって配列できる。

【0443】

エンコーディング装置は、レジデュアルサンプルに対する 1 次変換に基づいて現在ブロックに対する変換係数を導出することができる (S 1 7 3 0)。

【0444】

1 次変換は、M T S のように複数の変換カーネルを介して実行されることができ、この場合、イントラ予測モードに基づいて変換カーネルが選択されることができる。

40

【0445】

エンコーディング装置は、M T S を適用して変換係数を導出した後、現在ブロックの左上端特定領域、例えば、16 × 16 領域を除外した現在ブロックの残り領域をゼロアウトすることができる。

【0446】

また、エンコーディング装置 200 は、現在ブロックに対する変換係数に対して 2 次変換、または非分離変換、具体的に、L F N S T を実行するかどうかを決定し、変換係数に L F N S T を適用して修正された変換係数を導出することができる。具体的に、エンコーディング装置は、L F N S T 変換セット、L F N S T 変換セットに含まれている L F N S T カーネルに基づいて L F N S T を適用して修正された変換係数を導出することができる

50

(S 1 7 4 0)。

【 0 4 4 7 】

L F N S T は、変換対象になる係数を垂直または水平方向に分離して変換する 1 次変換と違って、係数を特定方向に分離せずに変換を適用する非分離変換である。このような非分離変換は、変換対象になる対象ブロック全体でない低周波領域にのみ変換を適用する低周波非分離変換である。

【 0 4 4 8 】

エンコーディング装置は、L F N S T カーネルを指示する L F N S T インデックスまたは M T S カーネルを指示する M T S インデックスのうち少なくとも一つをエンコーディングすることができる。

10

【 0 4 4 9 】

まず、エンコーディング装置は、L F N S T インデックスに対するシンタックス要素の値を導出し、L F N S T インデックスに対するシンタックス要素値を二進化することができる (S 1 7 5 0)。

【 0 4 5 0 】

本実施例に係る L F N S T インデックスのシンタックス要素は、L F N S T が適用されるかどうか及び L F N S T 変換セットに含まれている L F N S T カーネルのうちいずれかが一つを指示することができ、L F N S T 変換セットが二つの L F N S T カーネルを含む場合、L F N S T インデックスのシンタックス要素の値は三つである。

【 0 4 5 1 】

20

一実施例によって、L F N S T インデックスに対するシンタックス要素値は、現在ブロックに L F N S T が適用されない場合を指示する 0、L F N S T カーネルのうち 1 番目の L F N S T カーネルを指示する 1、L F N S T カーネルのうち 2 番目の L F N S T カーネルを指示する 2、として導出されることができる。

【 0 4 5 2 】

エンコーディング装置は、3 個の変換インデックスに対するシンタックス要素値をトランケイテッドユナリコード方式によって、0、1 0、1 1 で二進化できる。即ち、シンタックス要素に対する値 0 は「0」で、シンタックス要素に対する値 1 は「1 0」で、シンタックス要素に対する値 2 は「1 1」で、二進化されることができ、エンコーディング装置は、導出された変換インデックスに対するシンタックス要素に対して「0」、「1 0」、及び「1 1」のうちいずれかが一つで二進化できる。

30

【 0 4 5 3 】

エンコーディング装置は、現在ブロックのツリータイプによって L F N S T インデックスの 1 番目の b i n に対するコンテキスト情報を導出し、2 番目に対する既設定されたコンテキスト情報を導出し、コンテキスト情報に基づいてシンタックス要素 b i n スtring の b i n をエンコーディングすることができる (S 1 7 6 0)。

【 0 4 5 4 】

現在ブロックのツリータイプによって、1 番目の b i n に対するコンテキスト情報は、異なる値として導出されることができる。例えば、現在ブロックのツリータイプがシングルツリーである場合、1 番目の b i n は、第 1 のコンテキスト情報として導出され、現在ブロックのツリータイプがシングルツリーでない場合、1 番目の b i n は、第 2 のコンテキスト情報として導出されることができる。

40

【 0 4 5 5 】

一方、2 番目の b i n に対するコンテキスト情報は、現在ブロックのツリータイプにかかわらず一つの値として導出されることができる。即ち、一例によると、L F N S T インデックスの b i n スtring は、二つの b i n が全てバイパス方式でないコンテキストモデルに基づいてエンコーディングされることができる。

【 0 4 5 6 】

整理すると、エンコーディング装置は、トランケイテッドユナリコード方式に L F N S T インデックスの b i n スtring を二進化し、互いに異なるコンテキスト情報に基づい

50

て該当二進化値に対する L F N S T インデックスのシンタックス要素をエンコーディングする。

【 0 4 5 7 】

エンコーディング装置は、コーディングユニットレベルで L F N S T インデックス及び M T S インデックスのうち少なくとも一つがシグナリングされ、 L F N S T インデックスのシグナリング直後に M T S インデックスがシグナリングされるように、映像情報を構成及び出力することができ、量子化されたレジデュアル情報エンコーディング後に出力できる (S 1 7 7 0)。

【 0 4 5 8 】

また、エンコーディング装置は、現在ブロックに対する変換係数または修正された変換係数に基づいて量子化を実行して量子化された変換係数を導出し、量子化された変換係数に関する情報を含む映像情報をエンコーディング及び出力することができる。

10

【 0 4 5 9 】

エンコーディング装置は、量子化された変換係数に対する情報を含むレジデュアル情報を生成することができる。レジデュアル情報は、詳述した変換関連情報 / シンタックス要素を含むことができる。エンコーディング装置は、レジデュアル情報を含む映像 / ビデオ情報をエンコーディングしてビットストリーム形態で出力できる。

【 0 4 6 0 】

より具体的に、エンコーディング装置は、量子化された変換係数に関する情報を生成して、生成された量子化された変換係数に関する情報をエンコーディングすることができる。

20

【 0 4 6 1 】

本文書において、量子化 / 逆量子化及び / または変換 / 逆変換のうち少なくとも一つは省略されることができる。前記量子化 / 逆量子化が省略される場合、前記量子化された変換係数は、変換係数と呼ばれることができる。前記変換 / 逆変換が省略される場合、前記変換係数は、係数またはレジデュアル係数と呼ばれることもでき、または表現の統一性のために、依然として変換係数と呼ばれることもできる。

【 0 4 6 2 】

また、本文書において、量子化された変換係数及び変換係数は、各々、変換係数及びスケーリングされた (s c a l e d) 変換係数と呼ばれることができる。この場合、レジデュアル情報は、変換係数 (ら) に関する情報を含むことができ、前記変換係数 (ら) に関する情報は、レジデュアルコーディングシンタックスを介してシグナリングされることができる。前記レジデュアル情報 (または、前記変換係数 (ら) に関する情報) に基づいて変換係数が導出されることができ、前記変換係数に対する逆変換 (スケーリング) を介してスケーリングされた変換係数が導出されることができる。前記スケーリングされた変換係数に対する逆変換 (変換) に基づいてレジデュアルサンプルが導出されることができる。これは本文書の他の部分でも同様に適用 / 表現されることができる。

30

【 0 4 6 3 】

前述した実施例において、方法は、一連のステップまたはブロックで流れ図に基づいて説明されているが、本文書は、ステップの順序に限定されるものではなく、あるステップは、前述と異なるステップと、異なる順序にまたは同時に発生できる。また、当業者であれば、流れ図に示すステップが排他的でなく、他のステップが含まれ、または、流れ図の一つまたはそれ以上のステップが本文書の範囲に影響を及ぼさずに削除可能であることを理解することができる。

40

【 0 4 6 4 】

前述した本文書による方法は、ソフトウェア形態で具現されることができ、本文書によるエンコーディング装置及び / またはデコーディング装置は、例えば、 T V 、コンピュータ、スマートフォン、セットトップボックス、ディスプレイ装置などの映像処理を実行する装置に含まれることができる。

【 0 4 6 5 】

本文書において、実施例がソフトウェアで具現される時、前述した方法は、前述した機

50

能を遂行するモジュール（過程、機能など）で具現されることができる。モジュールは、メモリに格納され、プロセッサにより実行されることができる。メモリは、プロセッサの内部または外部にあり、よく知られた多様な手段でプロセッサと連結されることができる。プロセッサは、ASIC (application-specific integrated circuit)、他のチップセット、論理回路及び/またはデータ処理装置を含むことができる。メモリは、ROM (read-only memory)、RAM (random access memory)、フラッシュメモリ、メモリカード、格納媒体及び/または他の格納装置を含むことができる。即ち、本文書で説明した実施例は、プロセッサ、マイクロプロセッサ、コントローラまたはチップ上で具現されて実行されることができる。例えば、各図面で示す機能ユニットは、コンピュータ、プロセッサ、マイクロプロセッサ、コントローラまたはチップ上で具現されて実行されることができる。

10

【0466】

また、本文書が適用されるデコーディング装置及びエンコーディング装置は、マルチメディア放送受信装置、モバイル通信端末、ホームシネマビデオ装置、デジタルシネマビデオ装置、監視用カメラ、ビデオ対話装置、ビデオ通信のようリアルタイム通信装置、モバイルストリーミング装置、格納媒体、カムコーダ、注文型ビデオ (VOD) サービス提供装置、OTTビデオ (Over the top video) 装置、インターネットストリーミングサービス提供装置、3次元 (3D) ビデオ装置、画像電話ビデオ装置、及び医療用ビデオ装置などに含まれることができ、ビデオ信号またはデータ信号を処理するために使われることができる。例えば、OTTビデオ (Over the top video) 装置として、ゲームコンソール、ブルーレイプレーヤ、インターネット接続TV、ホームシアターシステム、スマートフォン、タブレットPC、DVR (Digital Video Recorder) などを含むことができる。

20

【0467】

また、本文書が適用される処理方法は、コンピュータで実行されるプログラムの形態で生産されることができ、コンピュータが読み取り可能な記録媒体に格納されることができる。また、本文書によるデータ構造を有するマルチメディアデータもコンピュータが読み取り可能な記録媒体に格納されることができる。前記コンピュータが読み取り可能な記録媒体は、コンピュータで読み出すことができるデータが格納される全ての種類の格納装置及び分散格納装置を含む。前記コンピュータが読み取り可能な記録媒体は、例えば、ブルーレイディスク (BD)、汎用直列バス (USB)、ROM、PROM、EPROM、EEPROM、RAM、CD-ROM、磁気テープ、フロッピーディスク、及び光学的データ格納装置を含むことができる。また、前記コンピュータが読み取り可能な記録媒体は、搬送波 (例えば、インターネットを介した送信) の形態で具現されたメディアを含む。また、エンコーディング方法で生成されたビットストリームがコンピュータが読み取り可能な記録媒体に格納され、または、有無線通信ネットワークを介して送信されることができる。また、本文書の実施例は、プログラムコードによるコンピュータプログラム製品で具現されることができ、前記プログラムコードは、本文書の実施例によりコンピュータで実行されることができる。前記プログラムコードは、コンピュータにより読み取り可能なキャリア上に格納されることができる。

30

40

【0468】

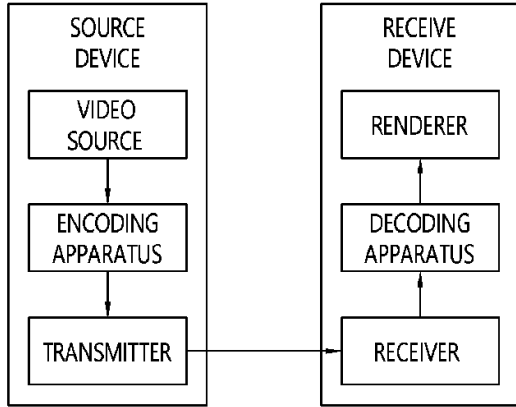
本明細書に記載された請求項は、多様な方式で組み合わせられることができる。例えば、本明細書の方法請求項の技術的特徴が組み合わせられて装置で具現されることができ、本明細書の装置請求項の技術的特徴が組み合わせられて方法で具現されることができる。また、本明細書の方法請求項の技術的特徴と装置請求項の技術的特徴が組み合わせられて装置で具現されることができ、本明細書の方法請求項の技術的特徴と装置請求項の技術的特徴が組み合わせられて方法で具現されることができる。

50

【 図 面 】

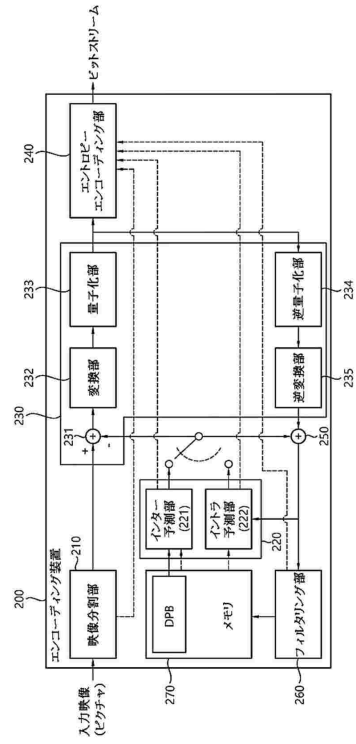
【 図 1 】

[図 1]



【 図 2 】

図 2

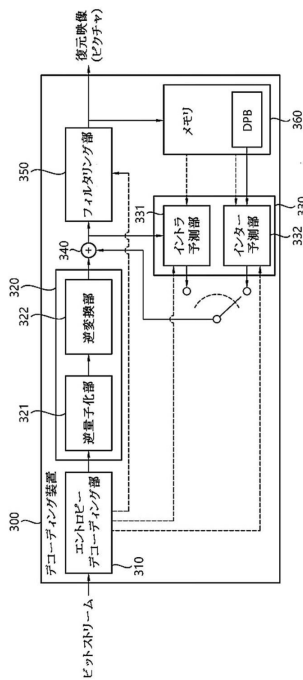


10

20

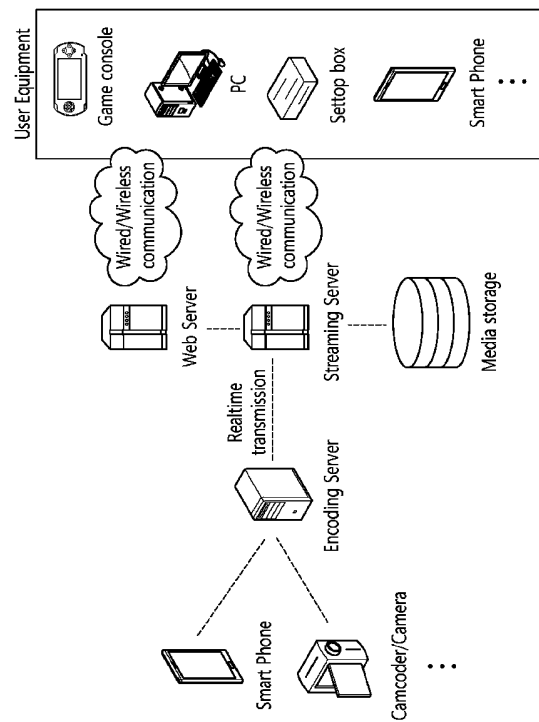
【 図 3 】

図 3



【 図 4 】

[図 4]



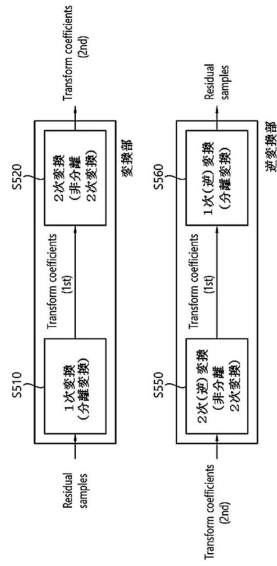
30

40

50

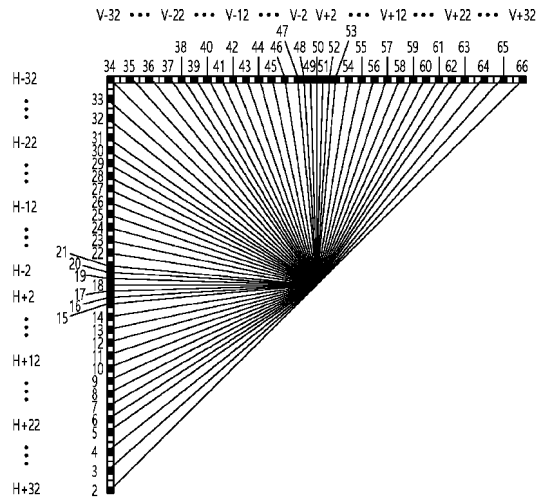
【 図 5 】

図 5



【 図 6 】

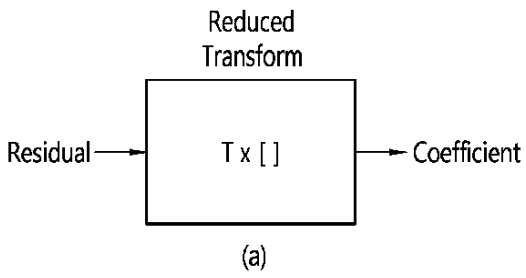
[図 6]



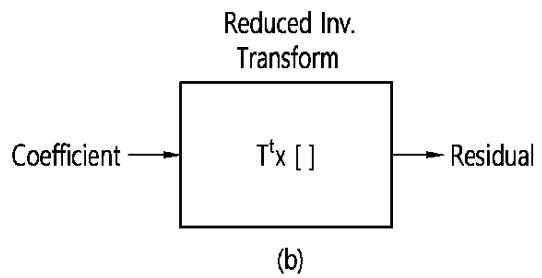
10

20

【 図 7 (a) 】



【 図 7 (b) 】



30

40

50

【 図 8 (a) 】

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36				
37	38	39	40				
41	42	43	44				
45	46	47	48				

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

(a)

【 図 8 (b) 】

1	9	17	25	33	37	41	45
2	10	18	26	34	38	42	46
3	11	19	27	35	39	43	47
4	12	20	28	36	40	44	48
5	13	21	29				
6	14	22	30				
7	15	23	31				
8	16	24	32				

1	5	9	13
2	6	10	14
3	7	11	15
4	8	12	16

(b)

10

【 図 9 (a) 】

1	3	6	10
2	5	9	13
4	8	12	15
7	11	14	16

(a)

【 図 9 (b) 】

1	3	6	X
2	5	X	X
4	8	X	X
7	X	X	X

(b)

20

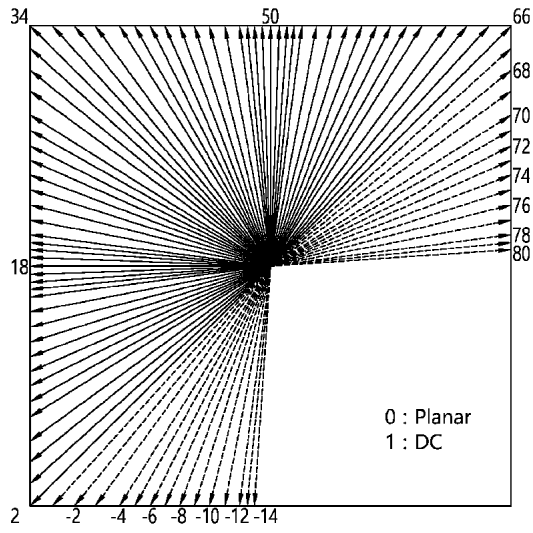
30

40

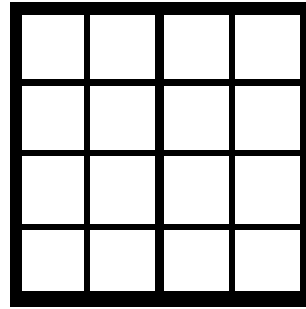
50

【 1 0 】

[E10]



【 1 1 (a) 】

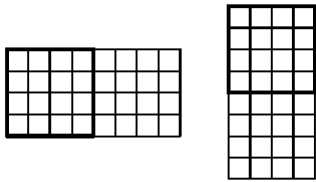


10

(a) 4x4

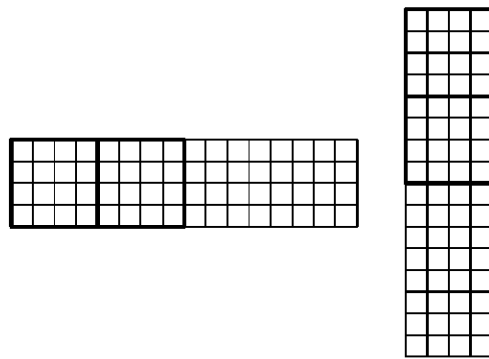
20

【 1 1 (b) 】



(b) 8x4 / 4x8

【 1 1 (c) 】



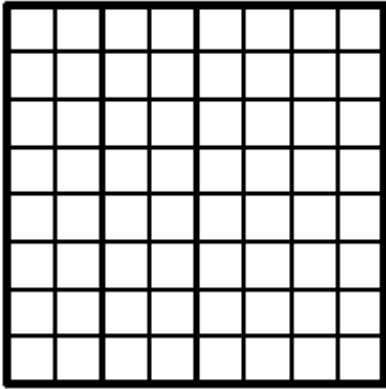
(c) 4xN / Nx4, when N≥16

30

40

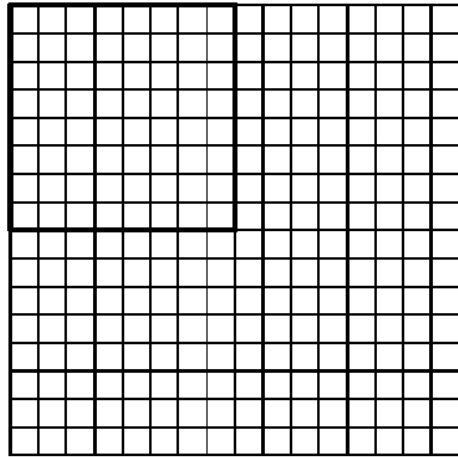
50

【図 1 1 (d)】



(d) 8x8

【図 1 1 (e)】

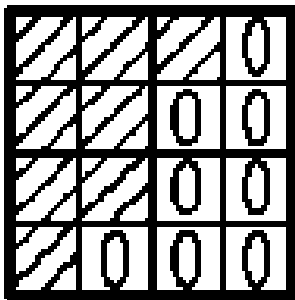


(e) MxN(M≥8, N≥8, M>8 or N>8)

10

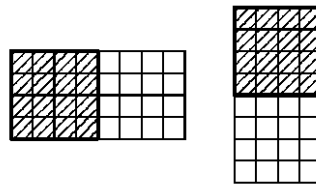
20

【図 1 2 (a)】



(a) 4x4

【図 1 2 (b)】



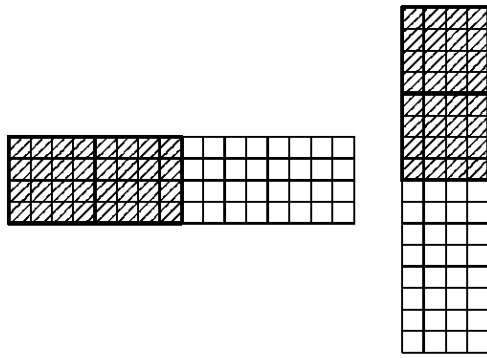
(b) 8x4 / 4x8

30

40

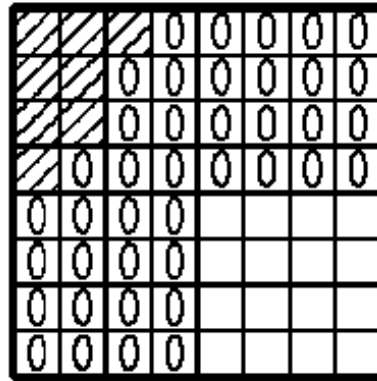
50

【 図 1 2 (c) 】



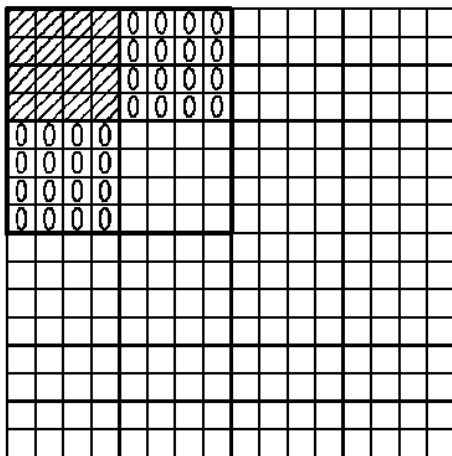
(c) $4 \times N / N \times 4$, when $N \geq 16$

【 図 1 2 (d) 】



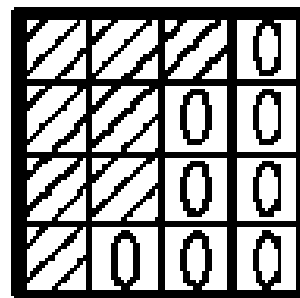
(d) 8×8

【 図 1 2 (e) 】



(e) $M \times N (M \geq 8, N \geq 8, M > 8 \text{ or } N > 8)$

【 図 1 3 (a) 】



(a) 4×4

10

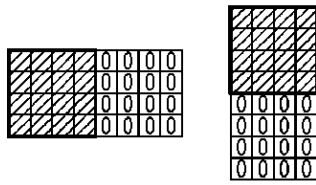
20

30

40

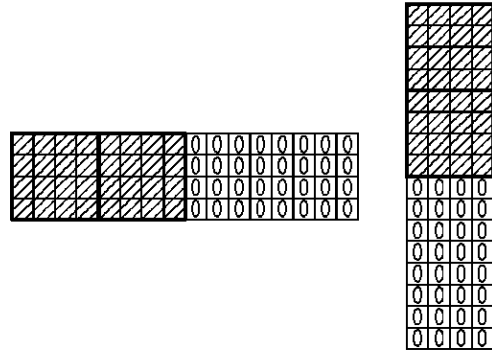
50

【 図 1 3 (b) 】



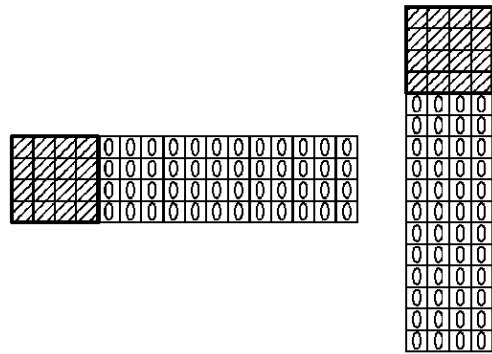
(b) 8x4 / 4x8

【 図 1 3 (c) 】



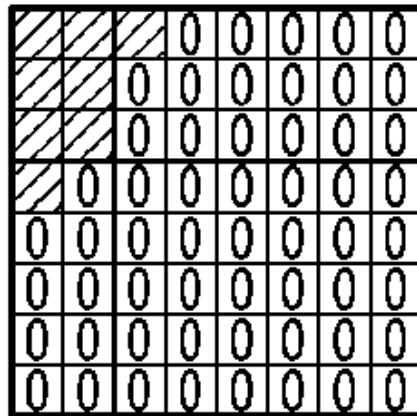
(c) 4xN / Nx4, when N ≥ 16

【 図 1 3 (d) 】



(d) 4xN / Nx4, when N ≥ 16

【 図 1 4 (a) 】



(a) 8x8

10

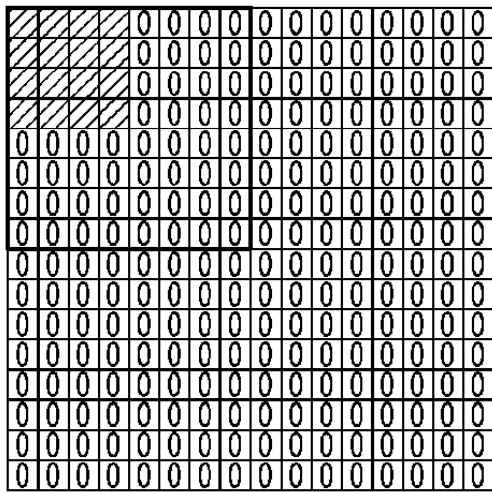
20

30

40

50

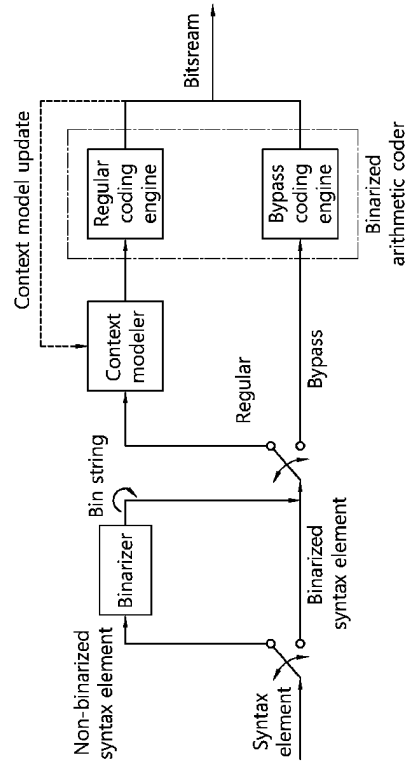
【 図 1 4 (b) 】



(b) $M \times N (M \geq 8, N \geq 8, M > 8 \text{ or } N > 8)$

【 図 1 5 】

[図 15]

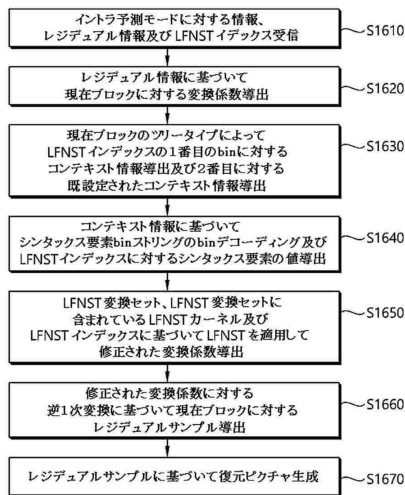


10

20

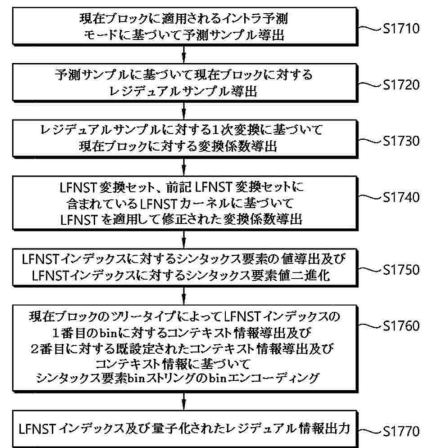
【 図 1 6 】

図 1 6



【 図 1 7 】

図 1 7



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)

弁理士 河合 章

(74)代理人 100114018

弁理士 南山 知広

(74)代理人 100159259

弁理士 竹本 実

(72)発明者 ク ムンモ

大韓民国, ソウル 0 6 7 7 2, ソチヨ - ク, ヤンジエ - デロ 1 1 - ギル, 1 9, エルジー エレクトロニクス インコーポレイティド, アイピー センター

(72)発明者 イム チェヒョン

大韓民国, ソウル 0 6 7 7 2, ソチヨ - ク, ヤンジエ - デロ 1 1 - ギル, 1 9, エルジー エレクトロニクス インコーポレイティド, アイピー センター

(72)発明者 キム スンファン

大韓民国, ソウル 0 6 7 7 2, ソチヨ - ク, ヤンジエ - デロ 1 1 - ギル, 1 9, エルジー エレクトロニクス インコーポレイティド, アイピー センター

(72)発明者 メウディ サレヒファー

大韓民国, ソウル 0 6 7 7 2, ソチヨ - ク, ヤンジエ - デロ 1 1 - ギル, 1 9, エルジー エレクトロニクス インコーポレイティド, アイピー センター

審査官 鈴木 順三

(56)参考文献 Benjamin Bross et al. , " Versatile Video Coding (Draft 6) " , [JVET-O2001-vE] (version 14) , Joint Video Experts Team(JVET) , 2019年07月31日 , pp.66,76-79,142,151-153,285-287, 304,387

Takeshi Tsukuba et al. , "Non-CE6:Context modeling for LFNST index coding" , [JVET-P027 1-v1] (version 1) , Joint Video Experts Team(JVET) , 2019年09月24日 , p.1

Jani Lainema et al. , "CE6-related:LFNST transform mapping" , [JVET-P0197-v2] (version 2) , Joint Video Experts Team(JVET) , 2019年10月02日 , p.6

Karam Naser et al. , "CE6 Related:Modified LFNST Index Coding for Fast Encoder Implementation" , [JVET-P0350-r3] (version 4) , Joint Video Experts Team(JVET) , 2019年10月06日 , p.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