

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6207790号  
(P6207790)

(45) 発行日 平成29年10月4日(2017. 10. 4)

(24) 登録日 平成29年9月15日(2017. 9. 15)

(51) Int. Cl. F I  
**HO 4 N 19/129 (2014. 01)** HO 4 N 19/129  
**HO 4 N 19/159 (2014. 01)** HO 4 N 19/159  
**HO 4 N 19/176 (2014. 01)** HO 4 N 19/176  
**HO 4 N 19/18 (2014. 01)** HO 4 N 19/18  
**HO 4 N 19/593 (2014. 01)** HO 4 N 19/593

請求項の数 10 (全 31 頁)

(21) 出願番号	特願2017-80690 (P2017-80690)	(73) 特許権者	000001007
(22) 出願日	平成29年4月14日(2017. 4. 14)		キヤノン株式会社
(62) 分割の表示	特願2014-552447 (P2014-552447) の分割		東京都大田区下丸子3丁目30番2号
原出願日	平成25年1月18日(2013. 1. 18)	(74) 代理人	100126240
(65) 公開番号	特開2017-184236 (P2017-184236A)		弁理士 阿部 琢磨
(43) 公開日	平成29年10月5日(2017. 10. 5)	(74) 代理人	100124442
審査請求日	平成29年4月14日(2017. 4. 14)		弁理士 黒岩 創吾
(31) 優先権主張番号	2012200319	(72) 発明者	クリストファー ジェームズ ロゼワーン
(32) 優先日	平成24年1月19日(2012. 1. 19)		オーストラリア国 2113 ニューサウス
(33) 優先権主張国	オーストラリア(AU)		ウェールズ州 ノースライド、トーマス
早期審査対象出願			ホルト ドライブ 1 キヤノン イン
			フォメーション システムズ リサーチ
			オーストラリア プロプライエタリー リ
			ミテッド内
			最終頁に続く

(54) 【発明の名称】 動画データを生成する方法、装置、プログラムおよび記憶媒体

(57) 【特許請求の範囲】

【請求項 1】

動画データを生成する方法であって、

所定のサイズである正方ブロックによって表わされる変換ユニットに関して、前記正方ブロックを構成する正方サブブロック内の残差係数のうちいずれかが有意係数であるか否かを示す有意係数グループフラグを含む、前記正方ブロック内の残差係数を表わすデータを取得する取得ステップと、

複数のイントラ予測モードのうちの前記変換ユニットに対応するイントラ予測モードに従って、複数のスキャン順序から、前記残差係数、および前記有意係数グループフラグのスキャン順序を決定する決定ステップと、

前記決定した有意係数グループフラグおよび残差係数のスキャン順序に従って、前記取得ステップにおいて取得されたデータから動画データを生成する生成ステップと、を有し、

前記決定ステップにおいて、

前記変換ユニットに対応するイントラ予測モードが前記複数のイントラ予測モードのうちの第1のイントラ予測モードの場合、前記残差係数のスキャン順序が第1の残差係数のスキャン順序であり、前記有意係数グループフラグのスキャン順序が第1の有意係数グループフラグのスキャン順序であると判断し、

前記イントラ予測モードが前記複数のイントラ予測モードのうちの前記第1のイントラ予測モードと異なる第2のイントラ予測モードの場合、前記残差係数のスキャン順序が第

1の残差係数のスキャン順序とは異なる第2の残差係数のスキャン順序であると判断し、前記有意係数グループフラグのスキャン順序が前記第1の有意係数グループフラグのスキャン順序とは異なる第2の有意係数グループフラグのスキャン順序であると判断する、ことを特徴とする方法。

【請求項2】

請求項1において、前記正方サブブロックは、互いに重複しない領域を表わすことを特徴とする方法。

【請求項3】

請求項1において、前記複数のスキャン順序は、後方斜め左下方向スキャン、後方水平方向スキャン、後方垂直方向スキャンを含むことを特徴とする方法。

10

【請求項4】

動画データを生成する方法を実行するためのプログラムであって、前記方法は、

所定のサイズである正方ブロックによって表わされる変換ユニットに関して、前記正方ブロックを構成する正方サブブロック内の残差係数のうちいずれかが有意係数であるか否かを示す有意係数グループフラグを含む、前記正方ブロック内の残差係数を表わすデータを取得する取得ステップと、

複数のイントラ予測モードのうちの前記変換ユニットに対応するイントラ予測モードに従って、複数のスキャン順序から、前記残差係数、および前記有意係数グループフラグのスキャン順序を決定する決定ステップと、

前記決定した有意係数グループフラグおよび残差係数のスキャン順序に従って、前記取得ステップにおいて取得されたデータから動画データを生成する生成ステップと、を有し、

20

前記決定ステップにおいて、

前記変換ユニットに対応するイントラ予測モードが前記複数のイントラ予測モードのうちの第1のイントラ予測モードの場合、前記残差係数のスキャン順序が第1の残差係数のスキャン順序であり、前記有意係数グループフラグのスキャン順序が第1の有意係数グループフラグのスキャン順序であると判断し、

前記イントラ予測モードが前記複数のイントラ予測モードのうちの前記第1のイントラ予測モードと異なる第2のイントラ予測モードの場合、前記残差係数のスキャン順序が第1の残差係数のスキャン順序とは異なる第2の残差係数のスキャン順序であると判断し、前記有意係数グループフラグのスキャン順序が前記第1の有意係数グループフラグのスキャン順序とは異なる第2の有意係数グループフラグのスキャン順序であると判断する、ことを特徴とするプログラム。

30

【請求項5】

請求項4において、前記正方サブブロックは、互いに重複しない領域を表わすことを特徴とするプログラム。

【請求項6】

請求項4において、前記複数のスキャン順序は、後方斜め左下方向スキャン、後方水平方向スキャン、後方垂直方向スキャンを含むことを特徴とするプログラム。

【請求項7】

40

請求項4ないし6のいずれか1項に記載のプログラムを記憶したコンピュータ可読記憶媒体。

【請求項8】

動画データを生成する装置であって、

所定のサイズである正方ブロックによって表わされる変換ユニットに関して、前記正方ブロックを構成する正方サブブロック内の残差係数のうちいずれかが有意係数であるか否かを示す有意係数グループフラグを含む、前記正方ブロック内の残差係数を表わすデータを取得する取得手段と、

複数のイントラ予測モードのうちの前記変換ユニットに対応するイントラ予測モードに従って、複数のスキャン順序のうち、前記残差係数、および前記有意係数グループフラ

50

グのスキャン順序を決定する決定手段と、

前記決定手段によって決定された有意係数グループフラグおよび残差係数のスキャン順序に従って、前記取得において取得されたデータから動画データを生成する生成手段と、を有し、

前記決定手段は、

前記変換ユニットに対応するイントラ予測モードが前記複数のイントラ予測モードのうちの第1のイントラ予測モードの場合、前記残差係数のスキャン順序が第1の残差係数のスキャン順序であり、前記有意係数グループフラグのスキャン順序が第1の有意係数グループフラグのスキャン順序であると判断し、

前記イントラ予測モードが前記複数のイントラ予測モードのうちの前記第1のイントラ予測モードと異なる第2のイントラ予測モードの場合、前記残差係数のスキャン順序が第1の残差係数のスキャン順序とは異なる第2の残差係数のスキャン順序であると判断し、前記有意係数グループフラグのスキャン順序が前記第1の有意係数グループフラグのスキャン順序とは異なる第2の有意係数グループフラグのスキャン順序であると判断する、ことを特徴とする生成装置。

【請求項9】

請求項8において、前記正方サブブロックは、互いに重複しない領域を表わすことを特徴とする生成装置。

【請求項10】

請求項8において、前記複数のスキャン順序は、後方斜め左下方向スキャン、後方水平方向スキャン、後方垂直方向スキャンを含むことを特徴とする生成装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、一般に、デジタル動画信号処理に関し、特に、変換ユニット(TU)の残差係数の符号化および復号化のための方法、装置、およびシステムに関し、変換ユニット(TU)は、正方形または非正方形とすることができる。

【背景技術】

【0002】

現在、動画データの伝送および記憶のための実用例を含む、動画符号化のための多くの実用例が存在する。多くの動画符号化規格もまた開発されており、他も現在開発中である。動画符号化規格における最近の動向では、「映像符号化共同研究部会」(Joint Collaborative Team on Video Coding: JCT-VC)と呼ばれるグループが結成されている。映像符号化共同研究部会(JCT-VC)は、動画符号化専門家グループ(VCEG)として知られる、国際電気通信連合(ITU)の電気通信標準化部門(ITU-T)の研究委員会16、研究課題6(SG16/Q6)のメンバー、および動画専門家組織(MPEG)としても知られる、国際標準化機構/国際電気標準会議合同技術委員会1/専門部会29/作業グループ11(ISO/IEC JTC1/SC29/WG11)のメンバーを含む。

【0003】

映像符号化共同研究部会(JCT-VC)は、「H.264/MPEG-4 AVC」として知られる既存の動画符号化規格より著しく優れている新しい動画符号化規格を作り出すことを目的としている。H.264/MPEG-4 AVC規格は、それ自体が、MPEG-4およびITU-T H.263などの以前の動画符号化規格を大幅に改善したものである。開発中の新しい動画符号化規格は、「高効率動画符号化(high efficiency video coding: HEVC)」と名称付けられている。映像符号化共同研究部会JCT-VCはまた、高解像度または高フレームレートで動作するように規格の実装をスケールアップする場合に困難を生じるという、高効率動画符号化(HEVC)のために提案された技術から生じる実装上の問題を考慮している。

【0004】

高圧縮効率の実現に対して困難を呈する H. 264 / M P E G - 4 A V C 動画符号化規格の 1 つの領域は、動画データを表すために使用される残差係数の符号化である。動画データは、一連のフレームにより形成され、各フレームは、サンプルの 2 次元配列を有する。典型的には、フレームは、1 つの輝度チャンネルと、2 つの色差チャンネルを有する。各フレームは、最大符号化ユニット ( L C U ) の配列に分解される。最大符号化ユニット ( L C U ) は、固定サイズを有し、エッジ寸法は、64 の輝度サンプルなどの、2 のべき乗で、等しい幅と高さを有する。符号化ツリーは、各最大符号化ユニット ( L C U ) を 4 つの符号化ユニット ( C U ) に細分割することを可能にし、それぞれの幅および高さが、もとの最大符号化ユニット ( L C U ) の半分である。各符号化ユニット ( C U ) は、さらに、4 つの等しいサイズの符号化ユニット ( C U ) に分割してもよい。そのような細分割処理は、最小符号化ユニット ( S C U ) サイズに達するまで再帰的に適用してもよく、符号化ユニット ( C U ) は、最小サポートサイズまで定義することができる。最大符号化ユニットの、符号化ユニットの階層への再帰的細分化は、四分木構造を有し、符号化ツリーと称される。この細分化処理は、一連のフラグとして通信ビットストリームで符号化され、ピンとしてコード化される。したがって、符号化ユニットは、正方形である。

#### 【 0 0 0 5 】

符号化ツリー内には、それ以上細分割されない 1 組の符号化ユニットが存在し、符号化ツリーの葉ノードを占める。変換ツリーは、これらの符号化ユニットで存在する。さらに、変換ツリーは、符号化ツリーで使用するような四分木構造を使用して、符号化ユニットを分解してもよい。変換ツリーの葉ノードでは、残差データが、変換ユニット ( T U ) を使用して符号化される。符号化ツリーとは対照的に、変換ツリーは、符号化ユニットを、非正方形の変換ユニットに細分割することができる。さらに、変換ツリー構造は、変換ユニット ( T U ) が、もとの符号化ユニットによってもたらされる領域のすべてを占めることを要求しない。

#### 【 0 0 0 6 】

符号化ツリーの葉ノードでの各符号化ユニットは、予測されたデータサンプルの 1 つまたは複数の配列に細分割され、それぞれが、予測ユニット ( P U ) として知られる。各予測ユニット ( P U ) は、イントラ予測処理またはインター予測処理を適用することによって導出される、入力動画フレームデータの一部の予測を含む。いくつかの方法を使用して、符号化ユニット ( C U ) 内で予測ユニット ( P U ) を符号化することができる。単一の予測ユニット ( P U ) は、符号化ユニット ( C U ) の領域全体を占めてもよく、または、符号化ユニット ( C U ) は、水平に、もしくは垂直に、2 つの等しいサイズの矩形予測ユニット ( P U ) に分割してもよい。さらに、符号化ユニット ( C U ) は、4 つの等しいサイズの正方予測ユニット ( P U ) に分割してもよい。

#### 【 0 0 0 7 】

動画符号器は、動画データを一連の構文要素に変換することによって、動画データをビットストリームに圧縮する。コンテキスト適応型 2 進算術符号化 ( c o n t e x t a d a p t i v e b i n a r y a r i t h m e t i c c o d i n g : C A B A C ) スキームが、開発中の高効率動画符号化 ( H E V C ) 規格において定義され、M P E G 4 - A V C / H. 264 動画圧縮規格で定義されるような同一の算術符号化スキームを使用する。開発中の高効率動画符号化 ( H E V C ) 規格では、コンテキスト適応型 2 進算術符号化 ( C A B A C ) が使用される場合、各構文要素は、一連のピンとして表され、ピンは、利用可能なピンのセットから選択される。利用可能なピンのセットは、コンテキストモデルから取得され、ピン毎に 1 コンテキストを有する。各コンテキストは、可能性の高いピン値 ( v a l M P S )、および算術符号化演算もしくは算術復号化演算のための確率状態を保持する。ピンはまた、バイパスコード化されてもよく、その場合、コンテキストとの関連はない。バイパスコード化されたピンは、ビットストリーム内の 1 ビットを消費し、したがって、1 値または 0 値となる確率が等しいピンと適合する。構文要素からそのような一連のピンを作り出すことは、構文要素「に起因するピン」として知られる。

#### 【 0 0 0 8 】

動画符号器または動画復号器では、別々のコンテキスト情報が各ピンに対して利用可能である場合、ピンに対するコンテキスト選択は、符号化効率を改善するための手段をもたらす。特に、符号化効率は、ピンの以前のインスタンスからの統計学的性質が、関連したコンテキスト情報を使用して、ピンの現在のインスタンスの統計学的性質と相関するように、特定のピンを選択することによって改善することができる。そのようなコンテキスト選択は、空間的ローカル情報をよく使用して、最適なコンテキストを決定する。

#### 【 0 0 0 9 】

開発中の高効率動画符号化 ( H E V C ) 規格、および H . 2 6 4 / M P E G - 4 A V C では、現在ブロックに対する予測が、他のフレームからの、または以前に復号化された現在ブロック内の隣接領域からの参照サンプルデータに基づいて導出される。予測と所望のサンプルデータとの間の差異は、残差として知られる。残差の周波数領域表現は、残差係数の 2 次元配列である。慣習的に 2 次元配列の左上隅は、低周波数情報を表す残差係数を備える。

10

#### 【 0 0 1 0 】

典型的な動画データでは、サンプル値の変更の大部分は、段階的であり、残差内に圧倒的多数の低周波数情報をもたらす結果となる。これは、2 次元配列の左上隅に位置する残差係数に対する強度がより高いことを明らかにする。

#### 【 0 0 1 1 】

残差係数の 2 次元配列の左上隅において圧倒的多数を占める低周波数情報の特性は、選択された 2 進化スキームによって利用され、ビットストリームにおける残差係数のサイズを最小化することができる。

20

#### 【 0 0 1 2 】

H M - 5 . 0 は、変換ユニット ( T U ) を、多数のサブセットに分割し、2 つのパス内の各サブセットにおける残差係数をスキャンする。第 1 のパスは、非ゼロ値 ( 有意 ) またはゼロ値 ( 非有意 ) であるとして残差係数の状態を示すフラグを符号化する。このデータは、有意度マップとして知られる。第 2 のパスは、係数レベルとして知られる、有意残差係数の強度および符号を符号化する。

#### 【 0 0 1 3 】

提供されたスキャンパターンは、残差係数の 2 次元配列を、1 次元配列にスキャンすることを可能にする。H M - 5 . 0 では、提供されたスキャンパターンは、有意度マップと係数レベルとの両方を処理するために使用される。提供されたスキャンパターンを使用して有意度マップをスキャンすることによって、2 次元有意度マップにおける最終有意係数の位置を判断することができる。スキャンパターンは、水平、垂直、または斜めとすることができる。

30

#### 【 0 0 1 4 】

高効率動画符号化 ( H E V C ) テストモデル 5 . 0 ( H M - 5 . 0 ) は、正方形および非正方形の両方である変換ユニット ( T U ) としても知られる、残差ブロックに対するサポートをもたらす。各変換ユニット ( T U ) は、残差係数の 1 セットを備える。等しいサイズの辺寸法を有する残差ブロックは、正方形の変換ユニット ( T U ) として知られ、等しくないサイズの辺寸法を有する残差ブロックは、非正方形の変換ユニット ( T U ) として知られる。

40

#### 【 0 0 1 5 】

H M - 5 . 0 でサポートされる変換ユニット ( T U ) サイズは、 $4 \times 4$ 、 $8 \times 8$ 、 $16 \times 16$ 、 $32 \times 32$ 、 $4 \times 16$ 、 $16 \times 4$ 、 $8 \times 32$ 、および  $32 \times 8$  である。典型的には、変換ユニット ( T U ) サイズは、輝度サンプルの点から記載されるが、 $4 : 2 : 0$  の色差フォーマットが使用される場合、各色差サンプルは、 $2 \times 2$  輝度サンプルの領域を占める。それに応じて、変換ユニット ( T U ) をスキャンして、色差残差データを符号化するために、 $4 \times 4$  輝度残差ブロックの場合に  $2 \times 2$  であるような、水平および垂直寸法が半分のスキャンパターンを使用する。残差係数のスキャンおよびコード化のために、 $16 \times 16$ 、 $32 \times 32$ 、 $4 \times 16$ 、 $16 \times 4$ 、 $8 \times 32$ 、および  $32 \times 8$  変換ユニット (

50

TU)が、サイズが $4 \times 4$ で、HM-5.0内に存在する対応するマップを用いて、多数のサブブロック、すなわち、変換ユニット(TU)スキンの下位レイヤに分割される。HM-5.0では、これらの変換ユニット(TU)サイズに対するサブブロックは、変換ユニット(TU)におけるサブセットと共に配置される。あるサブブロック内で共に配置される有意度マップの一部内の設定された有意係数フラグは、有意係数グループと称される。 $16 \times 16$ 、 $32 \times 32$ 、 $4 \times 16$ 、 $16 \times 4$ 、 $8 \times 32$ 、および $32 \times 8$ 変換ユニット(TU)に対して、有意度マップコード化は、2レベルスキンを利用する。上位レベルスキンは、後方斜め左下方向スキン(backward diagonal down-left scan)などのスキンを実行し、各サブブロックの有意係数グループを表すフラグをコード化または推測する。サブブロック内では、後方斜め左下方向スキンなどのスキンが実行され、1値の有意係数グループフラグを有するサブブロックに対する有意係数フラグをコード化する。 $16 \times 16$ 変換ユニット(TU)の場合には、 $4 \times 4$ 上位スキンを用いる。 $32 \times 32$ 変換ユニット(TU)の場合には、 $8 \times 8$ 上位スキンを用いる。 $16 \times 4$ 、 $4 \times 16$ 、 $32 \times 8$ 、および $8 \times 32$ 変換ユニット(TU)サイズの場合には、それぞれ、 $4 \times 1$ 、 $1 \times 4$ 、 $8 \times 2$ 、および $2 \times 8$ 上位スキンを用いる。

10

#### 【0016】

各変換ユニット(TU)では、残差係数データは、ビットストリームに符号化してもよい。各「残差係数」は、周波数(DCT)領域における変換ユニット内の画像特性を表し、変換ユニット内で独自の位置を占める数である。変換ユニットは、空間領域と周波数領域との間で変換することができる残差データサンプルのブロックである。周波数領域では、変換ユニット(TU)は、残差係数データとして、残差データサンプルを符号化する。変換ユニットの辺寸法は、2のべき乗で形成され、「輝度」チャンネルに対して4サンプルから32サンプルの範囲であり、「色差」チャンネルに対して2から16サンプルの範囲である。変換ユニット(TU)ツリーの葉ノードは、変換ユニット(TU)を含有するか、または、残差係数データを必要としない場合には、全く含有しない可能性がある。

20

#### 【0017】

変換ユニットの空間表現として、残差データサンプルの2次元配列があり、以下に詳細に説明するように、変形離散コサイン変換(DCT)などの変換からもたらされる周波数領域表現もまた、残差係数の2次元配列である。変換ユニット(TU)内の典型的なサンプルデータのスペクトル特性は、周波数領域表現が、空間表現よりもコンパクトであることなどである。さらに、変換ユニット(TU)において典型的な圧倒的多数の低周波数スペクトル情報は、変換ユニット(TU)の左上に向かってより大きな値の残差係数のクラスタリングをもたらし、低周波数残差係数が表される。

30

#### 【0018】

変形離散コサイン変換(DCT)または変形離散サイン変換(DST)は、残差変換を実施するために使用することができる。残差変換の実施は、各必要な変換ユニット(TU)サイズをサポートするよう構成される。動画符号器では、残差変換からの残差係数は、スケール変更され、量子化される。スケーリングおよび量子化により、残差係数の大きさが低減し、画質を低減するという代償を払って、ビットストリームにコード化されたデータのサイズを低減する。

40

#### 【0019】

開発中の高効率動画符号化(HEVC)規格の複雑度の一態様は、スキニングを実行するために必要な参照テーブルの数である。参照テーブルを追加することにメモリの不要な消費をもたらすため、必要な参照テーブルの数を減らすことは、複雑度を低減する一態様である。

#### 【発明の概要】

#### 【発明が解決しようとする課題】

#### 【0020】

本発明の目的は、既存の配列の1つまたは複数の不利な点を実質的に克服する、または、

50

少なくとも、改善することである。

【課題を解決するための手段】

【0021】

本開示の一態様では、動画データのビットストリームから変換ユニットの残差係数を復号化する方法を提供する。本方法は、

動画データのビットストリームから変換ユニットを受信するステップであって、変換ユニットは、上位正方レイヤおよび下位正方レイヤを有し、上位レイヤは、最大4つの有意係数グループフラグの、有意係数グループフラグのそれぞれが下位レイヤの非重複領域を表す場合の正方配列を表す、ステップと、

受信した変換ユニットに対する正方上位レイヤの有意係数グループフラグを決定するステップと、

決定した有意係数グループフラグに従って、正方下位レイヤの残差係数の値を決定して、動画データのビットストリームの変換ユニットを復号化するステップと

を備える。

【0022】

最大4つの有意係数グループフラグによって表される非重複領域のそれぞれは、それぞれが16の残差係数を有する正方形領域であることが好ましい。

【0023】

上位レイヤが4つの有意係数フラグを有することが望ましく、本方法は、所定のスキャンパターンにオフセットを適用することによって、下位レイヤの非重複領域のそれぞれに対するスキャンパターンを決定する、さらなるステップを備える。

【0024】

本方法はまた、ビットストリームで符号化された有意係数フラグを復号化することによって、正方上位レイヤの有意係数グループフラグを決定してもよい。

【0025】

正方上位レイヤの有意係数グループフラグを決定することは、有意係数フラグの少なくとも1つを推測することを備えることが好ましい。

【0026】

変換ユニットの下位レイヤがサブブロックとして形成され、決定した有意係数グループフラグに従って、正方下位レイヤの残差係数の値を決定することは、変換ユニットのサブブロックのそれぞれを繰り返し処理することを備えることが有利である。この繰り返し処理により、サブブロックのそれぞれを表す線形配列を形成し、対応するオフセット値を使用して、線形配列から各サブブロックを再構成することが望ましい。

【0027】

特定の実装形態では、上位レイヤは、4つの有意係数グループフラグの正方配列を表す。

【0028】

一実装形態では、変換ユニットの下位レイヤのスキャン方向は、変換ユニットに適用されるイントラ予測モードに従って選択される。他の実装形態では、変換ユニットの上位レイヤのスキャン方向は、変換ユニットに適用されるイントラ予測モードに従って選択される。さらに、変換ユニットの上位レイヤのスキャン方向は、斜め方向であり、変換ユニットの下位レイヤのスキャン方向から独立している。

【0029】

本開示の別の態様では、動画データのビットストリームにおける変換ユニットの残差係数を符号化する方法を提供する。本方法は、

変換ユニットの正方下位レイヤに対する残差係数の値を受信するステップと、

変換ユニットの正方上位レイヤに対する有意係数グループフラグを決定するステップであって、上位レイヤは、最大4つの有意係数グループフラグの、有意係数グループフラグのそれぞれが下位レイヤの非重複領域を表す場合の、正方配列を表す、ステップと、

正方形下位レイヤに対する残差係数の値と、正方上位レイヤの有意係数グループフラグとを符号化し、動画データのビットストリームにおける変換ユニットを符号化するステップ

10

20

30

40

50

と  
を備える。

【0030】

別の態様では、動画データを生成する方法を提供する。本方法は、動画データを生成する方法であって、

所定のサイズである正方ブロックによって表わされる変換ユニットに関して、前記正方ブロックを構成する正方サブブロック内の残差係数のうちいずれかが有意係数であるか否かを示す有意係数グループフラグを含む、前記正方ブロック内の残差係数を表わすデータを取得する取得ステップと、

複数のイントラ予測モードのうちの前記変換ユニットに対応するイントラ予測モードに従って、複数のスキャン順序から、前記残差係数、および前記有意係数グループフラグのスキャン順序を決定する決定ステップと、

前記決定した有意係数グループフラグおよび残差係数のスキャン順序に従って、前記取得ステップにおいて取得されたデータから動画データを生成する生成ステップと、を有し、

前記決定ステップにおいて、

前記変換ユニットに対応するイントラ予測モードが前記複数のイントラ予測モードのうちの第1のイントラ予測モードの場合、前記残差係数のスキャン順序が第1の残差係数のスキャン順序であり、前記有意係数グループフラグのスキャン順序が第1の有意係数グループフラグのスキャン順序であると判断し、

前記イントラ予測モードが前記複数のイントラ予測モードのうちの前記第1のイントラ予測モードと異なる第2のイントラ予測モードの場合、前記残差係数のスキャン順序が第1の残差係数のスキャン順序とは異なる第2の残差係数のスキャン順序であると判断し、前記有意係数グループフラグのスキャン順序が前記第1の有意係数グループフラグのスキャン順序とは異なる第2の有意係数グループフラグのスキャン順序であると判断する、ことを特徴とする。

【0031】

他の態様もまた開示する。

【0032】

本発明の少なくとも一実施形態を、以下の図面を参照して、ここで説明する。

【図面の簡単な説明】

【0033】

【図1】動画符号器の機能モジュールを示す模式的ブロック図である。

【図2】動画復号器の機能モジュールを示す模式的ブロック図である。

【図3(A)】図1の符号器で実施することができる汎用コンピュータシステムの模式的ブロック図である。

【図3(B)】図2の復号器で実施することができる汎用コンピュータシステムの模式的ブロック図である。

【図4】エントロピー符号器の機能モジュールを示す模式的ブロック図である。

【図5】エントロピー復号器の機能モジュールを示す模式的ブロック図である。

【図6】 $8 \times 8$ 変換ユニット(TU)の残差係数を符号化するための従来方法を示すフローダイアグラムである。

【図7】 $8 \times 8$ 変換ユニット(TU)の残差係数を復号化するための従来方法を示すフローダイアグラムである。

【図8】有意係数グループを使用して、 $8 \times 8$ 変換ユニット(TU)の残差係数を符号化するための、本開示による方法を示すフローダイアグラムである。

【図9】有意係数グループを使用して、 $8 \times 8$ 変換ユニット(TU)の残差係数を復号化するための、本開示による方法を示すフローダイアグラムである。

【図10】(A) $8 \times 8$ 変換ユニット(TU)の有意係数のグループを表すための、本開示による方法である。(B) $8 \times 8$ 変換ユニット(TU)の有意係数のグループを表す

10

20

30

40

50

ための、本開示による方法である。

【図 1 1】後方斜め方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための従来方法を示す図である。

【図 1 2】後方水平方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための従来方法を示す図である。

【図 1 3】後方垂直方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための従来方法を示す図である。

【図 1 4】2 レイヤ階層を使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための、本開示による方法を示す図である。

【図 1 5】(A) 後方斜め方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための、本開示による方法を示す図である。(B) 後方斜め方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための、本開示による方法を示す図である。(C) 後方斜め方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための、本開示による方法を示す図である。

10

【図 1 6】(A) 後方水平方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための、本開示による方法を示す図である。(B) 後方水平方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための、本開示による方法を示す図である。(C) 後方水平方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための、本開示による方法を示す図である。

【図 1 7】(A) 後方垂直方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための、本開示による方法を示す図である。(B) 後方垂直方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための、本開示による方法を示す図である。(C) 後方垂直方向スキャンを使用して、 $8 \times 8$  変換ユニット (TU) の残差係数をスキャンするための、本開示による方法を示す図である。

20

【図 1 8】(A) オフセットを利用する、スキャンングのための代替手法である。(B) オフセットを利用する、スキャンングのための代替手法である。(C) オフセットを利用する、スキャンングのための代替手法である。

【発明を実施するための形態】

【0034】

添付図面のいずれか一つまたは複数において、同じ参照番号を有するステップおよび/または特徴を参照する場合、それらのステップおよび/または特徴は、本説明の目的において、別異の意味が明らかでない限り、同じ機能または動作を意味する。

30

【0035】

図 1 は、動画符号器 100 の機能モジュールを示す模式的ブロック図である。図 2 は、対応する動画復号器 200 の機能モジュールを示す模式的ブロック図である。動画符号器 100 および動画復号器 200 は、図 3 A および図 3 B に示すような汎用コンピュータシステム 300 を使用して実現することができ、さまざまな機能モジュールを、コンピュータシステム 300 内の専用ハードウェアによって、コンピュータシステム 300 内で実行可能なソフトウェアによって、またはコンピュータシステム 300 内の専用ハードウェアおよび実行可能なソフトウェアの組み合わせによって実施することができる。

40

【0036】

図 3 A から分かるとおり、コンピュータシステム 300 は、コンピュータモジュール 301 と、キーボード 302、マウスポインタデバイス 303、スキャナ 326、カメラ 327、およびマイクロフォン 380 などの入力デバイスと、プリンタ 315、ディスプレイデバイス 314、および拡声器 317 などの出力デバイスとを備える。外部の変調器 - 復調器 (モデム) トランシーバデバイス 316 は、コンピュータモジュール 301 が使用し、接続部 321 を介して、通信ネットワーク 320 と通信することができる。通信ネットワーク 320 は、インターネット、セルラー電子通信ネットワーク、またはプライベート WAN などのワイドエリアネットワーク (WAN) としてもよい。接続部 321 が電話線である場合、モデム 316 は、従来の「ダイヤルアップ」モデムとしてもよい。あるいは

50

、接続部 3 2 1 が、大容量（例えば、ケーブル）接続である場合、モデム 3 1 6 は、ブロードバンドモデムとしてもよい。通信ネットワーク 3 2 0 に無線接続するために、無線モデムを使用してもよい。

#### 【 0 0 3 7 】

典型的には、コンピュータモジュール 3 0 1 は、少なくとも 1 つのプロセッサユニット 3 0 5、およびメモリユニット 3 0 6 を備える。例えば、メモリユニット 3 0 6 は、半導体ランダム・アクセス・メモリ（RAM）および半導体リード・オンリー・メモリ（ROM）を有してもよい。コンピュータモジュール 3 0 1 はまた、多数の入力／出力（I/O）インターフェース、すなわち、ビデオディスプレイ 3 1 4、拡声器 3 1 7、およびマイクロフォン 3 8 0 に結合するオーディオ・ビデオインターフェース 3 0 7 と、キーボード 3 0 2、マウス 3 0 3、スキャナ 3 2 6、カメラ 3 2 7、およびオブションのジョイスティックまたは他のヒューマン・インターフェース・デバイス（図示せず）に結合する I/O インターフェース 3 1 3 と、外部モデム 3 1 6 およびプリンタ 3 1 5 のためのインターフェース 3 0 8 とを備える。実装形態によっては、モデム 3 1 6 は、コンピュータモジュール 3 0 1 内に、例えば、インターフェース 3 0 8 内に、組み込んでもよい。コンピュータモジュール 3 0 1 はまた、ローカル・ネットワーク・インターフェース 3 1 1 を有し、ローカル・ネットワーク・インターフェース 3 1 1 は、ローカルエリアネットワーク（LAN）として知られるローカルエリア通信ネットワーク 3 2 2 に、接続部 3 2 3 を介して、コンピュータシステム 3 0 0 を接続することを可能にする。図 3 A に図示したように、ローカル通信ネットワーク 3 2 2 はまた、接続部 3 2 4 を介して、ワイドネットワーク 3 2 0 に接続してもよく、接続部 3 2 4 は、典型的には、いわゆる「ファイアウォール」デバイスまたは同様の機能を有するデバイスを備える。ローカル・ネットワーク・インターフェース 3 1 1 は、Ethernet（登録商標）回路カード、Bluetooth（登録商標）無線装置、または IEEE 8 0 2 . 1 1 無線装置を備えることができるが、多数の他の種類のインターフェースを、インターフェース 3 1 1 として用いてもよい。

#### 【 0 0 3 8 】

I/O インターフェース 3 0 8 および 3 1 3 は、シリアル接続およびパラレル接続のいずれか、またはその両方とすることができ、前者は、典型的には、ユニバーサルシリアルバス（USB）規格に従って実現され、対応する USB 接続部（図示せず）を有する。記憶デバイス 3 0 9 が設置され、典型的には、ハードディスクドライブ（HDD）3 1 0 を含む。フロッピー（登録商標）・ディスク・ドライブおよび磁気テープドライブ（図示せず）などの他の記憶デバイスもまた、使用してもよい。典型的には、光学ディスクドライブ 3 1 2 が、データの不揮発性ソースとして動作するよう設けられる。例えば、光学ディスク（例えば、CD-ROM、DVD、Blu-ray（登録商標）Disc）、USB-RAM、ポータブル、外部ハードドライブ、およびフロッピー（登録商標）ディスクなどのポータブル・メモリ・デバイスを、システム 3 0 0 への適切なデータソースとして使用してもよい。典型的には、HDD 3 1 0、光学ドライブ 3 1 2、ネットワーク 3 2 0 および 3 2 2、またはカメラ 3 2 7 のいずれかは、符号化される動画データに対するソースであり、またディスプレイ 3 1 4 は、格納もしくは再生される復号化された動画データの送信先となる可能性がある。

#### 【 0 0 3 9 】

典型的には、コンピュータモジュール 3 0 1 の構成要素 3 0 5 から 3 1 3 は、相互接続バス 3 0 4 を介し、当業者に既知である、コンピュータシステム 3 0 0 の動作における従来のモードで通信する。例えば、プロセッサ 3 0 5 は、接続部 3 1 8 を使用して、システムバス 3 0 4 に結合される。同じように、メモリ 3 0 6 および光学ディスクドライブ 3 1 2 は、接続部 3 1 9 によって、システムバス 3 0 4 に結合される。記載した構成を実現することが可能なコンピュータの例には、IBM-PC ならびに互換機、Sun Sparc station、Apple Mac（商標）、または同様のコンピュータシステムがある。

#### 【 0 0 4 0 】

10

20

30

40

50

必要に応じて、または所望する場合、符号器 100 および復号器 200 は、以下に説明する方法と同様に、コンピュータシステム 300 を使用して実現してもよく、符号器 100、復号器 200、ならびに図 10 および図 11 の処理は、説明するように、コンピュータシステム 300 内で実行可能な、1 つまたは複数のソフトウェア・アプリケーション・プログラム 333 として実現してもよい。特に、符号器 100、復号器 200、および記載した方法のステップは、コンピュータシステム 300 内で実行されるソフトウェア 333 における命令 331 (図 3 B 参照) によって実行される。ソフトウェア命令 331 は、1 つまたは複数のコードモジュールとして形成してもよく、それぞれが、1 つまたは複数の特定のタスクを実行する。ソフトウェアはまた、2 つの別々の部分に分割してもよく、第 1 の部分および対応するコードモジュールは、説明した方法を実行し、第 2 の部分および対応するコードモジュールは、第 1 の部分とユーザとの間のユーザインターフェースを管理する。

10

#### 【0041】

ソフトウェアは、コンピュータ可読媒体に格納することができ、コンピュータ可読媒体は、例えば、以下で説明する記憶デバイスを含む。ソフトウェアは、コンピュータ可読媒体からコンピュータシステム 300 内にロードされ、次いで、コンピュータシステム 300 によって実行される。コンピュータ可読媒体上に記録された、そのようなソフトウェアまたはコンピュータプログラムを有するコンピュータ可読媒体は、コンピュータプログラム製品である。コンピュータシステム 300 においてコンピュータプログラム製品を使用することにより、符号器 100、復号器 200、および説明した方法を実現するために有利な装置をもたらすことが好ましい。

20

#### 【0042】

典型的には、ソフトウェア 333 は、HDD 310 またはメモリ 306 に格納される。ソフトウェアは、コンピュータ可読媒体からコンピュータシステム 300 内にロードされ、コンピュータシステム 300 によって実行される。したがって、例えば、ソフトウェア 333 は、光学ディスクドライブ 312 によって読み取られる光学可読ディスク記憶媒体 (例えば、CD-ROM) 325 上に記憶してもよい。

#### 【0043】

場合によっては、アプリケーションプログラム 333 は、1 つまたは複数の CD-ROM 325 で符号化されてユーザにもたらされ、対応するドライブ 312 を介して読み込まれ、または、ネットワーク 320 もしくは 322 からユーザによって読み込まれてもよい。またさらに、ソフトウェアは、他のコンピュータ可読媒体からコンピュータシステム 300 内にロードすることもできる。コンピュータ可読記憶媒体は、コンピュータシステム 300 で実行および / または処理するための、記録された命令および / またはデータをもたらす、任意の非一時的有形記憶媒体を意味する。そのような記憶媒体の例には、フロッピー (登録商標) ディスク、磁気テープ、CD-ROM、DVD、Blu-ray (登録商標) ディスク、ハードディスクドライブ、ROM もしくは集積回路、USB メモリ、光磁気ディスク、または PCMCIA カードなどのコンピュータ可読カードがあり、そのようなデバイスは、コンピュータモジュール 301 に内蔵または外付けする。ソフトウェア、アプリケーションプログラム、命令、および / または動画データもしくは符号化された動画データをコンピュータモジュール 301 にもたらす可能性のある一時的または非有形コンピュータ可読伝送媒体の例には、他のコンピュータまたはネットワークデバイスへの無線もしくは赤外線伝送チャネルおよびネットワーク接続、ならびに電子メール伝送およびウェブサイトなどに記録された情報を含むインターネットまたはイントラネットがある。

30

40

#### 【0044】

アプリケーションプログラム 333 の第 2 の部分および上記の対応するコードモジュールは、ディスプレイ 314 上で表現されるか、または表示される、1 つまたは複数のグラフィカル・ユーザ・インターフェース (GUI) を実現するよう実行してもよい。典型的には、キーボード 302 およびマウス 303 の操作を通じて、コンピュータシステム 300 およびアプリケーションのユーザは、GUI と関連したアプリケーションに制御コマンド

50

を送りおよび／または入力するよう機能的に適合可能な方法でインターフェースを操作することができる。機能的に適合可能なユーザインターフェースの他の形態は、拡声器 317 を介して出力される音声プロンプトおよびマイクロフォン 380 を介して入力されるユーザ音声コマンドなどを使用するオーディオインターフェースなどとして実現してもよい。

#### 【0045】

図 3 B は、プロセッサ 305 および「メモリ」334 の詳細な模式的ブロック図である。メモリ 334 は、図 3 A におけるコンピュータモジュール 301 によってアクセス可能な全メモリモジュール（HDD 309 および半導体メモリ 306 を含む）の論理集合を表す。

10

#### 【0046】

コンピュータモジュール 301 は、最初に、電源を入れられると、電源投入時の自己診断（POST）プログラム 350 を実行する。典型的には、POST プログラム 350 は、図 3 A の半導体メモリ 306 の ROM 349 に格納される。ソフトウェアを格納する ROM 349 などのハードウェアデバイスは、ファームウェアと称される場合がある。POST プログラム 350 は、コンピュータモジュール 301 内のハードウェアを検査して、適切に機能することを保証し、典型的には、プロセッサ 305、メモリ 334（309、306）、および基本入出力システムソフトウェア（BIOS）モジュール 351 をチェックし、さらに典型的には、正確に動作するために、ROM 349 に格納される。POST プログラム 350 が良好に実行されると、BIOS 351 は、図 3 A のハードディスクドライブ 310 を起動する。ハードディスクドライブ 310 を起動することにより、ハードディスクドライブ 310 上にあるブートストラップ・ローダ・プログラム 352 が、プロセッサ 305 を介して実行される。これにより、オペレーティングシステム 353 が、RAM メモリ 306 にロードされ、オペレーティングシステム 353 は、動作を開始する。オペレーティングシステム 353 は、プロセッサ 305 により実行可能なシステム・レベル・アプリケーションであり、さまざまな高レベル機能を実行し、高レベル機能には、プロセッサ管理、メモリ管理、デバイス管理、記憶管理、ソフトウェア・アプリケーション・インターフェース、および汎用ユーザインターフェースがある。

20

#### 【0047】

オペレーティングシステム 353 は、メモリ 334（309、306）を管理して、コンピュータモジュール 301 で実行する各処理またはアプリケーションを、他の処理に割り当てられたメモリと衝突することなく実行するのに十分なメモリを確保することを保証する。さらに、図 3 A のシステム 300 で利用可能な異なる種類のメモリを適切に使用して、各処理を効果的に実行することを可能にしなければならない。したがって、集合的なメモリ 334 は、（特に言及しない限り）メモリの特定のセグメントを割り当てる方法を示すことは意図しないが、コンピュータシステム 300 によってアクセス可能なメモリの概観、およびそれらを使用する方法を示すことを意図する。

30

#### 【0048】

図 3 B に示すように、プロセッサ 305 は、制御ユニット 339、算術論理ユニット（ALU）340、および、キャッシュメモリと呼ばれることもある、ローカルまたは内部メモリ 348 を含む多数の機能モジュールを含む。典型的には、キャッシュメモリ 348 は、レジスタ部分に、多数の記憶レジスタ 344 から 346 を備える。1 つまたは複数の内部バス 341 は、これらの機能モジュールを、機能的に相互接続する。典型的には、プロセッサ 305 はまた、1 つまたは複数のインターフェース 342 を有し、システムバス 304 を介して、接続部 318 を用いて、外部デバイスと通信する。メモリ 334 は、接続部 319 を用いて、バス 304 に結合される。

40

#### 【0049】

アプリケーションプログラム 333 は、条件分岐命令およびループ命令を含むことができる一連の命令 331 を備える。プログラム 333 はまた、プログラム 333 の実行時に使用するデータ 332 を含んでもよい。命令 331 およびデータ 332 は、それぞれ、メモ

50

リ領域 3 2 8、3 2 9、3 3 0 および 3 3 5、3 3 6、3 3 7 に格納される。命令 3 3 1 およびメモリ領域 3 2 8 から 3 3 0 の相対的なサイズに応じて、特定の命令が、メモリ領域 3 3 0 で示される命令によって表される単一のメモリ領域に格納され得る。その代わりに、命令は、メモリ領域 3 2 8 および 3 2 9 で示される命令セグメントによって表されるように、それぞれが別々のメモリ領域に格納される多数の部分にセグメンテーションされてもよい。

#### 【 0 0 5 0 】

通常、プロセッサ 3 0 5 は、そこで実行される命令のセットを与えられる。プロセッサ 3 0 5 は、次の入力待ち、プロセッサ 3 0 5 は、他のセットの命令を実行することによって対応する。各入力は、多数のソースのうちの 1 つまたは複数からもたらすことができ、  
入力デバイス 3 0 2、3 0 3 の 1 つもしくは複数によって生成されたデータ、ネットワーク 3 2 0、3 0 2 のうち 1 つの外部ソースから受信したデータ、記憶デバイス 3 0 6、3 0 9 の 1 つから検索されたデータ、または対応するリーダ 3 1 2 に挿入された記憶媒体 3 2 5 から検索されたデータを含み、すべて図 3 A に示される。命令のセットを実行することで、場合によっては、データを出力する可能性がある。命令の実行はまた、メモリ 3 3 4 へのデータまたは変数の格納を含んでもよい。

#### 【 0 0 5 1 】

符号器 1 0 0、復号器 2 0 0、および説明した方法は、入力変数 3 5 4 を使用する。入力変数 3 5 4 は、メモリ 3 3 4 内の対応するメモリ領域 3 5 5、3 5 6、3 5 7 に格納される。符号器 1 0 0、復号器 2 0 0、および説明した方法は、出力変数 3 6 1 を生成する。  
出力変数 3 6 1 は、メモリ 3 3 4 内の対応するメモリ領域 3 6 2、3 6 3、3 6 4 に格納される。中間変数 3 5 8 は、メモリ領域 3 5 9、3 6 0、3 6 6、および 3 6 7 に格納することができる。

#### 【 0 0 5 2 】

図 3 B のプロセッサ 3 0 5 において、レジスタ 3 4 4、3 4 5、3 4 6、算術論理ユニット (A L U) 3 4 0、および制御ユニット 3 3 9 が協働し、プログラム 3 3 3 を構成する命令セット内のすべての命令に対する「取り出し、復号化、および実行」サイクルを行うのに必要な、一連のマイクロ操作を実行する。各取り出し、復号化、および実行サイクルは、以下を備える。

( a ) 取り出し動作：メモリ領域 3 2 8、3 2 9、3 3 0 から命令 3 3 1 を取り出しまたは読み取り

( b ) 復号化動作：制御ユニット 3 3 9 が、どの命令が取り出されたかを判断する

( c ) 実行動作：制御ユニット 3 3 9 および / または A L U 3 4 0 が命令を実行する

その後、次の命令のためのさらなる取り出し、復号化、および実行サイクルを実行してもよい。同様に、制御ユニット 3 3 9 がメモリ領域 3 3 2 に値を格納または書き込む格納サイクルを実行してもよい。

#### 【 0 0 5 3 】

説明する図 1、図 2、図 4、図 5、図 7 から図 1 0、および図 1 4 から図 1 7 の処理における各ステップまたはサブ処理は、プログラム 3 3 3 の 1 つまたは複数のセグメントと関連づけられ、プログラム 3 3 3 の当該セグメントに対する命令セット内のすべての命令に対して、取り出し、復号化、および実行サイクルを行うために協働するプロセッサ 3 0 5 内のレジスタ部分 3 4 4、3 4 5、3 4 7、A L U 3 4 0、および制御ユニット 3 3 9 によって実行される。

#### 【 0 0 5 4 】

あるいは、符号器 1 0 0、復号器 2 0 0、および説明した方法は、説明した方法の機能またはサブ機能を実行する 1 つまたは複数の集積回路などの専用ハードウェアで実現してもよい。そのような専用ハードウェアは、コンピュータ化された装置の形式としてもよく、グラフィックプロセッサ、デジタル信号プロセッサ、特定用途向け集積回路 (A S I C)、フィールド・プログラマブル・ゲート・アレイ (F P G A)、または 1 つもしくは複数のマイクロプロセッサおよび関連したメモリを含んでもよい。そのようなコンピュータ化

された装置を使用して、ハードウェアにおける何らかのコーディング動作およびハードウェアで実行するソフトウェアにおける他のコーディング動作を実行してもよい。

【0055】

上記のように、動画符号器100は、ハードディスクドライブ305にあり、プロセッサ305により実行中に制御される、ソフトウェア・アプリケーション・プログラム333の1つまたは複数のソフトウェア・コード・モジュールとして実現してもよい。特に、動画符号器100は、ソフトウェア・アプリケーション・プログラム333の1つまたは複数のソフトウェアモジュールとしてそれぞれが実現され得るモジュール102から112、114、および115を備える。

【0056】

動画符号器100は、高効率動画符号化(H E V C)動画復号化パイプラインの一例であるが、モジュール102から112、114、および115によって実行される処理段階は、V C - 1またはH . 2 6 4 / M P E G - 4 A V Cなどの他のビデオコーデックと共通である。動画符号器100は、輝度サンプルおよび色差サンプルを含む一連のフレームとして、符号化されていないフレームデータ101を受信する。動画符号器100は、例えば、フレームデータ101の各フレームを、符号化ユニット(C U)ツリーとして表すことが可能な、階層化された符号化ユニット(C U)のセットに分割する。

【0057】

動画符号器100は、マルチプレクサモジュール110からの、予測ユニット(P U)120として知られる予測されたデータサンプルの配列を受信することによって動作する。差異モジュール115は、予測ユニット(P U)120と、フレームデータ101から受信したデータサンプルの対応する配列との間の差異を出力する。差異は、残差データサンプル122として知られる。

【0058】

差異モジュール115からの残差データサンプル122は、変換モジュール102によって受信される。変換モジュール102は、空間表現から周波数領域表現に差異を変換し、変換ツリーにおける各変換ユニット(T U)に対する変換係数124を生成する。開発中の高効率動画符号化(H E V C)規格の場合、周波数領域表現への変換は、変形離散コサイン変換(D C T)を使用して実現され、従来のD C Tは、シフトおよび付加を用いて実現されるよう変更される。次いで、変換係数124は、スケール変更および量子化モジュール103に入力され、スケール変更ならびに量子化されて、残差係数126を生成する。スケール変更および量子化処理では、精度の損失が生じる。残差係数126は、スケール変更および量子化モジュール103によって実行されるスケーリングを反転する逆スケーリングモジュール105への入力と見なされ、残差係数126の再スケール変更バージョンである、再スケール変更された変換係数128を生成する。残差係数126はまた、符号化ビットストリーム113における残差係数を符号化するエントロピー符号器モジュール104への入力と見なされる。スケール変更および量子化モジュール103に起因する精度の損失のため、再スケール変更された変換係数128は、もとの変換係数124と同一ではない。逆スケーリングモジュール105からの再スケール変更された変換係数128は、次いで、逆変換モジュール106に出力される。逆変換モジュール106は、周波数領域から空間領域への逆変換を行い、復号器で生成される空間領域表現と同一である、再スケール変更された変換係数128の空間領域表現130を生成する。

【0059】

動き推定モジュール107は、フレームデータ101を、典型的にはメモリ306内に構成されるフレーム・バッファ・モジュール112に格納された以前のフレームデータと比較することによって、運動ベクトル132を生成する。次いで、運動ベクトル132は、フレーム・バッファ・モジュール112に格納されるサンプルをフィルタリングし、運動ベクトル132から導出される空間オフセットを考慮に入れることによってインター予測参照サンプル134を生成する、動き補償モジュール108に入力される。図1に示さないが、運動ベクトル132もまた、構文要素として、符号化ビットストリーム113でコ

10

20

30

40

50

ード化するために、エントロピー符号器モジュール104に通される。イントラフレーム予測モジュール109は、加算モジュール114から取得されるサンプル138を使用して、イントラ予測参照サンプル136を生成する。加算モジュール114は、マルチプレクサモジュール110の出力120と、逆変換モジュール106からの出力130とを合計する。

#### 【0060】

予測ユニット(PU)は、イントラ予測方法またはインター予測方法を使用してコード化してもよい。イントラ予測を使用するか、インター予測を使用するかの判断は、結果としての符号化ビットストリーム113の所望のビットレートと、イントラ予測方法またはインター予測方法により発生する画質劣化の量との間のレート-劣化トレードオフによる。マルチプレクサモジュール110は、現在の予測モード142により、図示しないが、当分野で既知である制御ロジックによって決定される、イントラフレーム予測モジュール109からのイントラ予測参照サンプル136、または動き補償ブロック108からのインター予測参照サンプル134のいずれかを選択する。予測モード142はまた、図示したように、エントロピー符号器104にもたらされ、説明するように、変換ユニットのスキャン順序を決定するか、または確立するために使用される。インターフレーム予測は、斜め方向スキャン順序のみを使用し、一方、イントラフレーム予測は、斜め方向スキャン順序、水平方向スキャン順序、または垂直方向スキャン順序を使用することができる。

#### 【0061】

加算モジュール114は、デブロッキング・フィルタ・モジュール111への入力である合計138を生成する。デブロッキング・フィルタ・モジュール111は、ブロック境界に沿ってフィルタリングを行い、メモリ306内に構成されるフレーム・バッファ・モジュール112に書き込まれるデブロック化サンプル140を生成する。フレーム・バッファ・モジュール112は、後に参照するための複数の過去フレームからのデータを保持するのに十分な容量を有するバッファである。

#### 【0062】

動画符号器100では、1つの変換ユニット(TU)内の残差データサンプル122が、入力フレームデータ101のデータサンプルと、入力フレームデータ101のデータサンプルの予測120との間の差異を発見することによって決定される。差異は、変換ユニット(TU)の残差係数の空間表現をもたらす。

#### 【0063】

エントロピー符号器モジュール104の動作中、変換ユニット(TU)の残差係数は、2次元有意度マップに変換される。変換ユニット(TU)における残差係数の有意度マップは、次いで、スキャン順序として知られている特定の順序でスキャンされ、有意係数フラグのリストと呼ばれる、フラグ値の1次元リストを形成する。スキャン順序は記述されるか、他にスキャンパターンによって指定すれば良い。例えば、イントラ予測モジュール109から予測モード142と共に受け取る。イントラ予測モジュール109は、スキャンパターンを選択するために使用される可能性があるイントラ予測モードを決定する。例えば、イントラ予測モード1(垂直イントラ予測)が選択された場合、水平方向スキャンが、図12に示すように使用される。イントラ予測モード0(平面イントラ予測)が選択された場合、斜め方向スキャンが、図11に示すように使用され、一方、イントラ予測モード2(水平イントラ予測)が選択された場合、垂直方向スキャンが、図13に示すように使用される。スキャンパターンは、水平、垂直、斜め、またはジグザグとすることができる。高効率動画符号化(HEVC)テストモデルのバージョン5では、逆方向へのスキャンを行うが、順方向へのスキャンも可能である。16×16、32×32、4×16、16×4、8×32、および32×8変換ユニット(TU)の場合、2レベルスキャンが定義され、変換ユニット(TU)が、サブブロックのセットに分割され、各サブブロックは、正方形である。上位では、スキャンは、後方斜め左下方向スキャンなどのスキャンを使用して、各下位をスキャンすることによって行われる。サブブロックレベルとしても知られる下位では、スキャンはまた、後方斜め左下方向スキャン

ンなどのスキャンを使用して行われる。H E V C 参照モデルバージョン 5 . 0 では、スキャン動作は、最終有意係数後の 1 残差係数で開始し（「後の」は、残差係数の後方スキャンの方向である）、有意度マップの左上位置に達するまで進行する。この特性を有し、H E V C 参照モデルバージョン 5 . 0 に従うスキャン動作は、「後方スキャン」として知られる。H E V C 参照ソフトウェアバージョン 5 . 0 では最終有意係数の位置は、変換ユニット（T U）における係数の座標を符号化することによって示される。当業者には、本文における形容詞「最終」の使用は、スキャンの特定順序に依存することが認識されるだろう。あるスキャンパターンによる、「最終」非ゼロ残差係数または対応する 1 値有意係数フラグである可能性があるものは、他のスキャンパターンによる「最終」ではないかも知れない。有意係数フラグのリストは、最終有意係数の前の各残差係数の有意性を示し、ビットストリーム 1 1 3 内にコード化される。最終有意係数フラグ値は、最終有意係数フラグの位置より前のコード化により、残差係数に有意であることが暗黙的に示されたため、ビットストリーム 1 1 3 内に明示的に符号化する必要はない。

10

#### 【 0 0 6 4 】

変換ユニット（T U）の左上に向けての、より大きな値の残差係数のクラスタリングは、その有意度のリストにおいて、より早くに、最有意フラグをもたらし、一方、有意なフラグは、リストの後方では、ほとんど見つからない。

#### 【 0 0 6 5 】

エントロピー符号器モジュール 1 0 4 はまた、スケールおよび量子化モジュール 1 0 3 から受信した入力残差係数データ（または、残差係数）1 2 6 から構文要素も生成する。エントロピー符号器モジュール 1 0 4 は、符号化ビットストリーム 1 1 3 を出力し、これを以下により詳細に説明する。開発中の高効率動画符号化（H E V C）規格の場合、符号化ビットストリーム 1 1 3 は、ネットワーク抽象化レイヤ（N A L）ユニットに描画される。フレームの各スライスは、1 つの N A L ユニットに格納される。

20

#### 【 0 0 6 6 】

エントロピー符号器モジュール 1 0 4 で実現されるエントロピー符号化方法にはいくつかの代替法がある。開発中の高効率動画符号化（H E V C）規格は、コンテキスト適応型 2 進算術符号化（C A B A C）をサポートし、コンテキスト適応型 2 進算術符号化（C A B A C）の変形は、H . 2 6 4 / M P E G - 4 A V C で見られる。代替のエントロピー符号化方式は、当分野で既知である、確率区間区分エントロピー（P I P E : p r o b a b i l i t y i n t e r v a l p a r t i t i o n i n g e n t r o p y）コードである。

30

#### 【 0 0 6 7 】

複数の動画符号化方法をサポートする動画符号器 1 0 0 に対し、サポートされるエントロピーコーディング方法の 1 つは、符号器 1 0 0 の構成により選択される。さらに、各フレームからの符号化ユニットを符号化する際、エントロピー符号器モジュール 1 0 4 は、各フレームが、フレーム毎に 1 つまたは複数のスライスを有するように符号化ビットストリーム 1 1 3 を書き込み、各スライスは、フレームの一部に対する画像データを含む。フレーム毎に 1 つのスライスを生成することで、各スライスの境界を描画することと関連したオーバーヘッドを低減する。しかしながら、フレームを複数のスライスに分割することも可能である。

40

#### 【 0 0 6 8 】

動画復号器 2 0 0 は、ハードディスクドライブ 3 0 5 にあり、プロセッサ 3 0 5 により実行中に制御される、ソフトウェア・アプリケーション・プログラム 3 3 3 の 1 つまたは複数のソフトウェア・コード・モジュールとして実現してもよい。特に、動画復号器 2 0 0 は、ソフトウェア・アプリケーション・プログラム 3 3 3 の 1 つまたは複数のソフトウェア・コード・モジュールとしてそれぞれが実現され得るモジュール 2 0 2 から 2 0 8、および 2 1 0 を備える。動画復号器 2 0 0 は、高効率動画符号化（H E V C）動画復号化パイプラインを参照して説明するが、モジュール 2 0 2 から 2 0 8、および 2 0 9 によって実行される処理段階は、H . 2 6 4 / M P E G - 4 A V C、M P E G - 2、および V C

50

- 1などのエントロピーコーディングを使用する他のビデオコーデックと共通である。

【0069】

符号化ビットストリーム113などの符号化ビットストリームは、動画復号器200によって受信される。符号化ビットストリーム113は、メモリ306、ハードディスクドライブ310、CD-ROM、Blu-ray（登録商標）ディスク、または他のコンピュータ可読記憶媒体から読み込むことができる。あるいは、符号化ビットストリーム113は、通信ネットワーク320に接続されたサーバまたは無線周波数受信器などの外部ソースから受信してもよい。符号化ビットストリーム113は、復号化すべきフレームデータを示す符号化構文要素を含む。

【0070】

符号化ビットストリーム113は、エントロピー復号器モジュール202に入力される。エントロピー復号器モジュール202は、符号化ビットストリーム113から構文要素を抽出し、その構文要素の値を、動画復号器200内の他のブロックに通す。エントロピー復号器モジュール202で実現されるエントロピー復号化方法は、エントロピー符号器モジュール104を参照して説明したものなど、複数あってもよい。残差係数データを表す構文要素データ220は、逆スケールおよび変換モジュール203に通され、運動ベクトル情報を表す構文要素データ222は、動き補償モジュール204に通される。逆スケールおよび変換モジュール203は、残差係数データについて逆スケーリングを実行し、再構築された変換係数を生成する。次いで、モジュール203は、逆変換を行い、再構築された変換係数を、逆変換モジュール106を参照して説明した逆変換のように、周波数領域表現から空間領域表現に変換し、残差サンプル224を生成する。

【0071】

動き補償モジュール204は、フレーム・バッファ・ブロック208からの以前のフレームデータ226と結合され、メモリ306内で構成される、エントロピー復号器モジュール202からの運動ベクトルデータ222を使用し、出力復号化フレームデータの予測である、予測ユニット(PU)に対するインター予測参照サンプル228を生成する。現在の符号化ユニットがイントラ予測を使用してコード化されたことを構文要素が示す場合、イントラフレーム予測モジュール205は、予測ユニット(PU)と空間的に隣接するサンプルを使用して、予測ユニット(PU)に対し、イントラ予測参照サンプル230を生成する。空間的に隣接するサンプルは、加算モジュール210から出力された合計232から取得される。マルチプレクサモジュール206は、符号化ビットストリーム113における構文要素によって示される現在の予測モードにより、予測ユニット(PU)に対し、イントラ予測参照サンプルまたはインター予測参照サンプルを選択する。マルチプレクサモジュール206から出力されたサンプル234の配列は、加算モジュール210によって、逆スケールおよび変換モジュール203からの残差サンプル224に加えられ、合計232を生成する。合計232は、次いで、デブロッキング・フィルタ・モジュール207とイントラフレーム予測モジュール205とのそれぞれに入力される。符号器100とは対照的に、イントラフレーム予測モジュール205は、エントロピー復号器202から予測モード236を受信する。マルチプレクサ206は、イントラフレーム予測/インターフレーム予測選択信号を、エントロピー復号器202から受信する。デブロッキング・フィルタ・モジュール207は、データブロック境界に沿ってフィルタリングを行い、データブロック境界に沿って視認可能なアーチファクトを平滑化する。デブロッキング・フィルタ・モジュール207の出力は、メモリ306内に構成されたフレーム・バッファ・モジュール208に書き込まれる。フレーム・バッファ・モジュール208は、今後参照するための複数の復号化フレームを保持するのに十分な記憶領域を提供する。復号化フレーム209はまた、フレーム・バッファ・モジュール208から出力される。

【0072】

エントロピー符号器104は、図4を参照して説明する。残差係数401などの構文要素は、バイナライザモジュール404に入力される。変換ユニット(TU)サイズ402は、バイナライザモジュール404に入力され、符号化される変換ユニット(TU)のサイ

10

20

30

40

50

ズを指示する。スキャンパターン 403 は、バイナライザモジュール 404 に入力される。バイナライザモジュール 404 は、各構文要素を、一連のピンにバイナライズする。各ピンは、ピン値 406 およびコンテキストインデックス 405 を備える。ピン値 406 およびコンテキストインデックス 405 は、コンテキストモデル 407 によって受信され、コンテキストモデル 407 は、コンテキスト 408 を出力し、コンテキストインデックス 405 に従って選択される。コンテキスト 408 は、ピン値 405 により更新される。コンテキスト 408 を更新する方法は、H.264/MPEG-4 AVC においてコンテキスト適応型 2 進算術符号化 (CABAC) によって使用される方法と一致する。バイナライザモジュール 404 は、以下の図 6 を参照して説明する方法 600 により、残差係数をバイナライズする。二値算術符号器 409 は、コンテキスト 408 およびピン値 406

10

#### 【0073】

エントロピー復号器 202 は、図 5 を参照して説明する。変換ユニット (TU) サイズ 502 およびスキャンパターン 501 は、逆バイナライザモジュール 503 によって受信される。スキャンパターン 501 は、エントロピー復号器 202 によって決定される予測モード 236 から決定してもよい。このことは、符号器 100 のイントラ予測モジュール 109 に関して上記したように、予測モード 236 に基づいてスキャンパターンを選択することによって行うことができる。逆バイナライザモジュール 503 は、バイナライザモジュール 404 の反転動作を実行することによって、残差係数 509 を出力する。コンテキストインデックス 504 は、復号化すべき各ピンに対し、逆バイナライザモジュール 503 から出力される。コンテキストモデル 505 は、コンテキストインデックス 504 によって選択されたコンテキスト 506 を出力する。二値算術復号器 507 は、コンテキスト 506 を使用して、符号化ビットストリーム 113 からのピン値 508 を復号化する。ピン値 508 は、コンテキストモデル 505 によって受信され、コンテキスト 506 を更新するために使用される。ピン値 508 はまた、逆バイナライザモジュール 503 によって受信される。逆バイナライザモジュール 503 は、以下の図 7 を参照して説明する方法 700 により、残差係数を復号化する。

20

#### 従来の 8×8 TU 符号化

後方斜め方向スキャンを使用して 8×8 変換ユニット (TU) 1100 を符号化する従来の方法は、図 6 および図 11 を参照して説明する。図 11 における 8×8 変換ユニット (TU) 1100 は、サブブロックを利用しない。後方斜め左下方向スキャン 1101 は、8×8 変換ユニット (TU) 1100 の全体にわたって適用される。図 6 の方法 600 は、後方斜め左下方向スキャン 1101 スキャンパターンを使用して適用される。

30

#### 【0074】

最終有意係数ポジションを符号化するステップ 601 は、8×8 変換ユニット (TU) 1100 における最終有意係数のポジションを符号化する。最終有意係数のポジションは、順方向に後方斜め左下方向スキャン 1101 を適用することによって 8×8 変換ユニット (TU) 1100 を検索する場合、最終非ゼロ残差係数のポジションとして定義される。

#### 【0075】

8×8 テーブルを参照するステップ 602、有意係数フラグを符号化するステップ 603、およびサブセット内の最後であるかを判断するステップ 604 で、16 個の係数のサブセットにおける有意係数をすべて符号化する。ステップ 603 および 604 は、後方スキャンが用いられた場合、最終有意係数を含有するサブセットについて、第 1 に動作する。8×8 テーブルを参照するステップ 602 は、8×8 後方斜め左下方向スキャン 1101 を保持するテーブルへの現在のサブセットに対するインデックスにおいて判断する。有意係数フラグを符号化するステップ 603 では、二値算術符号器 409 を使用して、符号化ビットストリーム 113 に、サブセット内の 1 つの有意係数を符号化する。サブセット内の最後であるかを判断するステップ 604 は、サブセット内のすべての有意係数がスキャンされた場合、ステップ 602 および 603 を終了する。後方スキャンの場合、このループは、最大値からゼロにカウントダウンする。最大値は 16 であり、最終有意係数を含有

40

50

するサブセットの場合を除き、最大値は、サブセットスキャンからの最終有意係数を省くよう構成される。

#### 【 0 0 7 6 】

残差レベルを符号化するステップ 6 0 5 は、サブセット内の各有意係数に対する残差レベルを符号化する。サブセットにおける残差レベルは、複数のステップで符号化され、各ステップは、サブセットにおける有意係数を繰り返し処理する。第 1 に、有意係数の絶対値が 1 より大きいことを示すフラグが符号化される。第 2 に、有意係数が 2 より大きい絶対値を有する、有意係数が 1 より大きい絶対値を有することを示すフラグが符号化される。第 3 に、各有意係数の符号を示すフラグが符号化される。第 4 に、有意係数が 2 よりも大きい値を有する場合、これらの 3 つの有意係数を引いた強度が符号化される。

10

#### 【 0 0 7 7 】

最終サブセットかを判断するステップ 6 0 6 により、変換ユニット ( T U ) 1 1 0 0 における第 1 のサブセットが符号化されるまで、ステップ 6 0 2、6 0 3、6 0 4、および 6 0 5 が繰り返えされ、サブセット 4、3、および 2 のいずれかが符号化されると、必要に応じ、方法 6 0 0 は終了する。

#### 【 0 0 7 8 】

後方水平方向スキャンを使用して 8 × 8 変換ユニット ( T U ) 1 2 0 0 を符号化する従来の方法は、図 6 および図 1 2 を参照して説明する。8 × 8 変換ユニット ( T U ) 1 2 0 0 は、サブブロックを利用しない。後方水平方向スキャン 1 2 0 1 は、8 × 8 変換ユニット ( T U ) 1 2 0 0 の全体にわたって適用される。方法 6 0 0 は、定義されたスキャンパターンを使用して適用される。

20

#### 【 0 0 7 9 】

後方垂直方向スキャンを使用して 8 × 8 変換ユニット ( T U ) 1 3 0 0 を符号化する従来の方法は、図 6 および図 1 3 を参照して説明する。8 × 8 変換ユニット ( T U ) 1 3 0 0 は、サブブロックを利用しない。後方垂直方向スキャン 1 3 0 1 は、8 × 8 変換ユニット ( T U ) 1 3 0 0 の全体にわたって適用される。方法 6 0 0 は、定義されたスキャンパターンを使用して適用される。

従来の 8 × 8 T U 復号化

後方斜め方向スキャンを使用して 8 × 8 変換ユニット ( T U ) 1 1 0 0 を復号化する従来の方法は、図 7 および図 1 1 を参照して説明する。8 × 8 変換ユニット ( T U ) 1 1 0 0 は、サブブロックを利用しない。後方斜め左下方向スキャン 1 1 0 1 は、8 × 8 変換ユニット ( T U ) 1 1 0 0 の全体にわたって適用される。方法 7 0 0 は、定義されたスキャンパターンを使用して適用される。

30

#### 【 0 0 8 0 】

最終有意係数ポジションを復号化するステップ 7 0 1 は、8 × 8 変換ユニット ( T U ) 1 1 0 0 における最終有意係数の位置を復号化し、スキャンパターンが順方向に適用された場合、最終非ゼロ残差係数として定義される。

#### 【 0 0 8 1 】

8 × 8 ポジションテーブルを参照するステップ 7 0 2、有意係数を復号化するステップ 7 0 3、およびサブセット内の最後であることをテストするステップ 7 0 3 は、サブセット内の各位置を繰り返し処理することによって、最終位置から第 1 の位置へ、サブセット内の有意係数フラグを復号化し、位置毎に 1 つの有意係数フラグを復号化する。8 × 8 ポジションテーブルを参照するステップ 7 0 2 では、サブセット内の現在の有意係数に対し、8 × 8 変換ユニット ( T U ) 1 1 0 0 内の位置を決定する。最終有意係数を含むサブセットの場合、最終位置は、最終有意係数ポジションがサブセットスキャンから除かれるか、そうでなければ、最終ポジションが 1 6 に設定されるよう定義される。

40

#### 【 0 0 8 2 】

残差レベルを復号化するステップ 7 0 5 では、サブセット内の各有意係数に対する強度および符号を復号化する。残差レベルを復号化するステップ 7 0 5 では、サブセット内の有意係数を 4 回繰り返し処理することによって、有意係数の残差レベルを復号化する。第 1

50

の繰り返しでは、有意係数の絶対値が1より大きいかどうかを示すフラグが復号化される。第2の繰り返しでは、1より大きい有意係数の絶対値が、また、2より大きいことを示すフラグが復号化される。第3の繰り返しでは、各有意係数に対する符号を示すフラグが復号化される。第4の繰り返しでは、2より大きい有意係数に対して、3を引いた強度が符号化され、強度が2より大きいことが既知である場合について、残差強度の再構成を可能にする。

#### 【0083】

最終サブセットかをテストするステップ706により、変換ユニット(TU)1100における第1のサブセットが復号化されるまで、ステップ702、703、704、および705が繰り返され、必要に応じ、サブセット4、3、および2のいずれかが符号化され、方法700は終了する。

10

#### 【0084】

後方水平方向スキャンを使用して8×8変換ユニット(TU)1200を復号化する従来の方法は、図7および図12を参照して説明する。8×8変換ユニット(TU)1200は、サブブロックを利用しない。後方水平方向スキャン1201は、8×8変換ユニット(TU)1200の全体にわたって適用される。方法700は、定義されたスキャンパターンを使用して適用される。

#### 【0085】

後方垂直方向スキャンを使用して8×8変換ユニット(TU)1300を復号化する従来の方法は、図7および図13を参照して説明する。8×8変換ユニット(TU)1300は、サブブロックを利用しない。後方垂直方向スキャン1301は、8×8変換ユニット(TU)1300の全体にわたって適用される。方法700は、定義されたスキャンパターンを使用して適用される。

20

実施形態 - 8×8 TUを符号化

本開示による、後方斜め左下方向スキャンを使用して8×8変換ユニット(TU)1500を符号化する方法800は、図8、図10A、図10B、図14、図15A、図15B、および図15Cを参照して説明する。図15Aに示す8×8変換ユニット(TU)1500は、4×4のサブブロックサイズを有する。変換ユニット(TU)1500は、符号化するためにスキャンすべき係数の正方下位レイヤを表す。図15Cに示すサブブロック1502などの各サブブロック内では、図15Cに示すように、後方斜め左下方向スキャン1505を適用する。8×8変換ユニット(TU)1500におけるサブブロックをスキャンする場合、図15Bに示すように、2×2後方斜め左下方向上位レイヤスキャン1504が、正方上位レイヤ1503表現に適用される。図15Bから分かるとおり、上位レイヤ1503は、下位レイヤサブブロック1502の2×2配列を含む。この方式では、有意係数グループフラグは、非重複領域を表し、サブブロックは、下位レイヤにおいて重複しない。したがって、図15Aのスキャンパターン1501は、図15Bのパターン1504を4回複製した図15Cのパターン1505の組み合わせを表す。この手法の利点の1つは、図11、図12、および図13における64(8×8)サイズ配列に対するパターンを格納する代わりに、パターン1501を4(2×2)+16(4×4)=20サイズ配列を使用して格納することができ、メモリの使用を大きく抑えることを可能にすることである。さらに、例えば、配列サイズ2×2、4×4、および8×8のいずれも正方形であることが理解されよう。

30

40

#### 【0086】

方法800は、定義されたサブブロックサイズ、上位レイヤサイズ、およびスキャンパターンを使用して適用される。方法800は、有意係数グループを使用して、変換ユニット(TU)の残差係数を符号化する。方法800は、図8を参照して説明する。変換ユニット(TU)は、多くのサブブロックに分割される。図10Aは、変換ユニット(TU)の例示的上位正方レイヤ1000を示す。

#### 【0087】

図14に示す8×8変換ユニット(TU)1400は、図15Aの8×8変換ユニット(TU)1500の

50

TU) 1500 の一例であり、2 レイヤ (上位および下位) 階層における残差係数を符号化する。8 × 8 変換ユニット (TU) 1400 は、4 × 4 サブブロック 1401 などの、等しいサイズのサブブロックに分割される。変換ユニット (TU) 1400 におけるサブブロックは、上位レイヤ 1402 によって表される。上位レイヤ 1402 は、有意係数グループフラグ 1403 などの、有意係数グループフラグを含み、そのそれぞれは、算出または推測することができる。有意係数グループフラグ 1403 などの有意係数グループフラグが算出される場合、有意係数グループフラグは、サブブロック 1401 などの対応するサブブロック内の、残差係数 1404 などの残差係数のいずれが有意であることを示す。有意係数グループフラグが推測される場合、対応するサブブロック内のすべての残差係数が有意ではないとすることが可能である。

10

#### 【0088】

図 8 に示す方法 800 では、最終有意係数ポジションを符号化するステップ 801 で、2 レベルスキャンパスに沿って、最終有意係数の座標を符号化する。2 × 2 テーブルポジションを参照するステップ 802 では、2 × 2 後方斜め左下方向上位レイヤスキャン 1504 に参照を適用することによって、現在サブブロックに対するポジションを決定する。これにより、関連するサブブロック 1502 を正確に識別することをもたらす。4 × 4 テーブルポジションを参照するステップ 803 では、4 × 4 後方斜め左下方向スキャン 1505 に参照を実行することによって、現在の有意係数に対するポジション内のオフセットを決定し、それにより、現在の 4 × 4 サブブロック 1502 内の正確な位置を識別する。有意係数フラグを決定するステップ 804 では、サブブロック 1502 内の、したがって、8 × 8 変換ユニット (TU) 1500 内の、決定したオフセットで、有意 (1 値) として残差係数の値をテストし、残差係数が非ゼロである場合、非有意 (ゼロ値) である。

20

#### 【0089】

サブブロック内の最後であることをテストするステップ 805 により、最終有意係数を含有しないサブブロック内のすべての位置がテストされるようにステップ 803 および 804 が繰り返され、最終有意係数を含有するサブセットに対し、すべての位置が最終有意係数に先行する (ここで、「先行する」とは、順方向のスキャンングの適用を仮定する)。パターン 1505 などの後方スキャンの場合、サブブロック内の最後は、常に、左上位置である。前方スキャンの場合、サブブロック内の最後は、最終有意係数を含有しないサブブロックに対する右下位置であり、前方スキャンの順序で、最終有意係数ポジションの直前の位置である。

30

#### 【0090】

有意係数グループフラグを決定するステップ 806 では、有意係数フラグを決定し、サブブロック内の有意係数フラグのいずれ (すなわち、少なくとも 1 つ) が 1 値であることを示す。最終有意係数を含有するサブブロックの場合、有意係数グループフラグは、1 値であると推測される。変換ユニット (TU) 1500 の左上隅に位置するサブブロックの場合も、有意係数グループフラグが 1 値であると推測される。有意係数グループを符号化するステップ 807 では、サブブロックに含有される有意度マップの部分を判断するための情報を符号化する。第 1 に、推測されなかった有意係数グループフラグは、ビットストリームに符号化される。第 2 に、有意係数グループフラグが 1 値である場合、サブセットに含有される有意度マップの部分が、ビットストリームに符号化される。残差値を符号化するステップ 808 は、有意度マップを 2 × 2 上位レイヤおよび 4 × 4 下位サブブロックレイヤに構成することを除き、残差レベルを符号化するステップ 605 と同様に動作する。最終サブブロックであることをテストするステップ 809 により、最終有意係数を含有するサブブロックから、変換ユニット (TU) 1500 の左上隅に位置するサブブロックへ、サブブロックの繰り返しが可能となる。このサブブロックが符号化されると、方法 800 は終了する。方法 800 は、動画符号器 100 によってサポートされるすべての変換ユニット (TU) において、4 × 4 サブブロックスキャンの使用を可能にすることが、当業者によって理解されよう。

40

#### 【0091】

50

有意係数グループフラグを決定する際の上位レイヤの動作は、図10Aに示す例示的上位レイヤ1000を参照して説明する。例示的上位レイヤ1000は、サブブロック毎に1つのsignificant\_coeff\_group\_flagから成る。例示的上位レイヤ1000における各サブブロックに対し、significant\_coeff\_group\_flagが算出または推測される。例示的上位レイヤ1000における左上サブブロックに対し、significant\_coeff\_group\_flagは常に、1値として推測される。最終有意係数を含有する例示的上位レイヤ1000におけるサブブロックに対しても、significant\_coeff\_group\_flagは1値として推測される。例示的上位レイヤ1000における他の位置に対し、significant\_coeff\_group\_flag X1001などのsignificant\_coeff\_group\_flag A1002などの右隣接部およびsignificant\_coeff\_group\_flag B1003などの下隣接部が両方とも1値である場合のみ、1値であると推測される。右隣接部または下隣接部のsignificant\_coeff\_group\_flagが例示的上位レイヤ1000の外側になる場合、その隣接部に対して0値が推測される。significant\_coeff\_group\_flagが推測されると、符号化ビットストリーム113で符号化されない。significant\_coeff\_group\_flagが推測されなかった場合、対応するサブブロック内のすべてのsignificant\_coeff\_flagが0値であることが既知である場合、0値significant\_coeff\_group\_flagが算出される。あるいは、対応するサブブロック内のsignificant\_coeff\_flagの少なくとも1つが1値である場合、1値significant\_coeff\_group\_flagが算出される。例示的上位レイヤ1000に対するsignificant\_coeff\_group\_flagの算出値は、後方斜め左下方向スキャンを使用して、符号化ビットストリーム113に符号化される。

#### 【0092】

例えば、図10Bを参照すると、上位レイヤ1004では、significant\_coeff\_group\_flag1005などの影付きsignificant\_coeff\_group\_flagが、右隣接部および下隣接部にに基づき、1値であると推測され、したがって、符号化ビットストリーム113で符号化されない。significant\_coeff\_group\_flag1006は、最終有意係数を含有するため、1値と推測される。significant\_coeff\_group\_flag1007などのsignificant\_coeff\_group\_flagは、最終有意係数の後に位置しているため、それらの位置に対しては、いかなる値も推測、算出、または符号化されない。significant\_coeff\_group\_flag1008は、上位レイヤ804の左上に位置し、常に1値として推測される。

#### 【0093】

上位レイヤにおける各significant\_coeff\_group\_flagを符号化すると、コンテキストがコンテキストモデル404から選択される。輝度サンプルを符号化する変換ユニット(TU)および色差サンプルを符号化する変換ユニット(TU)に対し、2つのコンテキストが利用可能であり、例示的上位レイヤ1000を符号化するために、全部で4つのコンテキストが存在する。例示的上位レイヤ1000における各significant\_coeff\_group\_flagに対するコンテキストの選択は、右隣接部および下隣接部に左右される。一例として、significant\_coeff\_group\_flag X1001に対するコンテキストは、significant\_coeff\_group\_flag A1002とsignificant\_coeff\_group\_flag B1003との値の論理ORを行うことによって決定される。例示的上位レイヤ1000の外側にある左隣接部および下隣接部は、コンテキストの選択のために、0値であると推測される。

#### 【0094】

10

20

30

40

50

本開示による、後方水平方向スキャンを使用して  $8 \times 8$  変換ユニット (TU) 1600 を符号化する方法は、図 8、図 16A、図 16B、および図 16C を参照して説明する。 $8 \times 8$  変換ユニット (TU) 1600 は、 $4 \times 4$  の下位サブブロックサイズを有する。サブブロック 1602 などの各サブブロック内では、後方水平方向スキャン 1605 が適用され、図 16A に示す全体的なスキャンパターン 1601 をもたらす。 $8 \times 8$  変換ユニット (TU) 1600 内のサブブロック 1602 をスキャンするために、 $2 \times 2$  後方水平方向上位レイヤスキャン 1604 が適用される。方法 800 は、定義されたサブブロックサイズ、および上位レイヤサイズ、およびスキャンパターンを使用して適用される。

#### 【0095】

本開示による、後方垂直方向スキャンを使用して  $8 \times 8$  変換ユニット (TU) 1700 を符号化する方法は、図 8、図 17A、図 17B、および図 17C を参照して説明する。 $8 \times 8$  変換ユニット (TU) 1700 は、 $4 \times 4$  のサブブロックサイズを有する。図 17C のサブブロック 1702 などの各下位レイヤサブブロック内では、後方垂直方向スキャン 1705 が適用される。 $8 \times 8$  変換ユニット (TU) 1700 におけるサブブロックをスキミングする場合、図 17B に示す  $2 \times 2$  後方垂直方向上位レイヤスキャン 1704 が適用され、図 17A に示す全体的なスキャンパターン 1701 をもたらす。方法 800 は、定義されたサブブロックサイズ、および上位レイヤサイズ、およびスキャンパターンを使用して適用される。

実施形態 -  $8 \times 8$  TU の復号化

本開示による、 $8 \times 8$  変換ユニット (TU) 1500 を復号化する方法は、図 9、図 15A、図 15B、および図 15C を参照して説明する。 $8 \times 8$  変換ユニット (TU) 1500 は、 $4 \times 4$  のサブブロックサイズを有する。サブブロック 1502 などの各サブブロック内では、後方斜め左下方向スキャン 1505 を適用する。 $8 \times 8$  変換ユニット (TU) 1500 におけるサブブロックをスキミングする場合、 $2 \times 2$  後方斜め左下方向スキャン 1504 が適用され、全体的なスキャンパターン 1501 をもたらす。方法 900 は、定義されたサブブロックサイズ、および上位レイヤサイズ、およびスキャンパターンを使用して適用される。

#### 【0096】

有意係数グループを使用して、変換ユニット (TU) の残差係数を復号化する方法 900 は、図 9 を参照して説明する。変換ユニット (TU) は、多くのサブブロックに分割される。最終有意係数ポジションを復号化するステップ 901 では、2 レベルスキャンパス 1501 に沿って、変換ユニット (TU) 1500 における最終有意係数の座標を復号化する。 $2 \times 2$  テーブルポジションを参照するステップ 902 では、 $2 \times 2$  後方斜め左下方向上位レイヤスキャン 1504 に参照を適用することによって、現在サブブロックに対するポジションを決定する。有意係数グループを復号化するステップ 903 では、有意係数グループフラグを決定するステップ 806 で記載したものと同様の処理を使用して、有意係数フラグの推測値を決定するか、または符号化ビットストリーム 113 から 1 つの有意係数グループフラグを復号化する。有意係数グループフラグをテストするステップ 904 では、決定した有意係数グループフラグの値をテストする。フラグが 1 値である場合、制御は  $4 \times 4$  テーブルポジションを参照するステップ 905 に進み、そうでなければ、フラグが 0 値の場合、制御は、サブブロックの最後であるかをテストするステップ 909 に進む。ステップ 905 では、 $4 \times 4$  後方斜め左下方向スキャンに参照を実行することによって、現在の有意係数に対するポジション内のオフセットを決定する。 $4 \times 4$  テーブルポジションを参照するステップ 905 の結果は、 $2 \times 2$  テーブルポジションを参照するステップ 902 の結果と組み合わせられ、 $8 \times 8$  変換ユニット (TU) 1500 にポジションオフセットをもたらす。次いで、有意係数フラグを復号化するステップ 906 で、サブブロック 1502 などのサブブロック内の 1 つの残差係数に対する有意係数フラグを復号化する。サブブロック内の最終有意係数に達し、サブブロック内の以前の有意係数が 1 値ではなく、対応する有意係数グループフラグが算出された（すなわち、推測されなかった）場合、サブブロック内の最終有意係数は、1 値であると推測される。サブブロック内の最後であ

10

20

30

40

50

るかをテストするステップ 907 では、サブブロック内のポジションをテストし、 $4 \times 4$  サブブロック 1502 が、その全体を処理されるまで、ステップ 905 および 906 を繰り返す。処理されるべき第 1 のサブブロックに対し、最終有意係数フラグは、サブブロック内全体で繰り返された後の（「後の」は、後方スキンの順序での定義である）位置のみを含有する。サブブロックの最終ポジションが処理された場合、残差レベルを復号化するステップ 908 では、残差レベルを復号化するステップ 705 と同様の方法で、残差レベルを復号化する。サブブロックの最後であるかをテストするステップ 909 は、変換ユニット（TU）1500 内のすべてのサブブロックにわたって繰り返しもたらされ、最終有意係数を含有するサブブロックから始まり、変換ユニット（TU）1500 の左上隅に位置するサブブロックが、ステップ 902 から 908 によって処理されるまで後方に処理され、その後、方法 900 は終了する。方法 900 は、動画復号器 200 によってサポートされるすべての変換ユニット（TU）サイズにおいて、 $4 \times 4$  サブブロックスキンの使用を可能にすることが、当業者によって理解されよう。

10

#### 【0097】

本開示による、 $8 \times 8$  変換ユニット（TU）1600 を復号化する方法は、図 9、図 16A、図 16B、および図 16C を参照して説明する。 $8 \times 8$  変換ユニット（TU）1600 は、 $4 \times 4$  のサブブロックサイズを有する。サブブロック 1602 などの各サブブロック内では、後方水平方向スキン 1605 を適用する。 $8 \times 8$  変換ユニット（TU）1600 内のサブブロックをスキニングする場合、 $2 \times 2$  後方水平方向スキン 1604 が適用される。方法 900 は、定義されたサブブロックサイズ、および上位レイヤサイズ、およびスキンパターンを使用して適用される。

20

#### 【0098】

本開示による、 $8 \times 8$  変換ユニット（TU）1700 を復号化する方法は、図 9、図 17A、図 17B、および図 17C を参照して説明する。 $8 \times 8$  変換ユニット（TU）1700 は、 $4 \times 4$  のサブブロックサイズを有する。サブブロック 1702 などの各サブブロック内では、後方垂直方向スキン 1705 が適用される。 $8 \times 8$  変換ユニット（TU）1700 内のサブブロックをスキニングする場合、 $2 \times 2$  後方垂直方向スキン 1704 が適用される。方法 900 は、定義されたサブブロックサイズ、および上位レイヤサイズ、およびスキンパターンを使用して適用される。

30

実施形態 -  $8 \times 8$  変換ユニットに対してのみの、斜め方向上位レイヤスキン

方法 800 および 900 が、変換ユニット（TU）1500、1600、および 1700 に適用されたが、それらは、サブブロックレイヤから上位レイヤに同じスキンパターンを適用した。1つの代替方法は、変換ユニット（TU）1500、1600、および 1700 に対する上位レイヤスキンのために、後方斜め左下方向スキンのみを使用することである。さらに、上位レイヤスキンは、すべての変換ユニットサイズに対して、後方斜め左下方向に設定してもよい。その場合、上位レイヤのスキンパターンは、下位レイヤのスキンパターンから独立する。そうすることで、 $2 \times 2$  後方水平方向スキン 1604 および  $2 \times 2$  後方斜め垂直方向スキン 1704 ためのメモリ領域が不要になる。実施形態 -  $32 \times 32$  上位レイヤスキンのための  $8 \times 8$  変換ユニットスキンの再利用  $32 \times 32$  変換ユニットは、 $8 \times 8$  上位レイヤスキンを利用し、後方斜め左下方向スキンを行う。 $2 \times 2$  後方斜め方向スキン 1504 および  $4 \times 4$  後方斜め方向スキン 1505 から派生して、 $8 \times 8$  上位レイヤスキンを  $8 \times 8$  後方斜め方向スキン 1500 と置き換えることが可能である。そのような構成により、 $8 \times 8$  上位レイヤスキンのための参照テーブルを不要にすることを可能にし、必要なメモリ領域を低減する。

40

#### 【0099】

方法 600 および 700 が、特定のサブブロックサイズで適用される場合に、動画符号器 100 または動画復号器 200 内のテーブルの数を減らすことを可能にすることが、当業者には明らかであろう。

実施形態 - オフセットを使用するスキニング

図 18A から図 18C は、線形配列の値が、2次元（2D）配列から導出され、オフセッ

50

トが線形配列で使用され、2D配列の異なるサブブロック間の区別を確立する、さらなる実装形態を示す。図18Aは、8×8変換ユニットが、順方向-下方スキャン（水平、および左から右）で、順方向-下方スキャンに従って、各サブブロックを横断して、A、B、C、およびDの順序で、4つの4×4ユニット（サブブロック）としてアクセスされることを示す。図18Aから図18Cに示した実装形態もまた、逆方向に8×8変換ユニットをスキャンすることが可能であることが、当業者に理解されるであろう。

【0100】

図18Bから分かるとおり、各サブブロックのスキャンは、対応する線形配列の値をもたらす。必要であれば、図18Bの4つの配列は、対応する変換ユニットの値をスキャンするアクセス順序を元から定義して格納してもよい。

10

【0101】

あるいは、図18Cに図示したように、図18Aの変換ユニットのスキャンは、16（4×4）個の値の単一线形配列として格納してもよく、ここでは、オフセット（X）が、4つの配列を再構成するために使用される。Xを0、4、32、および36へ変えることによって、図18Bの4つの配列のそれぞれを、再構成することができる。

総括

ここで開示したさまざまな構成により、上位および下位レベル正方レイヤを有する変換ユニットの符号化および復号化を提供し、ここで、上位レイヤとは、最大4つの有意係数グループフラグの正方配列（例えば2×2）を表す。このことは、例えば、図15B、図16B、および図17Bで示される。これにより、4×4サブブロック（それぞれ、図15C、図16C、および図17C）に適用される所望のスキャンパターンに従って、下位8×8正方レイヤ（それぞれ、図15A、図16A、および図17A）を効果的に横断することを可能にする。したがって、8×8変換ユニットは、4×4下位スキャンおよび2×2上位スキャンの組み合わせを使用して、符号化または復号化することができる。

20

【産業上の利用可能性】

【0102】

説明した構成は、コンピュータおよびデータ処理業に適用可能であり、特に、動画信号などの信号の符号化および復号化のためのデジタル信号処理に対して適用可能である。

【0103】

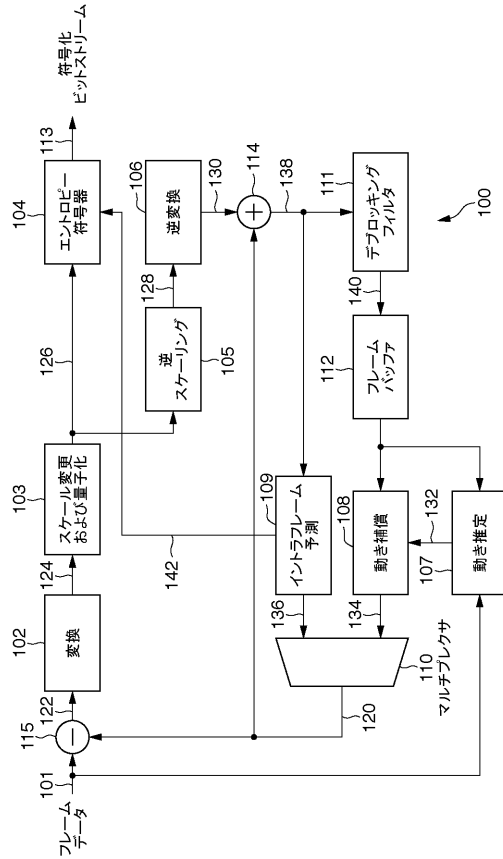
上記は、本発明のいくつかの実施形態のみを説明し、変形例および/または変更例は、本発明の範囲および精神から逸脱することなく行うことができ、本実施形態は、例示目的であって、制限するものではない。

30

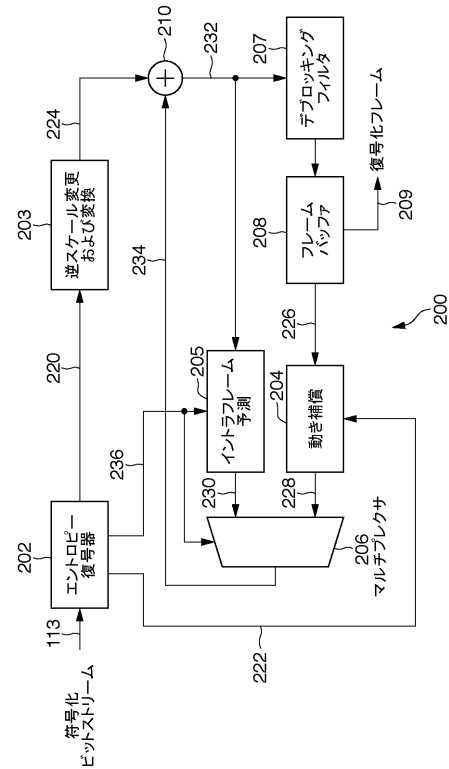
【0104】

（オーストラリアの場合のみ）本明細書の文脈において、「備える」という言葉は、「主に含むが、必ずしも単独ではない」、「有する」、または「含む」を意味し、「それのみから成ること」を意味しない。「備える」という言葉の変化形は、それに対応して意味が変化する。

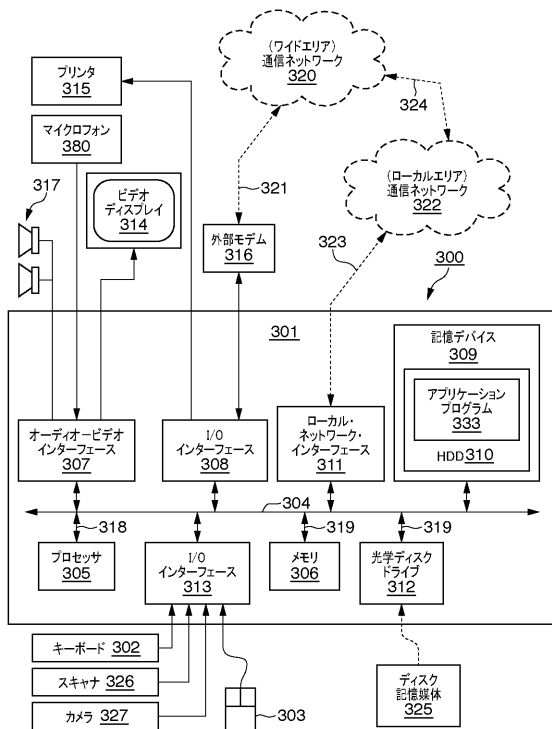
【図 1】



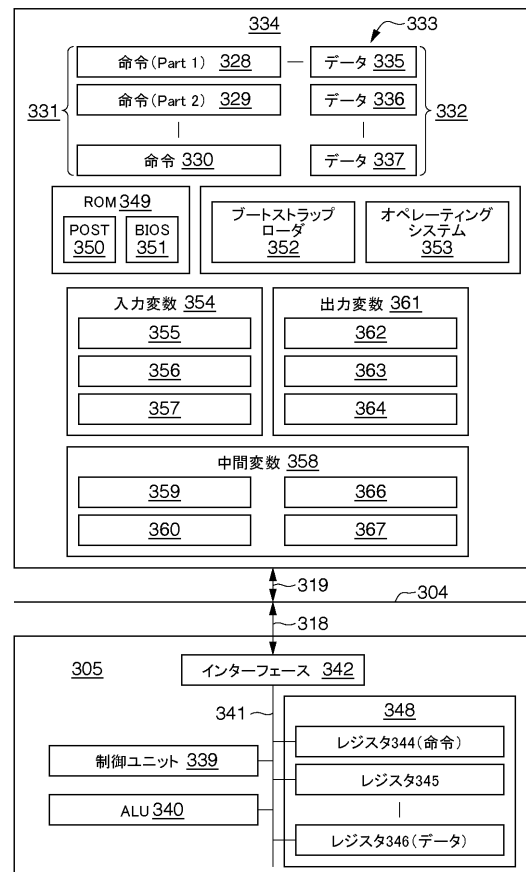
【図 2】



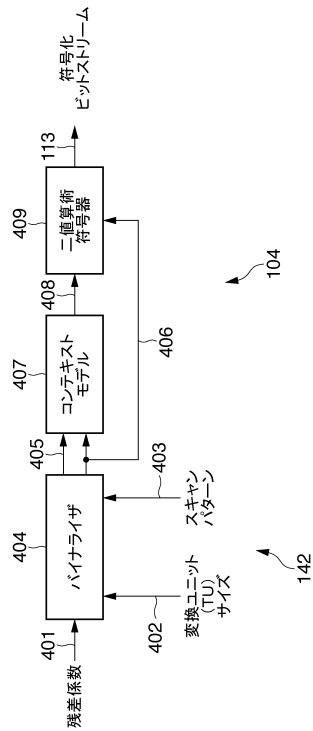
【図 3 ( A )】



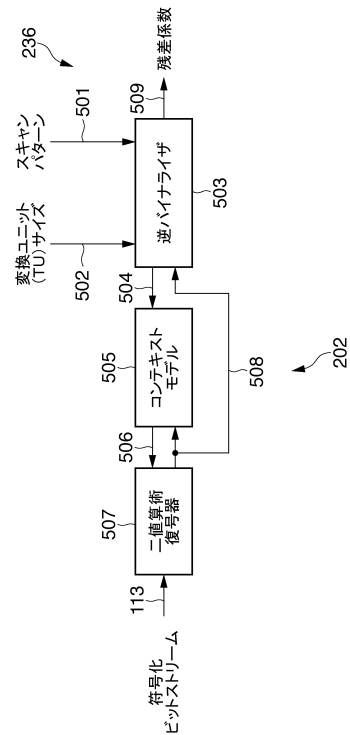
【図 3 ( B )】



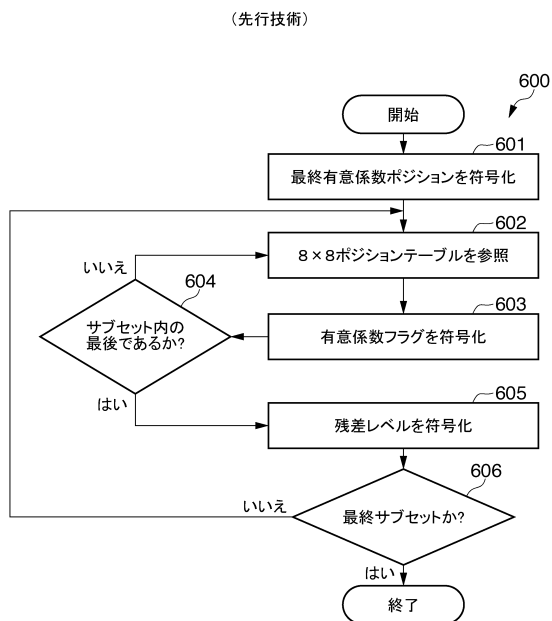
【 図 4 】



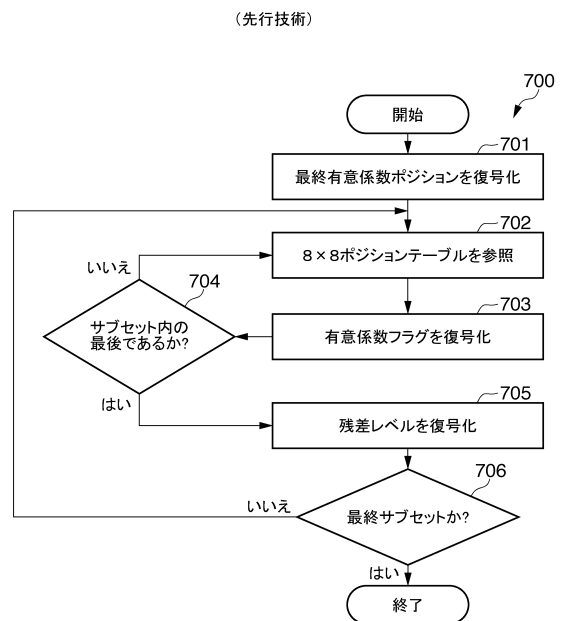
【 図 5 】



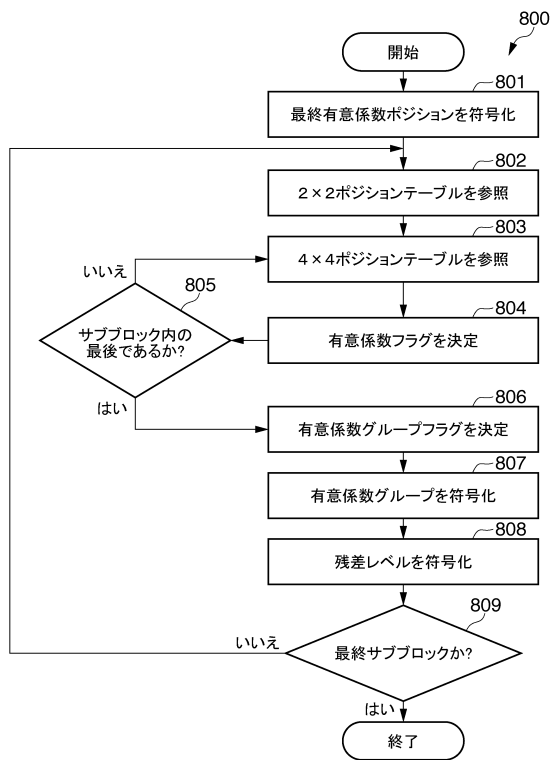
【 図 6 】



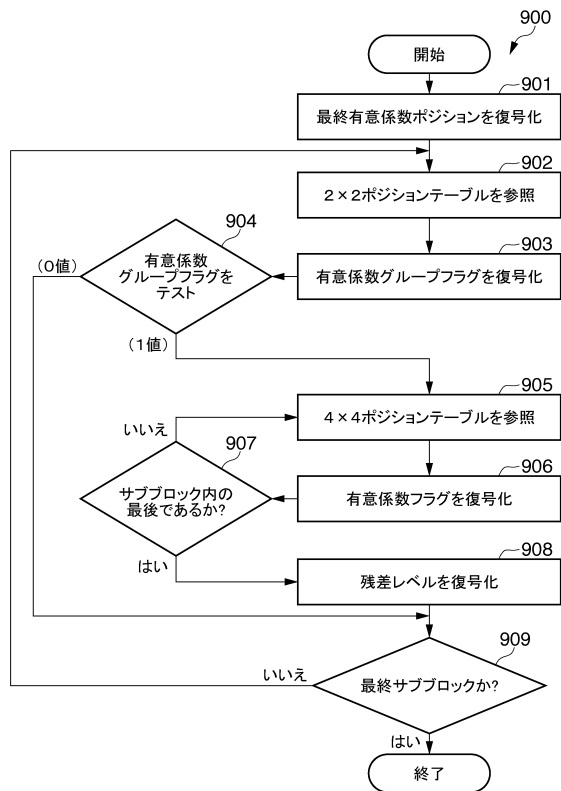
【圖 7】



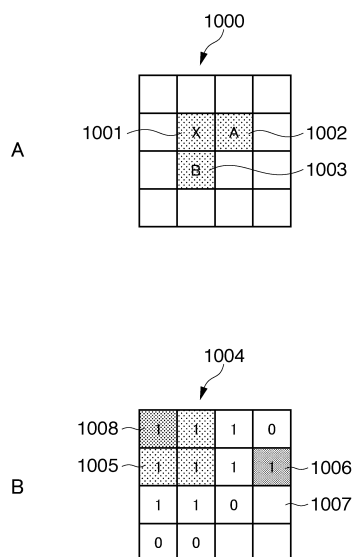
【図 8】



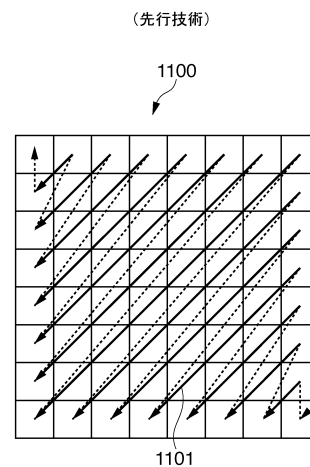
【図 9】



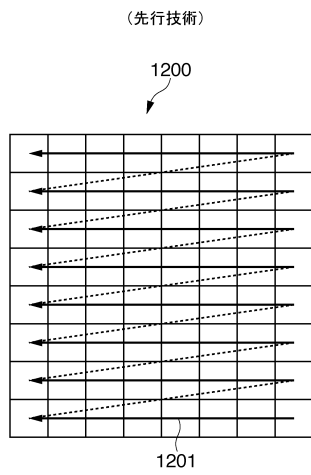
【図 10】



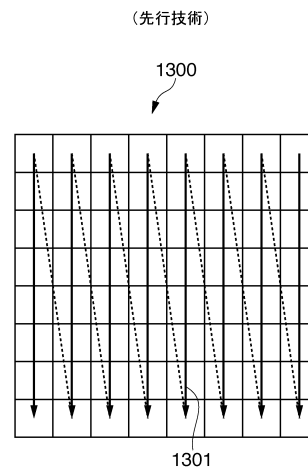
【図 11】



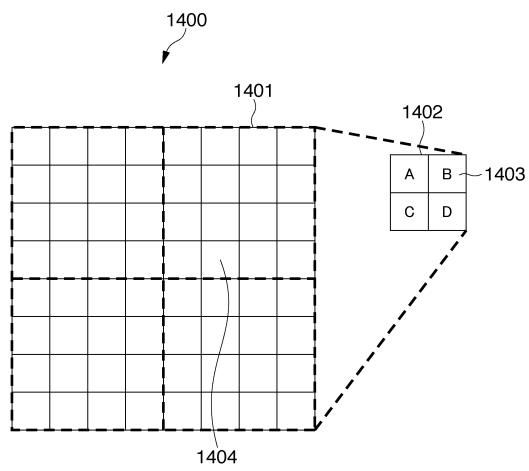
【図 1 2】



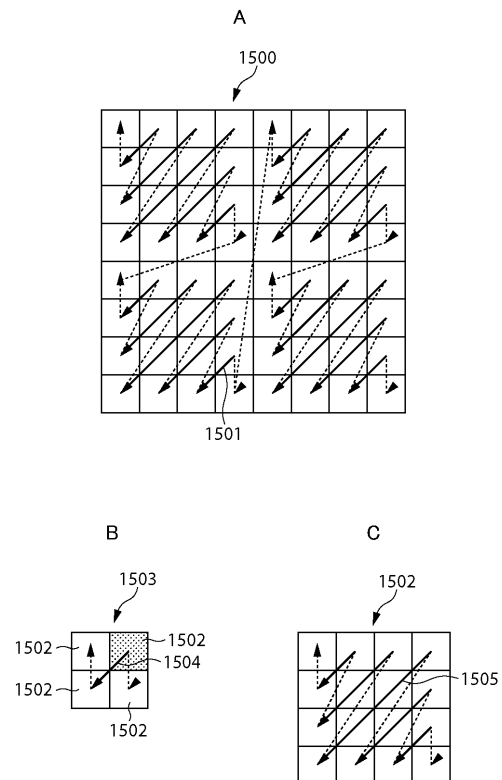
【図 1 3】



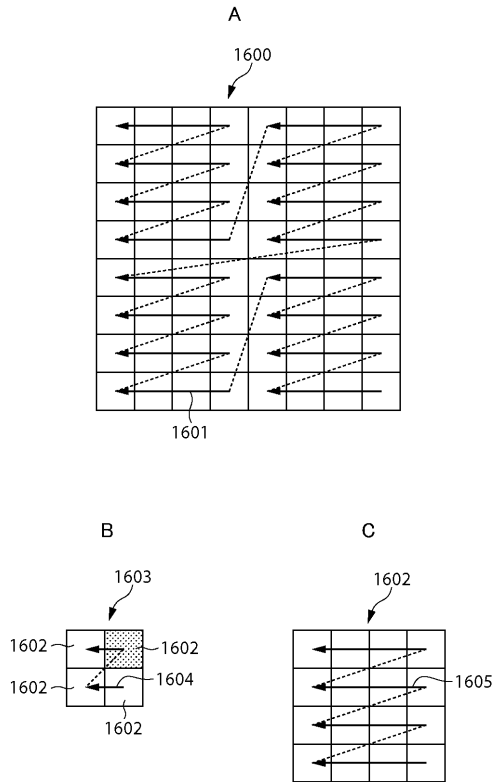
【図 1 4】



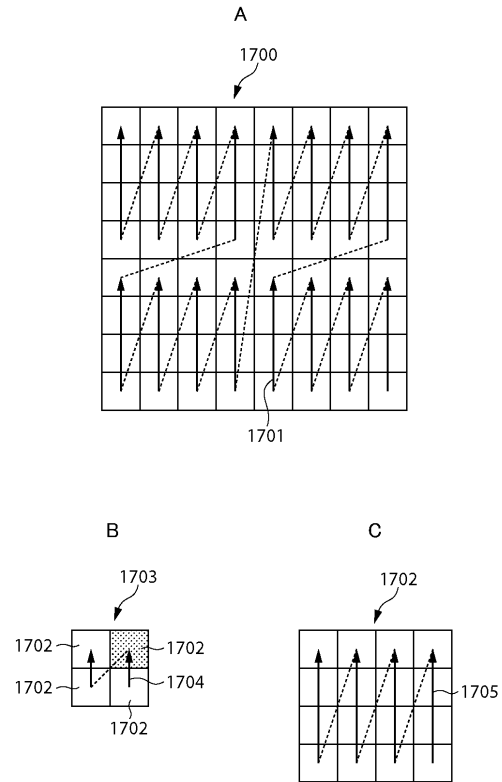
【図 1 5】



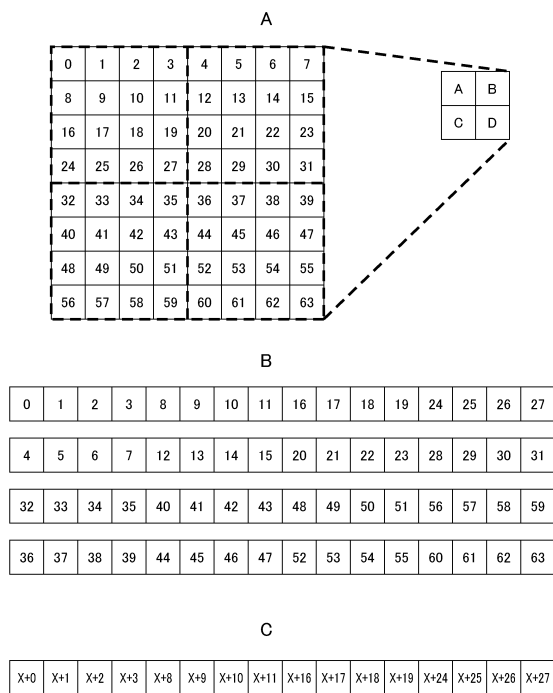
【図 16】



【図 17】



【図 18】



---

フロントページの続き

(72)発明者 アルヴィン エドレア ラビ奥斯  
オーストラリア国 2113 ニューサウスウェールズ州 ノースライド、トーマス ホルト ド  
ライブ 1 キヤノン インフォメーション システムズ リサーチ オーストラリア プロプラ  
イエタリー リミテッド内

審査官 坂東 大五郎

(56)参考文献 国際公開第2013/099892(WO, A1)

(58)調査した分野(Int.Cl., DB名)  
H04N 19/00 - 19/98