

(19) 日本国特許庁(JP)

## (12) 公表特許公報(A)

(11) 特許出願公表番号

特表2017-514353

(P2017-514353A)

(43) 公表日 平成29年6月1日(2017.6.1)

(51) Int.Cl.

**HO4N 19/12 (2014.01)**  
**HO4N 19/154 (2014.01)**  
**HO4N 19/176 (2014.01)**

F 1

HO 4 N 19/12  
HO 4 N 19/154  
HO 4 N 19/176

テーマコード(参考)

5 C 1 5 9

審査請求 未請求 予備審査請求 有 (全 39 頁)

(21) 出願番号 特願2016-557586 (P2016-557586)  
(86) (22) 出願日 平成27年3月12日 (2015.3.12)  
(85) 翻訳文提出日 平成28年11月11日 (2016.11.11)  
(86) 国際出願番号 PCT/US2015/020237  
(87) 国際公開番号 WO2015/179010  
(87) 国際公開日 平成27年11月26日 (2015.11.26)  
(31) 優先権主張番号 14/216,369  
(32) 優先日 平成26年3月17日 (2014.3.17)  
(33) 優先権主張国 米国(US)

(71) 出願人 595020643  
クアアルコム・インコーポレイテッド  
QUALCOMM INCORPORATED  
アメリカ合衆国、カリフォルニア州 92  
121-1714、サン・ディエゴ、モア  
ハウス・ドライブ 5775  
(74) 代理人 100108855  
弁理士 蔵田 昌俊  
(74) 代理人 100109830  
弁理士 福原 淑弘  
(74) 代理人 100158805  
弁理士 井関 守三  
(74) 代理人 100112807  
弁理士 岡田 貴志

最終頁に続く

(54) 【発明の名称】 メッシュベースの計算を使用した低複雑な順変換のためのシステムおよび方法

## (57) 【要約】

本明細書では、メッシュベースの計算を使用した低複雑な順変換のためのシステムおよび方法について説明する。本開示で説明する主題の一態様は、ビデオ情報を記憶するように構成されたメモリを備えるビデオエンコーダを提供する。ビデオエンコーダは、メモリと通信しているプロセッサをさらに備える。プロセッサは、変換を複数の変換段階に分解することを行うように構成される。プロセッサは、各変換段階における変換段階出力を決定するために複数の段階を使用してビデオ情報を変換することを行うようにさらに構成される。プロセッサは、各変換段階における変換段階出力を所定のビット深度に制約することを行うようにさらに構成される。プロセッサは、複数の段階の最後の段階の制約された変換出力に対して演算を実行すること、ここにおいて、演算が、所定のビット深度を有するデータとともに使用するためにのみ利用可能である、を行うようにさらに構成される。

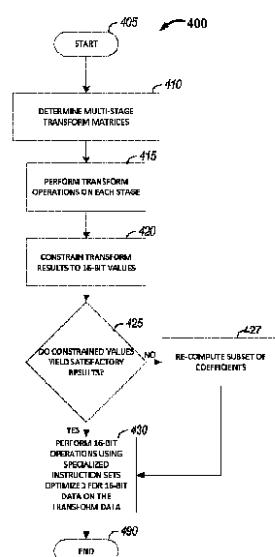


FIG. 4

**【特許請求の範囲】****【請求項 1】**

ビデオ情報を記憶するように構成されたメモリと、  
前記メモリと通信しているプロセッサであって、  
変換を複数の変換段階に分解することと、  
各変換段階における変換段階出力を決定するために前記複数の段階を使用して前記ビデオ情報を変換することと、  
各変換段階における前記変換段階出力を所定のビット深度に制約することと、  
前記複数の段階の最後の段階の前記制約された変換出力に対して演算を実行することと、  
ここにおいて、前記演算が、前記所定のビット深度を有するデータとともに使用するためにのみ利用可能である、  
を行うように構成されたプロセッサと  
を備える、ビデオエンコーダ。

**【請求項 2】**

前記プロセッサが、メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分解するようにさらに構成された、請求項 1 に記載のビデオエンコーダ。  
。

**【請求項 3】**

前記プロセッサは、クリッピングされた変換段階出力が必ず所定の範囲内に入るように前記変換段階出力をクリッピングすることによって前記変換段階出力を制約するようにさらに構成された、請求項 1 に記載のビデオエンコーダ。

**【請求項 4】**

前記所定のビット深度が 16 ビットである、請求項 1 に記載のビデオエンコーダ。

**【請求項 5】**

前記プロセッサが、前記制約された変換段階出力と前記変換段階出力との間の偏差を決定するようにさらに構成された、請求項 1 に記載のビデオエンコーダ。

**【請求項 6】**

前記偏差が、前記制約された変換段階出力と前記変換段階出力との間の前記差を備える、請求項 5 に記載のビデオエンコーダ。

**【請求項 7】**

前記プロセッサは、前記偏差が所定のしきい値よりも大きいとき、少なくとも 1 つの変換段階内で係数のサブセットを再計算するようにさらに構成された、請求項 5 に記載のビデオエンコーダ。

**【請求項 8】**

前記プロセッサが、少なくとも係数の前記再計算されたサブセットを使用して前記ビデオ情報を変換するようにさらに構成された、請求項 7 に記載のビデオエンコーダ。

**【請求項 9】**

ビデオを符号化する方法であって、  
ビデオ情報を記憶することと、  
変換を複数の変換段階に分解することと、  
各変換段階における変換段階出力を決定するために前記複数の段階を使用して前記ビデオ情報を変換することと、  
各変換段階における前記変換段階出力を所定のビット深度に制約することと、  
前記複数の段階の最後の段階の前記制約された変換出力に対して演算を実行することと、  
ここにおいて、前記演算が、前記所定のビット深度を有するデータとともに使用するためにのみ利用可能である、  
を備える方法。

**【請求項 10】**

メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分解することをさらに備える、請求項 9 に記載の方法。

10

20

30

40

50

**【請求項 1 1】**

クリッピングされた変換段階出力が必ず所定の範囲内に入るように前記変換段階出力をクリッピングすることによって前記変換段階出力を制約することをさらに備える、請求項9に記載の方法。

**【請求項 1 2】**

前記所定のビット深度が16ビットである、請求項9に記載の方法。

**【請求項 1 3】**

前記制約された変換段階出力と前記変換段階出力との間の偏差を決定することをさらに備える、請求項9に記載の方法。

**【請求項 1 4】**

前記偏差が、前記制約された変換段階出力と前記変換段階出力との間の前記差を備える、請求項13に記載の方法。

**【請求項 1 5】**

前記偏差が所定のしきい値よりも大きいとき、少なくとも1つの変換段階内で係数のサブセットを再計算することをさらに備える、請求項13に記載の方法。

**【請求項 1 6】**

少なくとも係数の前記再計算されたサブセットを使用して前記ビデオ情報を変換することをさらに備える、請求項15に記載の方法。

**【請求項 1 7】**

実行されたとき、装置に、  
ビデオ情報を記憶することと、  
変換を複数の変換段階に分解することと、  
各変換段階における変換段階出力を決定するために前記複数の段階を使用して前記ビデオ情報を変換することと、  
各変換段階における前記変換段階出力を所定のビット深度に制約することと、  
前記複数の段階の最後の段階の前記制約された変換出力に対して演算を実行することと、ここにおいて、前記演算が、前記所定のビット深度を有するデータとともに使用するためにのみ利用可能である、  
を行わせるコードを備える、非一時的コンピュータ可読媒体。

**【請求項 1 8】**

実行されたとき、前記装置に、メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分解することを行わせるコードをさらに備える、請求項17に記載のコンピュータ可読媒体。

**【請求項 1 9】**

実行されたとき、前記装置に、クリッピングされた変換段階出力が必ず所定の範囲内に入るように前記変換段階出力をクリッピングすることによって前記変換段階出力を制約することを行わせるコードをさらに備える、請求項17に記載のコンピュータ可読媒体。

**【請求項 2 0】**

実行されたとき、前記装置に、前記制約された変換段階出力と前記変換段階出力との間の偏差を決定することを行わせるコードをさらに備える、請求項17に記載のコンピュータ可読媒体。

**【請求項 2 1】**

前記偏差が、前記制約された変換段階出力と前記変換段階出力との間の前記差を備える、請求項20に記載のコンピュータ可読媒体。

**【請求項 2 2】**

実行されたとき、前記装置に、前記偏差が所定のしきい値よりも大きいとき、少なくとも1つの変換段階内で係数のサブセットを再計算することを行わせるコードをさらに備える、請求項20に記載のコンピュータ可読媒体。

**【請求項 2 3】**

実行されたとき、前記装置に、少なくとも係数の前記再計算されたサブセットを使用し

10

20

30

40

50

て前記ビデオ情報を変換することを行わせるコードをさらに備える、請求項 22 に記載のコンピュータ可読媒体。

**【請求項 24】**

ビデオを符号化するための装置であって、  
ビデオ情報を記憶するための手段と、  
変換を複数の変換段階に分解するための手段と、  
各変換段階における変換段階出力を決定するために前記複数の段階を使用して前記ビデオ情報を変換するための手段と、

各変換段階における前記変換段階出力を所定のビット深度に制約するための手段と、  
前記複数の段階の最後の段階の前記制約された変換出力に対して演算を実行するための手段と、ここにおいて、前記演算が、前記所定のビット深度を有するデータとともに使用するためにのみ利用可能である、

を備える、装置。

**【請求項 25】**

メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分解するための手段をさらに備える、請求項 25 に記載の装置。

**【請求項 26】**

クリッピングされた変換段階出力が必ず所定の範囲内に入るように前記変換段階出力をクリッピングすることによって前記変換段階出力を制約するための手段をさらに備える、請求項 25 に記載の装置。

**【請求項 27】**

前記制約された変換段階出力と前記変換段階出力との間の偏差を決定するための手段をさらに備える、請求項 25 に記載の装置。

**【請求項 28】**

前記偏差が、前記制約された変換段階出力と前記変換段階出力との間の前記差を備える、請求項 27 に記載の装置。

**【請求項 29】**

前記偏差が所定のしきい値よりも大きいとき、少なくとも 1 つの変換段階内で係数のサブセットを再計算するための手段をさらに備える、請求項 27 に記載の装置。

**【請求項 30】**

少なくとも係数の前記再計算されたサブセットを使用して前記ビデオ情報を変換するための手段をさらに備える、請求項 30 に記載の装置。

**【発明の詳細な説明】**

**【技術分野】**

**【0001】**

[0001] 本開示は、ビデオ符号化に関する。

**【背景技術】**

**【0002】**

[0002] デジタルビデオ能力は、デジタルテレビジョン、デジタルダイレクトブロードキャストシステム、ワイアレスブロードキャストシステム、携帯情報端末（PDA）、ラップトップコンピュータもしくはデスクトップコンピュータ、タブレットコンピュータ、電子ブックリーダー、デジタルカメラ、デジタル記録デバイス、デジタルメディアプレーヤ、ビデオゲームデバイス、ビデオゲームコンソール、セルラー電話もしくは衛星無線電話、スマートフォン、ビデオ会議デバイス、ビデオストリーミングデバイスなどを含む、広範囲のデバイスに組み込まれ得る。デジタルビデオデバイスは、MPEG-2、MPEG-4、ITU-T H.263、ITU-T H.264 / MPEG-4、Part 10、アドバンストビデオコーディング（AVC : Advanced Video Coding）、高効率ビデオコーディング（HEVC）によって定義された規格、およびそのような規格の拡張に記載されているビデオコーディング技法など、ビデオコーディング技法を実装する。ビデオデバイスは、そのようなビデオコーディング技法を実装することによって、デジタルビデ

10

20

30

40

50

才情報をより効率的に送信、受信、符号化、復号、および／または記憶し得る。

### 【0003】

[0003]ビデオコーディング技法は、ビデオシーケンスに固有の冗長性を低減または除去するための空間的（ピクチャ内）予測および／または時間的（ピクチャ間）予測を含む。ブロックベースのビデオコーディングでは、ビデオスライス（たとえば、ビデオフレームまたはビデオフレームの一部分）は、ツリーブロック、コーディングユニット（C U）、および／またはコーディングノードと呼ばれることがある、ビデオブロックに区分され得る。C Uの予測ビデオデータを決定するために、C Uはさらに1つまたは複数の予測ユニット（P U）に区分され得る。ビデオ圧縮技法はまた、C Uを、コーディングされるべきビデオブロックと予測ビデオデータとの間の差を表す残差ビデオブロックデータの1つまたは複数の変換ユニット（T U）に区分し得る。2次元離散コサイン変換（D C T）などの線形変換をT Uに適用して残差ビデオブロックデータをピクセル領域から周波数領域に変換して、さらなる圧縮を実現し得る。さらに、ピクチャのイントラコード化（I）スライスの中のビデオブロックは、同じピクチャ中の隣接ブロックにおける参照サンプルに対する空間的予測を使用して符号化され得る。ピクチャのインターフォーマット（PまたはB）スライス中のビデオブロックは、同じピクチャ中の隣接ブロック中の参照サンプルに対する空間的予測、または他の参照ピクチャ中の参照サンプルに対する時間的予測を使用し得る。ピクチャはフレームと呼ばれることがあり、参照ピクチャは参照フレームと呼ばれることがある。

10

### 【0004】

[0004]空間的予測または時間的予測は、コーディングされるべきブロックのための予測ブロックを生じる。残差データは、コーディングされるべき元のブロックと予測ブロックとの間のピクセル差分を表す。インターフォーマット（I）スライス中のビデオブロックは、同じピクチャ中の隣接ブロックにおける参照サンプルに対する空間的予測を使用して符号化され得る。ピクチャのインターフォーマット（PまたはB）スライス中のビデオブロックは、同じピクチャ中の隣接ブロック中の参照サンプルに対する空間的予測、または他の参照ピクチャ中の参照サンプルに対する時間的予測を使用し得る。ピクチャはフレームと呼ばれることがあり、参照ピクチャは参照フレームと呼ばれることがある。

20

### 【0005】

[0005]A V Cなどのより古いビデオ規格では、順変換および逆変換のサイズ（たとえば、 $4 \times 4$ および $8 \times 8$ ）は、ビデオ符号化のパフォーマンスのボトルネックとして作用しなかった。しかしながら、より最近のH E V C規格は、 $16 \times 16$ および $32 \times 32$ までの順方向変換サイズおよび逆変換サイズを利用し、これは、H E V Cプロセスの制限ファクタとして作用する。より大きい変換は、ピクセル領域から係数領域に変換するときに複雑さが増し、より多くのサイクルを処理することが必要になる。コーディング効率のために、規格は、ビデオエンコーダ中の大きい順変換ベクトルを複数の段階に分解するプロセス（たとえば、「メッシュベースの方法」、「バタフライ方法」または「偶数奇数分解」）と各段階において内部ビット深度を制約することから利益を得る。本明細書で開示する技法のいくつかの利点は、ビデオエンコーダ中の大きい順変換ベクトルを複数の段階に分解し、各段階において内部ビット深度を制約することによってビデオ符号化中のコーディング効率を改善し、計算リソース要件を低減することに関する。

30

### 【発明の概要】

### 【0006】

[0006]概して、本開示は、大きい順変換を複数の段階に分解し（たとえば、順変換を実装するメッシュベースの方法）、計算効率の良い命令セットを収容し得るレベルに各段階において内部ビット深度を制約することによってビデオ符号化パフォーマンスを改善することに関する技法について説明する。たとえば、変換に対する入力ビット深度は、9ビットであり得、開始内部ビット深度は、16ビットよりも大きくなり得るが、ビデオエンコ

40

50

ーダは、飽和論理を使用して（たとえば、内部値を16ビットにクリッピングして）変換の内部ビット深度を16ビットに制約するように構成され得る。ビデオ品質の損失を防ぐために、ビデオエンコーダは、偏差（たとえば、誤差）のレベルを測定し、レベルをしきい値と比較し、偏差のレベルがしきい値を超える場合、変換された係数のサブセットを再計算するようにさらに構成され得る。

## 【0007】

[0007] 1つまたは複数の例の詳細が以下の添付の図面および説明に記載されている。他の特徴、目的、および利点は、説明および図面、ならびに特許請求の範囲から明らかになろう。

## 【0008】

[0008] 本開示で説明する主題の一態様は、ビデオ情報を記憶するように構成されたメモリを備えるビデオエンコーダを提供する。ビデオエンコーダは、メモリと通信しているプロセッサをさらに備える。プロセッサは、変換を複数の変換段階に分解することを行うように構成される。プロセッサは、各変換段階における変換段階出力を決定するために複数の段階を使用してビデオ情報を変換することを行なうようにさらに構成される。プロセッサは、各変換段階における変換段階出力を所定のビット深度に制約することを行なうようにさらに構成される。プロセッサは、複数の段階の最後の段階の制約された変換出力に対して演算を実行すること、ここにおいて、演算が、所定のビット深度を有するデータとともに使用するためにのみ利用可能である、を行なうようにさらに構成される。

10

## 【0009】

[0009] 本開示で説明する主題の別の態様は、ビデオを符号化する方法を提供する。本方法は、ビデオ情報を記憶することを含む。本方法は、変換を複数の変換段階に分解することをさらに含む。本方法は、各変換段階における変換段階出力を決定するために複数の段階を使用してビデオ情報を変換することをさらに含む。本方法は、各変換段階における変換段階出力を所定のビット深度に制約することをさらに含む。本方法は、複数の段階の最後の段階の制約された変換出力に対して演算を実行すること、ここにおいて、演算が、所定のビット深度を有するデータとともに使用するためにのみ利用可能である、をさらに含む。

20

## 【0010】

[0010] 本開示で説明する主題の別の態様は非一時的コンピュータ可読媒体を提供する。本媒体は、実行されたとき、装置に、ビデオ情報を記憶することを行わせるコードを備える。本媒体は、実行されたとき、装置に、変換を複数の変換段階に分解することを行わせるコードをさらに備える。本媒体は、実行されたとき、装置に、各変換段階における変換段階出力を決定するために複数の段階を使用してビデオ情報を変換することを行わせるコードをさらに備える。本媒体は、実行されたとき、装置に、各変換段階における変換段階出力を所定のビット深度に制約することを行わせるコードをさらに備える。本媒体は、実行されたとき、装置に、複数の段階の最後の段階の制約された変換出力に対して演算を実行すること、ここにおいて、演算が、所定のビット深度を有するデータとともに使用するためにのみ利用可能である、を行わせるコードをさらに備える。

30

## 【0011】

[0011] 本開示で説明する主題の別の態様は、ビデオを符号化するための装置を提供する。本装置は、ビデオ情報を記憶するための手段を備える。本装置は、変換を複数の変換段階に分解するための手段をさらに備える。本装置は、各変換段階における変換段階出力を決定するために複数の段階を使用してビデオ情報を変換するための手段をさらに備える。本装置は、各変換段階における変換段階出力を所定のビット深度に制約するための手段をさらに備える。本装置は、複数の段階の最後の段階の制約された変換出力に対して演算を実行するための手段、ここにおいて、演算が、所定のビット深度を有するデータとともに使用するためにのみ利用可能である、をさらに備える。

40

## 【図面の簡単な説明】

## 【0012】

50

【図1】[0012]本開示で説明する態様による技法を利用し得る例示的なビデオ符号化および復号システムを示すブロック図。

【図2】[0013]本開示で説明する態様による技法を実装し得るビデオエンコーダの一例を示すブロック図。

【図3】[0014]本開示で説明する態様による技法を実装し得るビデオデコーダの一例を示すブロック図。

【図4】[0015]たとえば、メッシュベースの計算を使用した低複雑な順変換の方法のフローチャート。

【図5】[0016]低複雑な順変換のための方法のフローチャート。

【発明を実施するための形態】

【0013】

[0017]図面に示す様々な特徴は一定の縮尺で描かれていねことがある。したがって、様々な特徴の寸法は、明快のために恣意的に拡大または縮小されていることがある。さらに、図面のいくつかは、所与のシステム、方法またはデバイスの構成要素のすべてを示しているとは限らないことがある。最後に、本明細書および図の全体にわたって、同様の特徴を示すために同様の参照番号が使用されることがある。

【0014】

[0018]本開示で説明する技法は、概して、特に、高効率ビデオコーディング(HEVC)規格およびその拡張に関してビデオ符号化中の順変換に関する。

【0015】

[0019]ビデオコーディング規格は、ITU-T H.261と、ISO/IEC MPEG-1 Visualと、ITU-T H.262またはISO/IEC MPEGL-2 Visualと、ITU-T H.263と、ISO/IEC MPEGL-4 Visualと、そのスケーラブルビデオコーディング(SVC)拡張およびマルチビュービデオコーディング(MVC)拡張を含む(ISO/IEC MPEGL-4 AVCとしても知られる)ITU-T H.264とを含む。さらに、ITU-Tビデオコーディングエキスパートグループ(VCEG:Video Coding Experts Group)とISO/IECモーションピクチャエキスパートグループ(MPEG:Motion Picture Experts Group)とのジョイントコラボレーションチームオンビデオコーディング(JCT-VC:Joint Collaboration Team on Video Coding)によって開発された新しいビデオコーディング規格、高効率ビデオコーディング(HEVC)がある。

【0016】

[0020]上述のように、AVC規格は、最大 $8 \times 8$ 変換サイズしか利用しなかつたが、HEVC規格は、サイズが最大 $32 \times 32$ の(たとえば、HEVCエンコーダでの)順変換および逆変換を利用する。より大きい変換サイズは、HEVCにおける大きいコードブロックのコーディング効率が増加するが、それらはまた、より小さい変換サイズを使用することと比較して複雑さ、計算サイクル、および処理時間が増加する。本開示で説明する方法は、エンコーダがピクセル領域から係数領域にビデオ情報を変換するときに必要とされる複雑さおよびサイクルの増加を低減し得る。たとえば、いくつかの方法は、ビデオエンコーダ中の大きい順変換ベクトルを複数の段階に分解することと(たとえば、メッシュベースの方法)、各段階において内部ビット深度を制約することとを含む。

【0017】

[0021]いくつかの実装形態では、残差領域から係数領域に転換するために(たとえば、順変換を実装するメッシュベースの方法または「バタフライ」方法を使用して)大きい順変換を複数の段階に分解することは、行列乗算方法を使用するよりも効率的な処理を生じ得る。以下の付録Aのコードに、メッシュベースの $N \times N$ の変換の実装形態の一例を示す。一実装形態では、変換ユニット(たとえば、 $16 \times 16$ の変換)は、256個の残差ソースピクセル(たとえば、付録AにおけるPSRC)で開始し得、その各々がルーマ値を表し得る。プロセッサまたはエンコーダ(たとえば、図2のエンコーダ20の変換処理ユニット52)は、次いで、ソースピクセルごとに1つずつ、256個の出力係数(たとえ

10

20

30

40

50

ば、付録Aにおける $p_{DST}$ を決定し得る。第1の段階の後に、プロセッサまたはエンコーダ（たとえば、変換処理ユニット52）は、次いで、2つのピクセルの128個の和と2つのピクセルの128個の差と（たとえば、付録AのnEのものとnOのものと）を決定し得る。第2の処理段階中に、プロセッサまたはエンコーダは、ペアの和と差の和とを決定するために、nEとnOとの和および差を使用し得る。4つの段階の後に、プロセッサまたはエンコーダは、出力係数（たとえば、 $p_{DST}$ ）を生成し得る。（付録Aに示した）このメッシュベースの方法を使用して、フルサイズの変換は、いくつかのより小さい、あまり複雑でない変換に分解され得、これは、一緒に乗じられた場合、フルサイズの変換を再び生成することになる。

## 【0018】

10

[0022]上記で説明したメッシュ方法などのメッシュ方法または任意の他の分解方法を実行した後に、本開示で説明する方法は、各段階において内部ビット深度のあるレベルに制約するようにプロセッサまたはエンコーダを構成して、プロセッサまたはエンコーダが、そのレベルのためにさらに計算効率の良い命令セットを利用することを可能にし得る。実際、いくつかのエンコーダおよびデコーダは、いくつかのビット深度を有する入力とともに使用するためにのみ利用可能である計算効率の良い命令セットを含む。たとえば、変換に対する入力ビット深度は、9ビットであり得、開始内部ビット深度は、16ビットよりも大きくなり得るが、ビデオエンコーダは、変換の各分解された段階の内部ビット深度（たとえば、変換出力ビット深度だけではなく変換演算中のビット深度）を16ビットに制約するように構成され得る。各段階の内部ビット深度が16ビットに制約されるので、プロセッサは、16ビット演算とともに使用するために特に設計された計算効率の良い命令セット（たとえば、ARMアーキテクチャ、Advanced SIMD（NEON）、デジタル信号処理（DSP）など）を利用し得る。一実装形態では、ビデオエンコーダは、飽和論理を使用して（たとえば、内部値を16ビットにクリッピングして）各段階において内部ビット深度を制約し得る。

20

## 【0019】

[0023]場合によっては、変換段階においてビット深度を制約することは、最終ビデオ品質の低下をもたらし得る。この結果を防ぐために、本開示で説明する方法は、制約プロセスによって生じる偏差（たとえば、誤差）のレベルを測定し得る。本方法は、次いで、偏差のレベルを所定のしきい値（たとえば、耐久限界）と比較し、次いで、偏差のレベルが所定のしきい値を超える場合、変換された係数のサブセットを再計算し得る。このようにしてビット深度を制約し、係数のサブセットを再計算することによって、本開示で説明する方法により、ビデオエンコーダの変換処理ユニットが、ビデオ品質をも保持しながら、より少ない計算リソースを使用することが可能になり得る。

30

## 【0020】

[0024] ブロックベースの処理を使用するビデオコーデック（たとえば、ビデオフレームがビデオブロックまたはコーディングユニットに区分され得るHEVC）では、（たとえば、インター予測またはイントラ予測からの）予測ブロックまたは予測ユニットが元のピクセルから減算され得る。上記でさらに説明したように、残差データは、次いで、順変換（たとえば、離散コサイン変換）を使用して残差変換係数に変換され、量子化され、（たとえば、さらなる圧縮を達成するために）エントロピー符号化され得る。エントロピー符号化は、以下でさらに説明する様々なエントロピーコーディングエンジン（たとえば、CAVLC、CABACなど）を使用して実行され得る。その後、同じく以下でさらに説明するように、デコーダは、次いで、係数をエントロピー復号し、逆量子化し、逆変換し得る。最後に、係数は、予測ブロックに追加されて、再構成されたピクセルを形成し得る。

40

## 【0021】

[0025] ビデオコーディングの一実施形態では、画像ブロックは、最初に、再構成された、一時的におよび／または空間的に隣接するブロック中のピクセルを使用して予測され得る。（「残余」と呼ばれることがある）予測誤差が、次いで、変換され、量子化され得る。たとえば、Sが、サイズが $N \times N$ の残余ブロックである場合、変換されたブロックKは

50

、次のように行列乗算を使用して導出され得る。

【0 0 2 2】

【数1】

$$K = A * S * B$$

【0 0 2 3】

ここで、K、A、およびBも、N × Nのサイズである。Aは垂直変換行列であり、Bは水平変換行列である。いくつかの実施形態では、AおよびBは、互いの転置である（たとえば、 $B = A'$ であり、ここで、「」は転置を意味する）。他の実施形態では、AおよびBは、互いの転置でない。AおよびBが互いの転置であるとき、前の式は次のようになる。

10

【0 0 2 4】

【数2】

$$K = A * S * A'$$

20

【0 0 2 5】

[0026]各変換（AおよびB）は、様々な変換のいずれかを含み得る。いくつかの実施形態では、変換は、離散コサイン変換（DCT）、離散サイン変換（DST）、アダマール変換、ハール変換などのうちの1つを含む。

【0 0 2 6】

[0027]SVC拡張では、ビデオ情報の複数のレイヤがあり得る。最下層は、ベースレイヤ（BL）として働き得、最上層は、拡張レイヤ（EL）または「エンハンスマントレイヤ」として働き得る。最上層と最下層との間のすべてのレイヤは、ELまたはBLの一方または両方として働き得る。SVCは、品質スケーラビリティ（または信号対雑音比、SNR）、空間スケーラビリティ、および/または時間スケーラビリティを与えるために使用され得る。エンハンスマントレイヤは、ベースレイヤとは異なる空間解像度を有し得る。現在のブロックの予測は、SVCのために与えられる様々なレイヤを使用して実行され得る。そのような予測は、レイヤ間予測と呼ばれることがある。レイヤ間予測方法は、レイヤ間冗長性を低減するためにSVCにおいて利用され得る。レイヤ間予測のいくつかの例としては、レイヤ間イントラ予測、レイヤ間動き予測、およびレイヤ間残差予測があり得る。レイヤ間イントラ予測は、エンハンスマントレイヤ中の現在のブロックを予測するために、ベースレイヤ中のコロケートされたブロックの再構成を使用する。レイヤ間動き予測は、エンハンスマントレイヤ内の動きを予測するのにベースレイヤの動きを使用する。レイヤ間残差予測は、エンハンスマントレイヤの残差を予測するために、ベースレイヤの残差を使用する。「イントラBLモード」と呼ばれる、エンハンスマントレイヤのための1つの特定のコーディングモードは、ベースレイヤ中の対応する（「コロケートされた」と呼ばれる、たとえば、同じ空間的ロケーションに位置する）ブロックのテクスチャを使用して予測され得るテクスチャを含む。

30

【0 0 2 7】

[0028]レイヤ間残差予測では、ベースレイヤの残差は、エンハンスマントレイヤ中の現在のブロックを予測するために使用され得る。残差は、ビデオユニットの時間的予測とソースビデオユニットとの間の差分として定義され得る。残差予測では、ベースレイヤの残余はまた、現在のブロックを予測する際に考慮される。たとえば、現在のブロックは、エンハンスマントレイヤからの残余、エンハンスマントレイヤからの時間的予測、およびベースレイヤからの残余を使用して再構成され得る。現在のブロックは以下の式に従って再

40

50

構成され得る。

【0028】

【数3】

$$\hat{I}e = re + Pe + rb$$

【0029】

ここで、

10

【0030】

【数4】

$$\hat{I}e$$

【0031】

は現在のブロックの再構成を示し、 $r_e$ はエンハンスマントレイヤからの残余を示し、 $P_e$ はエンハンスマントレイヤからの時間的予測を示し、 $r_b$ はベースレイヤからの残余予測を示す。

【0032】

20

[0029] 差分領域を使用したインターフェーディングの場合、現在予測ブロックは、エンハンスマントレイヤ参照ピクチャ中の対応する予測ブロックサンプルと、スケーリングされたベースレイヤ参照ピクチャ中の対応する予測ブロックサンプルとの間の差分値に基づいて決定される。差分値は差分予測ブロック (difference predicted block) と呼ばれることがある。エンハンスマントレイヤ予測サンプルを取得するために、コロケートベースレイヤ再構成サンプルが差分予測ブロックに追加される。

【0033】

[0030] 本開示で説明する技法は、HEVCにおける順変換の行列乗算中の複雑な計算要件に関する問題に対処し得る。本技法は、エンコーダおよび／または変換処理ユニットが順変換行列乗算を実行し得る速度、効率、および有効性を改善し得る。

30

【0034】

[0031] 添付の図面を参照しながら、新規のシステム、装置、および方法の様々な態様について、以下でより十分に説明する。ただし、本開示は、多くの異なる形態で具現化される場合があり、本開示全体にわたって提示される任意の特定の構造または機能に限定されるものと解釈されるべきではない。むしろ、本開示が、入念で完全であり、本開示の範囲を当業者に十分に伝達するように、これらの態様が提供される。本明細書の教示に基づいて、単独で実装されるか、それとも本発明の何らかの他の態様と組み合わされるかにかかわらず、本開示の範囲が本明細書で開示される新規なシステム、装置、および方法のどんな態様も包含するものとすることを当業者は理解されたい。たとえば、本明細書に記載の任意の数の態様を使用して装置が実装され得、または方法が実施され得る。さらに、本発明の範囲は、本明細書に記載の本発明の様々な態様に加えて、またはそれ以外の、他の構造、機能、または構造および機能を使用して実施される装置または方法を包含するものとする。本明細書で開示されるすべての態様が、ある請求項の1つまたは複数の要素によって実施され得ることを理解されたい。

40

【0035】

[0032] 特定の態様が本明細書で説明されるが、これらの態様の多数の変形および置換が、本開示の範囲に含まれる。好適な態様のいくつかの利益および利点について説明するが、本開示の範囲は特定の利益、使用、または目的に限定されるものではない。むしろ、本開示の態様は、その一部が例として図面および好ましい態様の以下の説明において示される、異なるワイヤレス技術と、システム構成と、ネットワークと、伝送プロトコルとに幅

50

広く適用可能であることが意図されている。この詳細な説明および図面は、限定的であるのではなく本開示の説明に役立つものにすぎず、本開示の範囲は、添付の特許請求の範囲とその同等物とによって定義される。

### 【0036】

[0033] 図1は、本開示で説明する態様による技法を利用し得る例示的なビデオ符号化および復号システムを示すプロック図である。図1に示すように、ビデオ符号化および復号システム10は、宛先デバイス14によって後で復号されるべき符号化ビデオデータを提供するソースデバイス12を含む。特に、ソースデバイス12は、コンピュータ可読媒体16を介して宛先デバイス14にビデオデータを与える。ソースデバイス12および宛先デバイス14は、デスクトップコンピュータ、ノートブック（たとえば、ラップトップ）コンピュータ、タブレットコンピュータ、セットトップボックス、電話ハンドセット（たとえば、スマートフォン）、テレビジョン、カメラ、ディスプレイデバイス、デジタルメディアプレーヤ、ビデオゲームコンソール、ビデオストリーミングデバイスなどを含む、広範囲にわたるデバイスのいずれかを備え得る。場合によっては、ソースデバイス12および宛先デバイス14は、ワイヤレス通信のために装備され得る。  
10

### 【0037】

[0034] 上述のように、宛先デバイス14は、コンピュータ可読媒体16を介して、復号されるべき符号化ビデオデータを受信し得る。コンピュータ可読媒体16は、符号化されたビデオデータをソースデバイス12から宛先デバイス14に移動することができる、任意のタイプの媒体またはデバイスを備え得る。一例では、コンピュータ可読媒体16は、ソースデバイス12が符号化ビデオデータを宛先デバイス14にリアルタイムで直接送信するのを可能にするための通信媒体（図示せず）を備え得る。符号化ビデオデータは、ワイヤレス通信プロトコルなどの通信規格に従って変調され、宛先デバイス14に送信され得る。通信媒体は、無線周波数（RF）スペクトルあるいは1つまたは複数の物理伝送線路など、任意のワイヤレスまたはワイヤード通信媒体を備え得る。通信媒体は、ローカルエリアネットワーク、ワイドエリアネットワーク、または（インターネットなどの）グローバルネットワークのような、パケットベースのネットワークの一部を形成し得る。通信媒体は、ルータ、スイッチ、基地局、またはソースデバイス12から宛先デバイス14への通信を可能にするために有用であり得る任意の他の機器を含み得る。  
20

### 【0038】

[0035] いくつかの例では、符号化データは、出力インターフェース22からストレージデバイス（図示せず）に出力され得る。同様に、符号化データは入力インターフェース28によってストレージデバイスからアクセスされ得る。ストレージデバイスは、ハードドライブ、Blue-rayディスク、DVD、CD-ROM、フラッシュメモリ、揮発性もしくは不揮発性のメモリ、または符号化ビデオデータを記憶するための任意の他の適切なデジタル記憶媒体など、様々な分散されたまたはローカルにアクセスされるデータ記憶媒体のいずれをも含み得る。さらなる例では、ストレージデバイスは、ソースデバイス12によって生成された符号化ビデオを記憶することができるファイルサーバまたは別の中間ストレージデバイスに対応し得る。宛先デバイス14は、ストリーミングまたはダウンロードを介して、ストレージデバイスから記憶されたビデオデータにアクセスすることができる。ファイルサーバは、符号化ビデオデータを記憶し、その符号化ビデオデータを宛先デバイス14に送信することが可能な任意のタイプのサーバであり得る。例示的なファイルサーバには、ウェブサーバ（たとえば、ウェブサイト用の）、FTPサーバ、ネットワークアタッチドストレージ（NAS）デバイス、またはローカルディスクドライブがある。宛先デバイス14は、インターネット接続を含む、任意の標準のデータ接続を通して符号化ビデオデータにアクセスし得る。データ接続は、ファイルサーバに記憶された符号化されたビデオデータにアクセスするのに好適であるワイヤレスチャネル（たとえば、Wi-Fi（登録商標）接続）、有線接続（たとえば、DSL、ケーブルモデムなど）、または両方の組合せを含み得る。ストレージデバイスからの符号化されたビデオデータの送信は、ストリーミング送信、ダウンロード送信、またはそれらの組合せであり得る。  
30  
40  
50

## 【0039】

[0036]本開示の技法は、ワイヤレス応用またはワイヤレス設定に必ずしも制限されない。本技法は、無線テレビジョンブロードキャスト、ケーブルテレビジョン送信、衛星テレビジョン送信、HTTP上の動的適応ストリーミング(DASH:dynamic adaptive streaming over HTTP)などのインターネットストリーミングビデオ送信、データ記憶媒体上に符号化されたデジタルビデオ、データ記憶媒体上に記憶されたデジタルビデオの復号、または他の応用例など、様々なマルチメディア応用のいずれかをサポートするビデオコーディングに適用され得る。いくつかの例では、システム10は、ビデオストリーミング、ビデオ再生、ビデオブロードキャスト、ビデオテレフォニーなどの適用例のための一方向または双方向のビデオ送信をサポートするように構成され得る。

10

## 【0040】

[0037]図1の例では、ソースデバイス12は、ビデオソース18と、ビデオエンコーダ20と、出力インターフェース22とを含む。宛先デバイス14は、入力インターフェース28と、ビデオデコーダ30と、ディスプレイデバイス32とを含む。本開示によれば、ソースデバイス12のビデオエンコーダ20は、複数の規格または規格拡張に準拠するビデオデータを含む、ビットストリームをコーディングするための技法を適用するように構成され得る。他の例では、ソースデバイス12および宛先デバイス14は、他の構成要素または構成を含み得る。たとえば、ソースデバイス12は、外部カメラなどの外部ビデオソースからビデオデータを受信し得る。同様に、宛先デバイス14は、内蔵ディスプレイデバイス32ではなく、外部ディスプレイデバイスとインターフェースし得る。

20

## 【0041】

[0038]概して、本開示の技法は、ビデオ符号化デバイスによって実行されるが、これらの技法は、通常は「コーデック」と呼ばれるビデオエンコーダ/デコーダによっても実行され得る。さらに、本開示の技法はまた、ビデオプリプロセッサによって実行され得る。ソースデバイス12および宛先デバイス14は、ソースデバイス12が、宛先デバイス14に送信するためのコーディングされたビデオデータを生成するコーディングデバイスの例にすぎない。いくつかの例では、ソースデバイス12および宛先デバイス14は、それらがそれぞれビデオ符号化構成要素とビデオ復号構成要素とを含むように、実質的に対称的に動作し得る。したがって、システム10は、たとえば、ビデオストリーミング、ビデオ再生、ビデオブロードキャスト、ビデオテレフォニーなどのために、ソースデバイス12と宛先デバイス14との間の一方向または双方向のビデオ送信をサポートし得る。

30

## 【0042】

[0039]ソースデバイス12のビデオソース18は、ビデオカメラ、以前にキャプチャされたビデオを含んでいるビデオアーカイブ、ビデオコンテンツプロバイダからビデオを受信するためのビデオフィードインターフェースなどのビデオキャプチャデバイス(図示せず)を含み得る。さらなる代替として、ビデオソース18は、コンピュータグラフィックスベースのデータ、または、ライブビデオ、アーカイブされたビデオ、およびコンピュータ生成ビデオの組合せを生成することができる。場合によっては、ビデオソース18がビデオカメラである場合、ソースデバイス12および宛先デバイス14は、カメラ付き携帯電話またはビデオ付き携帯電話であり得る。別の実施形態では、本開示で説明する技法は、ビデオコーディング全般に適用可能であり得、ワイヤレスおよび/またはワイヤード適用例に適用され得る。各々の場合において、キャプチャされたビデオ、事前にキャプチャされたビデオ、またはコンピュータで生成されたビデオは、ビデオエンコーダ20によって符号化され得る。符号化されたビデオ情報は、次いで、出力インターフェース22によってコンピュータ可読媒体16に出力され得る。

40

## 【0043】

[0040]コンピュータ可読媒体16は、ワイヤレスブロードキャストまたはワイヤードネットワーク送信などの一時媒体、あるいはハードディスク、フラッシュドライブ、コンパクトディスク、デジタルビデオディスク、Blu-ray(登録商標)ディスク、または他のコンピュータ可読媒体などの記憶媒体(すなわち、非一時的記憶媒体)を含み得る。

50

いくつかの例では、ネットワークサーバ（図示せず）は、ソースデバイス 12 から符号化ビデオデータを受信し、たとえば、ネットワーク送信、直接ワイヤード通信などを介して、その符号化ビデオデータを宛先デバイス 14 に与え得る。同様に、ディスクスタンピング設備など、媒体製造設備のコンピューティングデバイスは、ソースデバイス 12 から符号化されたビデオデータを受信し、その符号化されたビデオデータを包含しているディスクを生成し得る。したがって、コンピュータ可読媒体 16 は、様々な形態の 1つまたは複数のコンピュータ可読媒体を含み得る。

#### 【 0 0 4 4 】

[0041]宛先デバイス 14 の入力インターフェース 28 は、コンピュータ可読媒体 16 から情報を受信し得る。コンピュータ可読媒体 16 の情報は、ビデオエンコーダ 20 によって定義されるシンタックス情報を含み得る。シンタックス情報はまた、ビデオデコーダ 30 によって使用され得、これは、ロックおよび他のコード化ユニットの特性および／または処理を記述するシンタックス要素を含み得る。ディスプレイデバイス 32 は、復号されたビデオデータをユーザに表示し得、陰極線管（C R T）、液晶ディスプレイ（L C D）、プラズマディスプレイ、有機発光ダイオード（O L E D）ディスプレイ、または別のタイプのディスプレイデバイスなどの様々なディスプレイデバイスのうちのいずれかを備え得る。10

#### 【 0 0 4 5 】

[0042]ビデオエンコーダ 20 およびビデオデコーダ 30 は、高効率ビデオコーディング（H E V C）規格またはその変形形態のいずれか（たとえば、H E V C テストモデル（H M））などのビデオコーディング規格に従って動作し得る。代替的に、ビデオエンコーダ 20 およびビデオデコーダ 30 は、I T U - T H . 2 6 4 規格（M P E G - 4）、P a r t 1 0、アドバンストビデオコーディング（A V C）、I T U - T H . 2 6 3、I T U - T H . 2 6 2（I S O / I E C M P E G - 2 V i s u a l）、I S O / I E C M P E G - 1 V i s u a l、I T U - T H . 2 6 1、またはいずれのそのような規格の拡張などの他のプロプライエタリまたは業界規格に従って動作し得る。いくつかの態様では、ビデオエンコーダ 20 およびビデオデコーダ 30 は、共通のデータストリームまたは別個のデータストリーム中のオーディオとビデオの両方の符号化を処理するために、オーディオエンコーダ、オーディオデコーダ、M U X - D E M U X ユニット（図示せず）、または他のハードウェアおよびソフトウェアと一体化され得る。20

#### 【 0 0 4 6 】

[0043]ビデオエンコーダ 20 およびビデオデコーダ 30 は各々、1つまたは複数のマイクロプロセッサ、デジタル信号プロセッサ（D S P）、特定用途向け集積回路（A S I C）、フィールドプログラマブルゲートアレイ（F P G A）、ディスクリート論理、ソフトウェア、ハードウェア、ファームウェア、またはそれらの任意の組合せなどの様々な適切なエンコーダ回路のいずれかとして実装され得る。本開示の技法を実行する必要がある場合、ビデオエンコーダ 20 および／またはビデオデコーダ 30 は、好適な非一時的コンピュータ可読媒体中にソフトウェアの命令を記憶し、1つまたは複数のプロセッサを使用してハードウェアで命令を実行し得る。ビデオエンコーダ 20 およびビデオデコーダ 30 の各々は、1つまたは複数のエンコーダまたはデコーダに含まれ得、そのいずれも、それぞれのデバイスにおいて複合エンコーダ／デコーダ（コーデック）の一部として統合され得る。ビデオエンコーダ 20 および／またはビデオデコーダ 30 を含むデバイスは、集積回路、マイクロプロセッサ、および／またはセルフオンなどのワイヤレス通信デバイスを備え得る。30

#### 【 0 0 4 7 】

[0044]H E V C 規格は、ビデオフレームまたはピクチャが、ルーマサンプルとクロマサンプルの両方を含む一連のツリープロックまたは最大コーディングユニット（L C U）に分割され得ることを規定する。ビットストリーム内のシンタックスデータが、ピクセルの数に関して最大のコーディングユニットであるL C Uのサイズを定義し得る。スライスは、コーディングの順序で、いくつかの連続するツリープロックを含む。ビデオフレームま40

たはピクチャは、1つまたは複数のスライスに区分化され得る。各ツリーブロックは、4分木に従ってコーディングユニット(CU)に分割され得る。概して、4分木データ構造はCUごとに1つのノードを含み、ルートノードはツリーブロックに対応する。CUが4つのサブCUに分割される場合、CUに対応するノードは、4つのリーフノードを含み、その各々は、サブCUの1つに対応する。

#### 【0048】

[0045] 4分木データ構造の各ノードは、対応するCUのためのシンタックスデータを与える。たとえば、4分木内のノードは、そのノードに対応するCUがサブCUに分割されるか否かを示すスプリットフラグを含み得る。CUのシンタックス要素は、再帰的に定義され得、CUがサブCUに分割されるか否かに依存し得る。CUがこれ以上分割されない場合、そのCUはリーフCUと呼ばれる。本開示では、元のリーフCUの明示的な分割が存在しない場合でも、リーフCUの4つのサブCUはリーフCUと呼ばれる。たとえば、 $16 \times 16$  サイズのCUがさらに分割されない場合、この $16 \times 16$  CUが決して分割されなくても、4つの $8 \times 8$  サブCUはリーフCUとも呼ばれることになる。

10

#### 【0049】

[0046] CUは、CUがサイズの特異性を有しないことを別にすれば、H.264規格のマクロブロックと同様の目的を有する。たとえば、ツリーブロックは、4つの子ノード(サブCUとも呼ばれる)に分割され得、各子ノードは、次に親ノードとなり、別の4つの子ノードに分割され得る。4分木のリーフノードと呼ばれる、最後の分割されない子ノードは、リーフCUとも呼ばれるコーディングノードを備える。コーディングされたビットストリームに関連するシンタックスデータは、最大CU深度と呼ばれる、ツリーブロックが分割され得る最大回数を定義し得、また、コーディングノードの最小サイズを定義し得る。それに応じて、ビットストリームは最小コーディングユニット(SCU)をも定義し得る。本開示では、HEVCのコンテキストにおけるCU、PU、またはTU、あるいは他の規格のコンテキストにおける同様のデータ構造(たとえば、H.264/AVCにおけるマクロブロックおよびそのサブブロック)のいずれかを指すために「ブロック」という用語を使用する。

20

#### 【0050】

[0047] CUは、コーディングノードと、コーディングノードと関連付けられた予測ユニット(PU)および変換ユニット(TU)とを含む。CUのサイズは、コーディングノードのサイズに対応し、形状が方形でなければならない。CUのサイズは、 $8 \times 8$  ピクセルから、いくつかの事例では、 $64 \times 64$  ピクセル以上の最大値を有するツリーブロックのサイズにまで及び得る。各CUは、1つまたは複数のPUと1つまたは複数のTUとを含し得る。CUに関連付けられたシンタックスデータは、たとえば、1つまたは複数のPUへのCUの区分を記述し得る。区分モードは、CUがスキップモード符号化もしくは直接モード符号化されるか、イントラ予測モード符号化されるか、またはインター予測モード符号化されるかで異なり得る。PUは、形状が非方形に区分され得る。CUに関連するシンタックスデータはまた、たとえば、4分木に従うCUの1つまたは複数のTUへの区分を記述し得る。TUは、形状が方形または非方形(たとえば、長方形)であり得る。

30

#### 【0051】

[0048] HEVC規格は、異なるCUに対しては異なり得る、TUに従った変換を可能にする。TUは通常、区分されたLCUに対して定義される所与のCU内のPUのサイズに基づくサイズにされるが、必ずそうであるとは限らない。TUは、一般に、PUと同じサイズであるか、またはそれよりも小さい。いくつかの例では、CUに対応する残差サンプルは、「残差4分木」(RQT: residual quad tree)と呼ばれる4分木構造を使用して、より小さいユニットに再分割され得る。RQTのリーフノードは、変換ユニット(TU)と呼ばれ得る。TUに関連するピクセル差分値は、変換係数を生成するために変換され、その変換係数は量子化され得る。

40

#### 【0052】

[0049] リーフCUは、1つまたは複数の予測ユニット(PU)を含み得る。概して、P

50

Uは、対応するCUの全部または一部分に対応する空間エリアを表し、そのPUの参照サンプルを取り出すためのデータを含み得る。その上、PUは、予測に関するデータを含む。たとえば、PUがイントラモードで符号化されるとき、PUに関するデータは、PUに対応するTUに関するイントラ予測モードを記述するデータを含み得る残差4分木(RQT)内に含まれ得る。別の例として、PUがインターモードで符号化されるとき、PUは、PUのための1つまたは複数の動きベクトルを定義するデータを含み得る。PUのための動きベクトルを定義するデータは、たとえば、動きベクトルの水平成分、動きベクトルの垂直成分、動きベクトルの分解能(たとえば、1/4ピクセル精度または1/8ピクセル精度)、動きベクトルが指す参照ピクチャ、および/または動きベクトルの参照ピクチャリスト(たとえば、リスト0、リスト1、またはリストC)を記述することができる。

10

### 【0053】

[0050] 1つまたは複数のPUを有するリーフCUは、1つまたは複数の変換ユニット(TU)を含み得る。変換ユニットは、上で論じられたように、RQT(TUの4分木構造とも呼ばれる)を使用して規定され得る。たとえば、分割フラグは、リーフCUが4つの変換ユニットに分割されるかどうかを示し得る。次いで、各変換ユニットは、さらなるサブTUに、さらに分割され得る。TUがこれ以上分割されないとき、そのTUはリーフTUと呼ばれ得る。一般に、イントラコーディングの場合、リーフCUに属するすべてのリーフTUは同じイントラ予測モードを共有する。すなわち、同じイントラ予測モードが、概して、リーフCUのすべてのTUの予測値を計算するために適用される。イントラコーディングの場合、ビデオエンコーダは、イントラ予測モードを使用して各リーフTUの残差値を、TUに対応するCUの一部分と元のブロックとの間の差分として計算し得る。TUは、必ずしもPUのサイズに制限されるとは限らない。したがって、TUは、PUよりも大きくまたは小さくなり得る。イントラコーディングの場合、PUは、同じCUのための対応するリーフTUとコロケートされ得る。いくつかの例では、リーフTUの最大サイズは、対応するリーフCUのサイズに対応し得る。

20

### 【0054】

[0051] その上、リーフCUのTUは、残差4分木(RQT)と呼ばれる、それぞれの4分木データ構造にも関連付けられ得る。すなわち、リーフCUは、リーフCUがどのようにTUに区分されるかを示す4分木を含み得る。TU4分木のルートノードは概してリーフCUに対応し、CU4分木のルートノードは概してツリーブロック(またはLCU)に対応する。分割されないRQTのTUはリーフTUと呼ばれる。概して、本開示では、別段に明記されていない限り、リーフCUおよびリーフTUに言及するためにそれぞれCUおよびTUという用語を使用する。

30

### 【0055】

[0052] ビデオシーケンスは、通常、一連のビデオフレームまたはピクチャを含む。ピクチャグループ(GOP)は、一般に、ビデオピクチャのうちの一連の1つまたは複数を備える。GOPは、GOP中に含まれるいくつかのピクチャを記述するシンタックスデータを、GOPのヘッダ中、ピクチャの1つもしくは複数のヘッダ中、または他の場所に含み得る。ピクチャの各スライスは、それぞれのスライスの符号化モードを記述するスライスシンタックスデータを含み得る。図1のビデオエンコーダ20は、ビデオデータを符号化するために個々のビデオスライス内のビデオブロックに対して動作し得る。ビデオブロックは、CU内のコーディングノードに対応し得る。ビデオブロックは、固定サイズまたは可変サイズを有することができ、指定されたコーディング規格に従ってサイズが異なり得る。

40

### 【0056】

[0053] HEVCは、様々なPUサイズにおける予測をサポートする。特定のCUのサイズが $2N \times 2N$ であると仮定すると、HEVCは、 $2N \times 2N$ または $N \times N$ のPUサイズにおけるイントラ予測と、 $2N \times 2N$ 、 $2N \times N$ 、 $N \times 2N$ 、または $N \times N$ の対称なPUサイズにおけるインター予測とをサポートする。HEVCは、また、 $2N \times nU$ 、 $2N \times$

50

$n D$ 、 $n L \times 2 N$ 、および $n R \times 2 N$ のPUサイズでのインター予測のための、非対称な区分をサポートする。非対称区分では、CUの一方向は区分されないが、他の方向は25%および75%に区分される。25%パーティションに対応するCUの部分は、「n」、ならびにそれに続く「Up」、「Down」、「Left」、または「Right」の指示によって示される。したがって、たとえば、「2N × nU」は、上部で $2N \times 0.5N$ PU、および下部で $2N \times 1.5N$ PUに水平に区分される $2N \times 2N$ CUを指す。

#### 【0057】

[0054]本開示では、「 $N \times N$ 」および「 $N \times N (N \text{ by } N)$ 」は、垂直方向の寸法および水平方向の寸法に関するビデオブロックのピクセル寸法、たとえば、 $16 \times 16$ ピクセルまたは $16 \times 16 (16 \text{ by } 16)$ ピクセルを指すために互換的に使用され得る。概して、 $16 \times 16$ ブロックは、垂直方向に $16$ ピクセル( $y = 16$ )、および水平方向に $16$ ピクセル( $x = 16$ )を有する。同様に、 $N \times N$ ブロックは、垂直方向に $N$ ピクセル、水平方向に $N$ ピクセルを有し得、 $N$ が非負の整数値を表す。ブロック中のピクセルは、行および列に配置され得る。さらに、ブロックは、必ずしも、水平方向において垂直方向と同一個数のピクセルを有するとは限らない。たとえば、ブロックは $N \times M$ ピクセルを備え得、ここで、 $M$ は必ずしも $N$ に等しいとは限らない。

10

#### 【0058】

[0055]CUのPUを使用したイントラ予測コーディングまたはインター予測コーディングに続いて、ビデオエンコーダ20は、CUのTUのための残差データを計算し得る。PUは、(ピクセル領域とも呼ばれる)空間領域において予測ピクセルデータを生成する方法またはモードを記述するシンタックスデータを備え得、TUは、変換、たとえば、残差ビデオデータへの離散コサイン変換(DCT)、整数変換、ウェーブレット変換、または概念的に同様の変換の適用後の、変換領域における係数を備え得る。残差データは、符号化されていないピクチャのピクセルと、PUに対応する予測値との間のピクセル差分に対応し得る。ビデオエンコーダ20は、CUのための残差データを含むTUを形成し、次いで、CUのための変換係数を生成するために、TUを変換し得る。

20

#### 【0059】

[0056]変換係数を生成するための変換の後で、ビデオエンコーダ20は変換係数の量子化を実行し得る。量子化は、その最も広義の通常の意味を有することを意図された広義の用語である。一実施形態では、量子化は、さらなる圧縮を提供する、係数を表すのに使用されるデータの量をおそらくは減らすために変換係数が量子化されるプロセスを指す。量子化プロセスは、係数の一部または全部に関連するビット深度を低減し得る。たとえば、 $n$ ビットの値は、量子化中に $m$ ビットの値に切り捨てられ得、ここで、 $n$ は $m$ よりも大きい。

30

#### 【0060】

[0057]量子化の後に、ビデオエンコーダは、変換係数を走査して、量子化された変換係数を含む2次元行列から1次元ベクトルを生成し得る。走査は、アレイの前部により高いエネルギー(したがって、より低い周波数)係数を配置し、アレイの後部により低いエネルギー(したがって、より高い周波数)係数を配置するように設計され得る。いくつかの例では、ビデオエンコーダ20は、エントロピー符号化され得るシリアル化ベクトルを生成するために、量子化された変換係数を走査するためにあらかじめ定義された走査順序を利用し得る。他の例では、ビデオエンコーダ20は、適応スキヤンを実行することができる。1次元ベクトルを形成するために、量子化された変換係数を走査した後、ビデオエンコーダ20は、たとえば、コンテキスト適応型可変長コーディング(CAVLC)、コンテキスト適応型バイナリ算術コーディング(CABAC)、シンタックスベースコンテキスト適応型バイナリ算術コーディング(SBAC)、確率間隔区分エントロピー(PIP-E)コーディング、または別のエントロピー符号化方法に従って、1次元ベクトルをエントロピー符号化し得る。ビデオエンコーダ20はまた、ビデオデータを復号する際にビデオデコーダ30によって使用するための、符号化ビデオデータに関連付けられたシンタッ

40

50

クス要素をエントロピー符号化し得る。

【0061】

[0058]ビデオエンコーダ20は、さらに、ブロックベースのシンタックスデータ、フレームベースのシンタックスデータ、およびGOPベースのシンタックスデータなどのシンタックスデータを、たとえば、フレームヘッダ、ブロックヘッダ、ライスヘッダ、またはGOPヘッダ中で、ビデオエンコーダ30へ送り得る。GOPシンタックスデータは、それぞれのGOP中のいくつかのフレームを記述し得、フレームシンタックスデータは、対応するフレームを符号化するために使用された符号化／予測モードを示し得る。

【0062】

[0059]図2は、本開示で説明する態様による技法を実装し得るビデオエンコーダの一例を示すブロック図である。ビデオエンコーダ20のユニットのうちの1つまたは複数は、本開示の技法のいずれかまたはすべてを実行するように構成され得る。一例として、変換処理ユニット52は、本開示で説明する変換技法のいずれかまたはすべてを実行するように構成され得る。しかしながら、本開示の態様はそのように限定されない。いくつかの例では、本開示で説明する技法は、ビデオエンコーダ20の様々な構成要素間で共有され得る。いくつかの例では、プロセッサ(図示せず)が、本開示で説明する技法のいずれかまたはすべてを実行するように構成され得る。

10

【0063】

[0060]ビデオエンコーダ20は、ビデオスライス内のビデオブロックのイントラコーディングおよびインターフォーマンスを実行し得る。イントラコーディングは、所与のビデオフレームまたはピクチャ内のビデオの空間冗長性を低減または除去するために空間的予測に依拠する。インターフォーマンスは、ビデオシーケンスの隣接するフレームまたはピクチャ内のビデオの時間冗長性を低減または除去するために時間的予測に依拠する。イントラモード(Iモード)は、いくつかの空間ベースコーディングモードのいずれかを指すことがある。単一方向予測(Pモード)または双予測(Bモード)などのインターモードは、いくつかの時間ベースのコーディングモードのいずれかを指し得る。

20

【0064】

[0061]ビデオエンコーダ20は、符号化されるべきビデオフレーム内の現在のビデオブロックを受信し得る。図2の例では、ビデオエンコーダ20は、モード選択ユニット40と、参照フレームメモリ64と、加算器50と、変換処理ユニット52と、量子化ユニット54と、エントロピー符号化ユニット56とを含む。モード選択ユニット40は、動き推定ユニット42と、動き補償ユニット44と、イントラ予測ユニット46と、パーティションユニット48とを含む。ビデオブロック再構成のために、ビデオエンコーダ20はまた、逆量子化ユニット58と、逆変換ユニット60と、加算器62とを含み得る。再構成されたビデオからブロックキネスマティファクトを除去するためにブロック境界をフィルタ処理するデブロッキングフィルタ(図示せず)を含めることもできる。所望される場合、デブロッキングフィルタは、通常、加算器62の出力をフィルタ処理することになる。デブロッキングフィルタに加えて、追加のフィルタ(ループ内またはループ後)も使用され得る。そのようなフィルタは、簡潔のために図示されないが、所望される場合、加算器50の出力を(インループフィルタとして)フィルタ処理し得る。

30

【0065】

[0062]符号化プロセス中に、ビデオエンコーダ20は、コーディングされるべきビデオフレームまたはスライスを受信し得る。フレームまたはスライスは複数のビデオブロックに分割され得る。動き推定ユニット42および動き補償ユニット44は、時間的予測を提供するために、1つまたは複数の参照フレーム内の1つまたは複数のブロックに対して、受信されたビデオブロックのインター予測コーディングを実行し得る。イントラ予測ユニット46は、代替的に、空間的予測を行うために、コーディングされるべきブロックと同じフレームまたはスライス中の1つまたは複数の隣接ブロックに対する受信されたビデオブロックのイントラ予測コーディングを実行し得る。ビデオエンコーダ20は、たとえば、ビデオデータのブロックごとに適当なコーディングモードを選択するために、複数のコ

40

50

ーディングパスを実行し得る。

**【0066】**

[0063]その上、区分ユニット48は、前のコーディングパスでの前の区分方式の評価に基づいて、ビデオデータのブロックをサブブロックに区分し得る。たとえば、区分ユニット48は、最初にフレームまたはスライスをLCUに区分し、レート歪み分析（たとえば、レート歪み最適化）に基づいて、LCUの各々をサブCUに区分し得る。モード選択ユニット40は、サブCUへのLCUの区分を示す4分木データ構造をさらに生成し得る。4分木のリーフノードCUは、1つまたは複数のPUと1つまたは複数のTUとを含み得る。

**【0067】**

[0064]モード選択ユニット40は、たとえば、誤差結果に基づいてコーディングモードのうちの1つ、すなわち、イントラまたはインターを選択し、得られたイントラコーディングまたはインターコーディングされたブロックを、残差ブロックデータを生成するために加算器50に与え、参照フレームとして使用するための符号化されたブロックを再構成するために加算器62に与え得る。モード選択ユニット40はまた、動きベクトル、イントラモードインジケータ、区分情報、および他のそのようなシンタックス情報などの、シンタックス要素をエントロピー符号化ユニット56に与え得る。

**【0068】**

[0065]動き推定ユニット42と動き補償ユニット44とは、高度に統合され得るが、概念的な目的のために別々に示してある。動き推定ユニット42によって実行される動き推定は、ビデオブロックに関する動きを推定する動きベクトルを生成するプロセスである。動きベクトルは、たとえば、現在のフレーム（または他のコード化ユニット）内でコーディングされている現在のブロックに対する参照フレーム（または他のコード化ユニット）内の予測ブロックに対する現在のビデオフレームまたはピクチャ内のビデオブロックのPUの変位を示し得る。予測ブロックは、絶対差分和（SAD：sum of absolute difference）、2乗差分和（SSD：sum of square difference）、または他の差分メトリックによって決定され得るピクセル差分に関して、コーディングされるべきブロックにぴったり一致することがわかるブロックである。いくつかの例では、ビデオエンコーダ20は、参照フレームメモリ64に記憶されている参照ピクチャの、サブ整数ピクセル位置に対する値を計算し得る。たとえば、ビデオエンコーダ20は、参照ピクチャの4分の1ピクセル位置、8分の1ピクセル位置、または他の分数ピクセル位置の値を補間し得る。したがって、動き推定ユニット42は、フルピクセル位置と分数ピクセル位置とに対する動き探索を実行し、分数ピクセル精度で動きベクトルを出力し得る。

**【0069】**

[0066]動き推定ユニット42は、PUの位置を参照ピクチャの予測ブロックの位置と比較することによって、インターコード化スライス中のビデオブロックに関するPUの動きベクトルを計算する。参照ピクチャは、第1の参照ピクチャリスト（リスト0）または第2の参照ピクチャリスト（リスト1）から選択され得、それらの参照ピクチャリストの各々は、参照フレームメモリ64に記憶された1つまたは複数の参照ピクチャを識別する。動き推定ユニット42は、計算された動きベクトルをエントロピー符号化ユニット56と動き補償ユニット44とに送る。

**【0070】**

[0067]動き補償ユニット44によって実行される動き補償は、動き推定ユニット42によって決定された動きベクトルに基づいて、予測ブロックをフェッチまたは生成することを伴い得る。この場合も、いくつかの例では、動き推定ユニット42と動き補償ユニット44とは機能的に統合され得る。現在のビデオブロックのPUの動きベクトルを受信すると、動き補償ユニット44は、動きベクトルが参照ピクチャリストのうちの1つにおいて指す予測ブロックの位置を特定し得る。加算器50は、以下で説明するように、コーディングされている現在のビデオブロックのピクセル値から予測ブロックのピクセル値を減算し、ピクセル差分値を形成することによって、残差ビデオブロックを形成し得る。動き推

10

20

30

40

50

定ユニット42はルーマ成分に対して動き推定を実行し得、動き補償ユニット44は、クロマ成分とルーマ成分の両方のためにルーマ成分に基づいて計算された動きベクトルを使用し得る。モード選択ユニット40はまた、ビデオスライスのビデオブロックを復号する際にビデオデコーダ30が使用するビデオブロックとビデオスライスとに関連するシンタックス要素を生成し得る。シンタックス要素は、ビデオシーケンスレベル、ビデオフレームレベル、ビデオスライスレベル、ビデオC Uレベル、またはビデオP Uレベルのうちの1つまたは複数における予測情報を表し得る。たとえば、動き補償ユニット44は、C U、P U、およびT Uのサイズを含むビデオブロック情報、ならびにイントラモード予測に関する動きベクトル情報を示すシンタックス要素を生成し得る。

## 【0071】

10

[0068]イントラ予測ユニット46は、上記で説明したように、動き推定ユニット42と動き補償ユニット44とによって実行されるインター予測の代替として、現在のブロックをイントラ予測または計算し得る。特に、イントラ予測ユニット46は、現在のブロックを符号化するために使用すべきイントラ予測モードを決定し得る。いくつかの例では、イントラ予測ユニット46は、たとえば別々の符号化パス中に、様々なイントラ予測モードを使用して現在のブロックを符号化し得、イントラ予測ユニット46（または、いくつかの例では、モード選択ユニット40）は、テストされたモードから使用すべき適当なイントラ予測モードを選択し得る。

## 【0072】

20

[0069]たとえば、イントラ予測ユニット46は、様々なテストされたイントラ予測モードのためのレート歪み分析を使用してレート歪み値を計算し、テストされたモードの間で最良のレート歪み特性を有するイントラ予測モードを選択し得る。レート歪み分析は、概して、符号化ブロックと、符号化ブロックを生成するために符号化された元の符号化されていないブロックとの間の歪み（または誤差）の量、ならびに、符号化ブロックを生成するために使用されるビットレート（すなわち、ビット数）を決定する。イントラ予測ユニット46は、どのイントラ予測モードがブロックについて最良のレート歪み値を呈するかを決定するために、様々な符号化ブロックの歪みおよびレートから比を計算し得る。

## 【0073】

30

[0070]ブロックのためのイントラ予測モードを選択した後、イントラ予測ユニット46は、ブロックのための選択されたイントラ予測モードを示す情報をエントロピー符号化ユニット56に与え得る。エントロピー符号化ユニット56は、選択されたイントラ予測モードを示す情報を符号化し得る。ビデオエンコーダ20は、複数のイントラ予測モードインデックステーブルおよび複数の修正されたイントラ予測モードインデックステーブル（コードワードマッピングテーブルとも呼ばれる）と、様々なブロックの符号化コンテキストの定義と、コンテキストの各々について使用すべき、最確イントラ予測モード、イントラ予測モードインデックステーブル、および修正されたイントラ予測モードインデックステーブルの指示とを含み得る構成データを送信ビットストリーム中に含め得る。

## 【0074】

40

[0071]ビデオエンコーダ20は、モード選択ユニット40からの予測データを、コーディングされている元のビデオブロックから減算することによって、残差ビデオブロックを形成する。加算器50は、この減算演算を実行し得る。変換処理ユニット52は、離散コサイン変換（DCT）または概念的には類似の変換などの変換を残差ブロックに適用し、残差変換係数の値を備えるビデオブロックを生成し得る。変換処理ユニット52は、DCTに概念的に類似する他の変換を実行し得る。ウェーブレット変換、整数変換、サブバンド変換または他のタイプの変換が使用され得る。変換処理ユニット52は、次いで、変換を残差ブロックに適用して、残差変換係数のブロックを生成し得る。変換処理ユニット52は、残差情報をピクセル値領域から周波数領域などの変換領域に変換し得る。より具体的には、変換の適用の前に、T Uは、ピクセル領域中に残差ビデオデータを備え得、変換の適用の後、T Uは、周波数領域中に残差ビデオデータを表す変換係数を備え得る。

## 【0075】

50

[0072] 従来、ビデオエンコーダ20は、実装されるビデオ圧縮規格によってサポートされるTUの様々なサイズの各々について別個のコンテキストモデルを維持する。HEVC規格の場合、ビデオ符号化効率を向上させるために、たとえば、 $32 \times 32$ から $128 \times 128$ までの追加の変換ユニットサイズが使用され得るが、追加のTUサイズによって、追加の変換ユニットサイズの各々にコンテキストモデルを維持するためにメモリおよび計算要件も増大する。場合によっては、TUサイズが大きくなるにつれて使用されるコンテキストが多くなることがあり、したがって、より大きいTUサイズ用のより多くのコンテキストを維持するためにメモリおよび計算要件も増大する可能性がある。この問題の影響を低減するために、変換処理ユニット52は、行列乗算中に（「順変換」と呼ばれることがある）変換を簡略化し、そのピット深度を制約することに関する上記および以下で説明する方法（たとえば、以下で、図4～図5に関して説明する内部ピット深度制約方法）のいずれかを実行するようにさらに構成され得る。

10

## 【0076】

[0073] 変換処理ユニット52は、得られた変換係数を量子化ユニット54に送り得る。量子化ユニット54は、次いで、ピットレートをさらに低減するために変換係数を量子化し得る。量子化プロセスは、係数の一部または全部に関連するピット深度を低減し得る。量子化の程度は、量子化パラメータを調整することによって変更され得る。いくつかの例では、量子化ユニット54は、次いで、量子化された変換係数を含む行列の走査を実行し得る。代替的に、エントロピー符号化ユニット56が走査を実行し得る。

20

## 【0077】

[0074] 一例として、変換処理ユニット52は、変換結果を所定のピット深度値（たとえば、16ピットのピット深度または他のピット深度値）に制約し得る。一実装形態では、変換処理ユニット52は、1つまたは複数の内部変換段階において変換結果を制約し得る。ビデオエンコーダ20は、次いで、所定のピット深度値のために最適化された特殊な命令セットを利用し得る。このようにして、変換処理ユニット52は、より速い処理速度を受け得る。このプロセスについて、図4中でさらに説明し、実証する。

20

## 【0078】

[0075] 上記で説明した例では、変換処理ユニット52は、変換結果を16ピット値に制約するように構成される。他の場合には、変換処理ユニット52は、制約された値が満足な結果をもたらすことになると決定するようにさらに構成され得る。このプロセスについて、図5中でさらに説明し、実証する。

30

## 【0079】

[0076] 量子化の後に、エントロピー符号化ユニット56は、量子化変換係数をエントロピーコーディングし得る。たとえば、エントロピー符号化ユニット56は、コンテキスト適応型可変長コーディング(CAVLC)、コンテキスト適応型バイナリ算術コーディング(CABAC)、シンタックスペースコンテキスト適応型バイナリ算術コーディング(SBAC)、確率間隔区分エントロピー(PIPE)コーディング、または別のエントロピー符号化技法を実行し得る。コンテキストベースエントロピー符号化の場合、コンテキストは隣接ブロックに基づき得る。エントロピー符号化ユニット56によるエントロピー符号化の後、符号化されたピットストリームは、別のデバイス（たとえば、ビデオデコーダ30）に送信されるか、または後で送信するかもしくは取り出すためにアーカイブされ得る。

40

## 【0080】

[0077] 逆量子化ユニット58および逆変換ユニット60は、それぞれ逆量子化および逆変換を適用して、たとえば参照ブロックとして後で使用するために、画素領域において残差ブロックを再構成し得る。動き補償ユニット44は、残差ブロックを参照フレームメモリ64のフレームのうちの1つの予測ブロックに加算することによって参照ブロックを計算し得る。動き補償ユニット44はまた、再構成された残差ブロックに1つまたは複数の補間フィルタを適用して、動き推定において使用するサブ整数ピクセル値を計算し得る。加算器62は、参照フレームメモリ64に記憶するために再構成されたビデオブロックを

50

生成するのに、再構成された残差ブロックを、動き補償ユニット44によって生成された動き補償予測ブロックに加算し得る。再構成されたビデオブロックは、次いで、後続のビデオフレーム中のブロックをインターフォーメーションするための参照ブロックとして動き推定ユニット42および動き補償ユニット44によって使用され得る。

#### 【0081】

[0078]図3は、本開示で説明する態様による技法を実装し得るビデオデコーダの一例を示すブロック図である。本開示で説明する技法は、ビデオデコーダ30の様々な構成要素を利用し得る。いくつかの例では、プロセッサ(図示せず)が、技法のいずれかまたはすべてを実行するように構成され得る。

#### 【0082】

[0079]図3の例では、ビデオデコーダ30は、エントロピー復号ユニット70と、動き補償ユニット72とイントラ予測ユニット74とをさらに含む予測ユニット81と、逆量子化ユニット76と、逆変換ユニット78と、参照ピクチャ(フレーム)メモリ82と、加算器80とを含む。ビデオデコーダ30は、ビデオエンコーダ20(たとえば、図1および図2参照)に関して説明した符号化パスとは概して逆の復号パスを実行し得る。動き補償ユニット72は、エントロピー復号ユニット70から受信された動きベクトルに基づいて、予測データを生成し得、一方、イントラ予測ユニット74は、エントロピー復号ユニット70から受信されたイントラ予測モードインジケータに基づいて、予測データを生成し得る。

#### 【0083】

[0080]従来、ビデオエンコーダ30は、実装されるビデオ圧縮規格によってサポートされるTUの様々なサイズの各々について別個のコンテキストモデルを維持することになる。HEVC規格の場合、ビデオ符号化効率を向上させるために、たとえば、 $32 \times 32$ から $128 \times 128$ までの追加の変換ユニットサイズが利用され得るが、追加のTUサイズによって、追加の変換ユニットサイズの各々にコンテキストモデルを維持するためにメモリおよび計算要件も増大する。

#### 【0084】

[0081]復号プロセス中に、ビデオデコーダ30は、ビデオエンコーダ20から、符号化されたビデオスライスのビデオブロックと、関連するシンタックス要素とを表す、符号化されたビデオビットストリームを受信し得る。ビデオデコーダ30のエントロピー復号ユニット70は、量子化された係数と、動きベクトルまたはイントラ予測モードインジケータと、他のシンタックス要素とを生成するために、ビットストリームをエントロピー復号する。エントロピー復号ユニット70は、次いで、動きベクトルと他のシンタックス要素とを動き補償ユニット72に転送し得る。ビデオデコーダ30は、ビデオスライスレベルおよび/またはビデオブロックレベルで構文要素を受信し得る。

#### 【0085】

[0082]ビデオスライスが、イントラコード化(I)スライスとしてコーディングされるとき、イントラ予測ユニット74は、現在のフレームまたはピクチャの以前に復号されたブロックから、シグナリングされたイントラ予測モードおよびデータに基づいて、現在のビデオスライスのビデオブロックのための予測データを生成し得る。ビデオフレームが、インターフォーメーションされた(たとえば、B、P、またはGPB)スライスとしてコーディングされる時に、動き補償ユニット72は、エントロピー復号ユニット70から受け取られた動きベクトルと他の構文要素とに基づいて、現在のビデオスライスのビデオブロックの予測ブロックを生成し得る。予測ブロックは、参照ピクチャリストのうちの1つ内の参照ピクチャのうちの1つから生成され得る。ビデオデコーダ30は、参照ピクチャ(フレーム)メモリ82に記憶された参照ピクチャに基づいて、デフォルト構成技法を使用して、参照フレームリスト、すなわち、リスト0とリスト1とを構成し得る。動き補償ユニット72は、動きベクトルと他のシンタックス要素とをパースすることによって現在のビデオスライスのビデオブロックのための予測情報を決定し、復号されている現在のビデオブロックのための予測ブロックを生成するために予測情報を使用し得る。たとえば、動き

10

20

30

40

50

補償ユニット72は、ビデオスライスのビデオブロックをコーディングするために使用される予測モード（たとえば、イントラ予測またはインター予測）と、インター予測スタイル（たとえば、Bスライス、Pスライス、またはGPBスライス）と、スライスの参照ピクチャリストのうちの1つまたは複数のための構築情報と、スライスの各インター符号化ビデオブロックのための動きベクトルと、スライスの各インターコード化ビデオブロックのためのインター予測ステータスと、現在のビデオスライス中のビデオブロックを復号するための他の情報とを決定するために、受信されたシンタックス要素のいくつかを使用し得る。

#### 【0086】

[0083]動き補償ユニット72は、補間フィルタに基づいて補間を実行し得る。動き補償ユニット72は、参照ブロックのサブ整数ピクセルの補間された値を計算するために、ビデオブロックの符号化の間にビデオエンコーダ20によって使用された補間フィルタを使用し得る。この場合、動き補償ユニット72は、受信したシンタックス要素から、ビデオエンコーダ20によって使用された補間フィルタを決定し、予測ブロックを生成するために、その補間フィルタを使用し得る。10

#### 【0087】

[0084]逆量子化ユニット76は、ビットストリーム中で与えられ、エントロピー復号ユニット70によって復号された量子化変換係数を逆量子化（*inverse quantize*）、たとえば、逆量子化（*de-quantize*）し得る。逆量子化処理は、量子化の程度、および同様に、適用されるべき逆量子化の程度を決定するために、ビデオスライスの中の各ビデオブロックに対してビデオデコーダ30によって計算された量子化パラメータQPYの使用を含み得る。20

#### 【0088】

[0085]逆変換ユニット78は、ピクセル領域内の残差ブロックを生成するために、逆変換、たとえば、逆DCT、逆整数変換、または概念的には類似の逆変換プロセスを、変換係数に適用し得る。動き補償ユニット72が、動きベクトルおよび他のシンタックス要素に基づいて現在のビデオブロックのための予測ブロックを生成した後、ビデオデコーダ30は、逆変換ユニット78からの残差ブロックを動き補償ユニット72によって生成された対応する予測ブロックと加算することによって、復号ビデオブロックを形成し得る。加算器80は、この加算演算を実行し得る。ブロッキネスアーティファクトを除去するために、復号ブロックをフィルタ処理するためのデブロッキングフィルタも適用され得る。（コーディンググループ中またはコーディンググループ後のいずれかの）他のループフィルタも、ピクセルの遷移を平滑化し、または場合によっては、ビデオ品質を改善するために使用され得る。所与のフレームまたはピクチャ中の復号ビデオブロックは、次いで、参照ピクチャ（フレーム）メモリ82に記憶され得、この参照ピクチャ（フレーム）メモリ82は、後続の動き補償のために使用される参照ピクチャを記憶し得る。参照ピクチャ（フレーム）メモリ82はまた、図1のディスプレイデバイス32などのディスプレイデバイス上に後で提示するために復号ビデオを記憶し得る。30

#### 【0089】

[0086]図4に、たとえば、メッシュベースの計算を使用した低複雑な順変換の方法400のフローチャートを示す。方法400は、プロセッサまたは、たとえば、図2のエンコーダ20などのエンコーダによって実行され得る。一実施形態では、図2のエンコーダ20の変換処理ユニット52は、方法400を実行するように構成され得る。本方法の様々なブロックが、変換処理ユニットによって実行されるものとして説明したが、本方法の様々なブロックが、他のプロセッサ、エンコーダ、またはそれらのユニットによって実行され得ることを理解されたい。40

#### 【0090】

[0087]方法400により、エンコーダは、16ビットのデータのために最適化された特殊な命令セット（たとえば、ARMアーキテクチャ、Advanced SIMD（NEON）、デジタル信号処理（DSP）など）を使用して16ビット演算を実行することを50

可能にするために変換結果を 16 ビット値に制約することが可能になる。上記で説明したように、ビット深度を制約することは、コーディング効率を改善し、ビデオ符号化中の計算リソース要件を低減し得る。しかしながら、場合によっては、ビット深度を制約することはまた、変換出力の品質を減少させ得る。したがって、16 ビット演算を実行する前に、エンコーダ（たとえば、エンコーダ 20 の変換処理ユニット 52）は、制約された値が満足な結果をもたらすことになると決定し、それに応じて調整するようにさらに構成され得、これについて、図 5 に関してさらに説明する。

#### 【0091】

[0088] 方法 400 は、ブロック 405において開始する。次いで、ブロック 410において、変換処理ユニット 52 は、フルサイズの順方向変換行列を複数の、あまり複雑でない段階に分解し得る（たとえば、多段変換行列を決定し得る）。分解は、メッシュベースの分解方法を使用することによって行われ得る。一実装形態では、フルサイズの変換行列を分解するために計算リソースを使用するのではなく、変換処理ユニット 52 は、代わりに、メモリまたはコード化ビットストリームから複数の段階を取り出し得る。いくつかの実装形態では、最初の順変換行列は、 $32 \times 32$  の変換行列であり得る。他の実装形態では、変換行列は、別のサイズであり得る。

10

#### 【0092】

[0089] 次いで、ブロック 415において、変換処理ユニット 52 は、複数の段階の各々に対して適切な変換演算を実行し得る。場合によっては、変換演算が実行された後、これらの段階のうちの 1 つまたは複数は、16 ビットよりも大きい変換結果を含み得る。これらの変換結果は、16 ビットよりも大きいシステムのために設計された内部計算動作（たとえば、32 ビット演算）を必要とし得、これは、変換処理ユニット 52 の効率を減少させ得る。

20

#### 【0093】

[0090] したがって、ブロック 420において、変換処理ユニット 52 は、複数の段階の各々における変換結果の内部ビット深度を 16 ビット値に制約し得る。変換処理ユニット 52 は、飽和論理を使用して、たとえば、内部値を 16 ビット値にクリッピングすることによってこの制約されたプロセスを実行し得る。より詳細には、変換処理ユニット 52 は、変換出力ビット深度を制約するだけでなく、変換演算中に内部でビット深度を制約し得る。これを達成するために、変換処理ユニット 52 は、各中間変換段階の出力においてクリッピング関数を実行し得る。一実装形態では、変換処理ユニット 52 は、制約されたプロセスを実行するために付録 A に示すコードに付加されるクリッピング関数を使用し得、ここで、 $a$  および  $n$  は、16 ビットの範囲内に制約される和とともに追加されるべき値を表す。1つの例示的なクリッピング関数は、 $y = \text{CLIP}_3(x, \min_{val}, \max_{val})$  として表され得る。この例示的な機能では、 $x$  は、バイト数を表し得る。 $x$  が  $\min_{val}$  よりも小さい場合、 $y$  は  $\min_{val}$  に設定され得る。さらに、 $x$  が  $\max_{val}$  よりも大きい場合、 $y$  は  $\max_{val}$  に設定され得る。 $x$  が  $\min_{val}$  と  $\max_{val}$  との範囲の間にいる場合、 $y$  は、 $x$  に設定され得る。より具体的な例示的なクリッピング関数は、 $(a + b)_{sat} = \text{CLIP}_3(-32768, 32767, a + b)$  または  $(a_0 * b_0 + a_1 * b_1 + \dots)_{sat} = \text{CLIP}_3(-32768, 32767, a_0 * b_0 + a_1 * b_1 + \dots)$  のいずれかのうちの 1 つとして表され得、これは、上記のクリッピング例と同様の論理ツリーに従う。クリッピング関数の他の構造は、制約プロセスを実行するために実装され得る。いずれの場合も、加算演算、減算演算、および / または乗算および加算演算について、飽和論理は、結果を 16 ビットに制約し得る。

30

#### 【0094】

[0091] 各段階において内部ビット深度を 16 ビットに制約することによって、変換処理ユニット 52 は、そのレベルのために計算効率の良い命令セットを利用することが可能になり得る。たとえば、変換に対する入力ビット深度は 9 ビットであり得、開始内部ビット深度は、16 ビットよりも大きくなり得る。一例では、変換処理ユニット 52 は、特定のビットレベルでの演算（たとえば、16 ビット演算、32 ビット演算、または 64 ビット

40

50

演算など)しか実行することが可能でないことがある。したがって、内部ビット深度が16ビットよりも大きいとき、変換処理ユニット52は、少なくとも32ビットレベルで内部演算を実行する必要があり得る。この結果を回避するために、変換処理ユニット52は、(たとえば、32ビットの内部演算が必要とされないように)変換の各分解された段階の内部ビット深度を16ビットに制約し得る。言い換えれば、各段階の内部ビット深度が16ビットに制約されるので、プロセッサは、16ビット演算とともに使用するために特に設計された計算効率の良い命令セット(たとえば、ARMアーキテクチャ、Advanced SIMD(NEON)、デジタル信号処理(DSP)など)を利用し得る。

#### 【0095】

[0092]次いで、随意のブロック425において、変換処理ユニット52は、随意に、各段階における制約された値の偏差(たとえば、誤差)をしきい値と比較することによって制約された値が満足な結果をもたらすことになるかどうかを決定し得る。たとえば、方法500は、ブロック425において、図5に関して以下で説明する方法500のブロック510および520を実行することによって制約された値が満足な結果をもたらすことになるかどうかを決定し得る。制約された値が満足な結果をもたらすことになる場合、方法400は、ブロック430に続く。そうでない場合、ブロック427において、変換処理ユニット52は、(たとえば、方法500のブロック530に関して以下で説明するように)偏差がしきい値よりも低くなるまで係数のサブセットを再計算し得る。この方法500により、変換処理ユニット52は、ビデオ品質をも保持しながら、コーディング効率を増加することが可能になり得る。制約された値が満足な結果をもたらすことになると変換処理ユニット52が決定すると、次いで、ブロック430において、変換処理ユニット52は、16ビットのデータのために最適化された特殊な命令セット(たとえば、ARMアーキテクチャ、Advanced SIMD(NEON)、デジタル信号処理(DSP)など)を使用して16ビット演算を変換データに対して実行することを続け得る。ブロック490において、本方法は終了する。

#### 【0096】

[0093]図5に、低複雑な順変換のための方法500のフローチャートを示す。方法500は、プロセッサまたは、図2に関して上記で説明したエンコーダ20などのエンコーダによって実行され得る。一実施形態では、エンコーダの変換処理ユニット(たとえば、図2のエンコーダ20の変換処理ユニット52)は、方法500を実行するために使用され得る。実際、方法500が、エンコーダ20の変換処理ユニット52によって実行されるものとして記述されているが、方法500が、エンコーダの異なるプロセッサ、エンコーダ、または処理ユニットによって実行され得ることを理解されたい。一実施形態では、方法500は、最初に、制約する値を(たとえば、図4の方法からの制約された値)が満足な結果をもたらすことになるかどうかを決定し、それに応じて調整する。図4に関して上記で説明したように、変換処理ユニット52が内部ビット深度を制約すると、変換出力の品質をも低下させ得る。図5に関して説明する方法は、偏差が所定のしきい値を下回るまで、品質の低下がどの程度まで発生し得るのかを決定し、それに応じて調整する。

#### 【0097】

[0094]方法500は、ブロック505において開始する。方法500の始めに、変換処理ユニット52は、フルサイズの順方向変換行列を複数の、あまり複雑でない段階にすでに分解している。たとえば、順変換行列は、図4に関して説明した方法に従って分解されていることがある。各段階の内部ビット深度は、図4に関してさらに説明するように、特定のビット深度値(たとえば、16ビットのビット深度値)にすでに制約されていることがある。

#### 【0098】

[0095]次いで、ブロック510において、変換処理ユニット52は、制約された値(たとえば、最終係数)の元の値からの偏差を決定し得る。一例として、飽和論理が使用されるとき、最終係数値は、元の値から逸脱し得る(たとえば、あるレベルの誤差を含み得る)。

10

20

30

40

50

## 【0099】

[0096] 偏差を計算した後に、ブロック520において、変換処理ユニット52は、偏差が所定のしきい値制限よりも大きいかどうかを決定し得る。そうである場合、方法500はブロック530に進む。ブロック530において、変換処理ユニット52は、(たとえば、ACレベルの係数のサブセットを再計算するために変換を部分的に適用することによって)係数のサブセットを再計算し、制約された値の同じサブセットを再計算された結果と置き換える。このプロセスは、計算の要件(たとえば、サイクル)および複雑さを最小化しながら、最終係数値が有する、元の値からの偏差を補正し得る。一実装形態では、DCレベルの係数がACレベルの係数よりも大きいダイナミックレンジを有するので、係数のサブセットは、DCレベルの係数のみを含み得、したがって、それらは、元の値から逸脱する可能性が高い。他の実装形態では、増加精度をさらに高めるために、係数のサブセットは、DCレベルの係数を含んでいることに加えて、DCレベルに近い値をもつACレベルの係数を含み得る。

10

## 【0100】

[0097] 上記で説明したように係数の小さいサブセットを再計算した後に、変換処理ユニットは、次いで、ブロック510に戻り、偏差をしきい値と再び照合し得る。(ブロック520において)偏差が所定のしきい値耐久制限よりも小さいと変換処理ユニット52が決定すると、方法500は、ブロック590に進み、終了する。ブロック590において、変換処理ユニット52は、制約された値が満足な結果をもたらすことになると決定し、係数の再計算されたサブセットを与え得る。変換処理ユニット52は、次いで、図4に関して上記で説明した、方法400のブロック427において係数の再計算されたサブセットを利用し得る。

20

## 【0101】

[0098] 上記の方法のすべてについて、N×Nの変換に関して説明した。しかしながら、本開示で説明する方法は、N×Nの変換に限定されない。本方法はまた、変換における次元数にかかわらず、任意のサイズの変換に関して実施され得る。

20

## 【0102】

[0099] 上記例に応じて、本明細書で説明した技法のいずれかのいくつかの行為またはイベントが、異なるシーケンスで実行され得、全体的に追加、マージ、または除外され得る(たとえば、すべての説明した行為またはイベントが本技法の実施のために必要であるとは限らない)ことを認識されたい。さらに、いくつかの例では、行為またはイベントは、連続的にではなく、たとえば、マルチスレッド処理、割込み処理、または複数のプロセッサを通して同時に実行され得る。

30

## 【0103】

[00100] 1つまたは複数の例では、説明する機能は、ハードウェア、ソフトウェア、ファームウェア、またはその任意の組合せで実施され得る。ソフトウェアで実装される場合、機能は、1つまたは複数の命令またはコードとして、コンピュータ可読媒体上に記憶されるか、あるいはコンピュータ可読媒体を介して送信され、ハードウェアベースの処理ユニットによって実行され得る。コンピュータ可読媒体は、データ記憶媒体などの有形媒体に対応する、コンピュータ可読記憶媒体を含み得るか、または、たとえば、通信プロトコルに従って、ある場所から別の場所へのコンピュータプログラムの転送を可能にする任意の媒体を含む通信媒体を含み得る。このようにして、コンピュータ可読媒体は、一般に、(1)非一時的である有形のコンピュータ可読記憶媒体、または(2)信号または搬送波のような通信媒体に対応し得る。データ記憶媒体は、本開示で説明した技法の実装のための命令、コードおよび/またはデータ構造を取り出すために、1つまたは複数のコンピュータあるいは1つまたは複数のプロセッサによってアクセスされ得る、任意の利用可能な媒体であり得る。コンピュータプログラム製品はコンピュータ可読媒体を含み得る。

40

## 【0104】

[00101] 限定ではなく例として、そのようなコンピュータ可読記憶媒体は、RAM、ROM、EEPROM(登録商標)、CD-ROMもしくは他の光ディスクストレージ、磁

50

気ディスクストレージ、もしくは他の磁気ストレージデバイス、フラッシュメモリ、または、命令もしくはデータ構造の形態の所望のプログラムコードを記憶するために使用され得、コンピュータによってアクセスされ得る任意の他の媒体を備え得る。また、いかなる接続もコンピュータ可読媒体と適切に呼ばれる。たとえば、命令が、ウェブサイト、サーバ、または他のリモートソースから、同軸ケーブル、光ファイバーケーブル、ツイストペア、デジタル加入者回線（D S L）、または赤外線、無線、およびマイクロ波などのワイヤレス技術を使用して送信される場合、同軸ケーブル、光ファイバーケーブル、ツイストペア、D S L、または赤外線、無線、およびマイクロ波などのワイヤレス技術は、媒体の定義に含まれる。しかしながら、コンピュータ可読記憶媒体およびデータ記憶媒体は、接続、搬送波、信号、または他の一時的媒体を含むのではなく、非一時的な有形の記憶媒体を対象とすることを理解されたい。本明細書で使用するディスク（disk）およびディスク（disc）は、コンパクトディスク（disc）（C D）、レーザーディスク（登録商標）（disc）、光ディスク（disc）、デジタル多用途ディスク（disc）（D V D）、フロッピー（登録商標）ディスク（disk）およびB l u - r a yディスク（disc）を含み、ここで、ディスク（disk）は、通常、データを磁気的に再生し、ディスク（disc）は、データをレーザーで光学的に再生する。上記の組合せも、コンピュータ可読媒体の範囲内に含めるべきである。

10

## 【0105】

[00102]命令は、1つもしくは複数のデジタル信号プロセッサ（D S P）、汎用マイクロプロセッサ、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）、あるいは他の同等の集積回路またはディスクリート論理回路などの1つもしくは複数のプロセッサによって実行され得る。したがって、本明細書で使用される「プロセッサ」という用語は、前述の構造、または、本明細書で説明された技法の実装に好適な任意の他の構造のいずれかを指すことがある。加えて、いくつかの態様では、本明細書で説明された機能は、符号化および復号のために構成されるか、または複合コーデックに組み込まれる、専用のハードウェアモジュールおよび／またはソフトウェアモジュール内で提供され得る。また、本技法は、1つまたは複数の回路または論理要素で十分に実装され得る。

20

## 【0106】

[00103]本開示の技法は、ワイヤレスハンドセット、集積回路（I C）またはI Cのセット（たとえば、チップセット）を含む、多種多様なデバイスまたは装置で実装され得る。本開示では、開示する技法を実行するように構成されたデバイスの機能的態様を強調するために様々な構成要素、モジュール、またはユニットについて説明したが、それらの構成要素、モジュール、またはユニットを、必ずしも異なるハードウェアユニットによって実現する必要があるとは限らない。むしろ、上記で説明されたように、様々なユニットが、好適なソフトウェアおよび／またはファームウェアとともに、上記で説明された1つまたは複数のプロセッサを含めて、コーデックハードウェアユニットにおいて組み合わされるか、または相互動作ハードウェアユニットの集合によって提供され得る。

30

## 【0107】

[00104]様々な例について説明した。これらおよび他の例は、以下の特許請求の範囲に含まれる。

40

## 【0108】

付録A：16×16順変換のメッシュベースの実装の例

```

for (j=0; j<16; j++)
{
    /* E and O */
    for (k=0;k<8;k++)
    {
        nE[k] = pSrc[k] + pSrc[15-k];
        nO[k] = pSrc[k] - pSrc[15-k];
    }
}

```

50

```

}

/* EE and EO */
for (k=0;k<4;k++)
{
    nEE[k] = nE[k] + nE[7-k];
    nEO[k] = nE[k] - nE[7-k];
}
/* EEE and EEO */
nEEE[0] = nEE[0] + nEE[3];
nEO[0] = nEE[0] - nEE[3];
nEEE[1] = nEE[1] + nEE[2];
nEO[1] = nEE[1] - nEE[2];

pDst[ 0 ] = (anTransCoef16[ 0 ][0]*nEEE[0] + anTransCoef16[ 0 ][1]*nEEE[1] + 4)>>3;
pDst[ 8 ] = (anTransCoef16[ 8 ][0]*nEEE[0] + anTransCoef16[ 8 ][1]*nEEE[1] + 4)>>3;
pDst[ 4 ] = (anTransCoef16[ 4 ][0]*nEO[0] + anTransCoef16[ 4 ][1]*nEO[1] + 4)>>3;
pDst[ 12 ] = (anTransCoef16[12][0]*nEO[0] + anTransCoef16[12][1]*nEO[1] + 4)>>3; 20

for (k=2;k<16;k+=4)
{
    pDst[ k ] = (anTransCoef16[k][0]*nEO[0] + anTransCoef16[k][1]*nEO[1] +
+ anTransCoef16[k][2]*nEO[2] + anTransCoef16[k][3]*nEO[3] + 4)>>3;
}

for (k=1;k<16;k+=2) 30
{
    pDst[ k ] = (anTransCoef16[k][0]*nO[0] + anTransCoef16[k][1]*nO[1] +
+ anTransCoef16[k][2]*nO[2] + anTransCoef16[k][3]*nO[3] +
+ anTransCoef16[k][4]*nO[4] + anTransCoef16[k][5]*nO[5] + anTransCoef16[k][6]*nO[6] + anTransCoef16[k][7]*nO[7] + 4)>>3;
}

pSrc += 16;
pDst += 16;

}

pSrc = pCoef;
pDst = pRes;
for (j=0; j<16; j++)
{
    /* E and O*/
    for (k=0;k<8;k++)
    {
        nE[k] = pSrc[k*16] + pSrc[(15-k)*16];
        nO[k] = pSrc[k*16] - pSrc[(15-k)*16]; 50
    }
}

```

```

}

/* EE and EO */
for (k=0;k<4;k++)
{
    nEE[k] = nE[k] + nE[7-k];
    nEO[k] = nE[k] - nE[7-k];
}
/* EEE and EEO */
nEEE[0] = nEE[0] + nEE[3];
nEEO[0] = nEE[0] - nEE[3];
nEEE[1] = nEE[1] + nEE[2];
nEEO[1] = nEE[1] - nEE[2];

    pDst[ 0      ] = (anTransCoef16[ 0][0]*nEEE[0] + anTransCoef16[ 0][1]*nE
EE[1] + 512)>>10;
    pDst[ 8*16  ] = (anTransCoef16[ 8][0]*nEEE[0] + anTransCoef16[ 8][1]*nE
EE[1] + 512)>>10;
    pDst[ 4*16  ] = (anTransCoef16[ 4][0]*nEEO[0] + anTransCoef16[ 4][1]*nE
EO[1] + 512)>>10;
    pDst[ 12*16 ] = (anTransCoef16[12][0]*nEEO[0] + anTransCoef16[12][1]*nE
EO[1] + 512)>>10; 20

    for (k=2;k<16;k+=4)
    {
        pDst[ k*16 ] = (anTransCoef16[k][0]*nEO[0] + anTransCoef16[k][1]*nEO
[1] + anTransCoef16[k][2]*nEO[2] + anTransCoef16[k][3]*nEO[3] + 512)>>10;
    }

    for (k=1;k<16;k+=2) 30
    {
        pDst[ k*16 ] = (anTransCoef16[k][0]*nO[0] + anTransCoef16[k][1]*nO[1]
+ anTransCoef16[k][2]*nO[2] + anTransCoef16[k][3]*nO[3] +
        anTransCoef16[k][4]*nO[4] + anTransCoef16[k][5]*nO[5] + anTransCoef16[k][6]*nO[6] + anTransCoef16[k][7]*nO[7] + 512)>>10;
    }

    pSrc++;
    pDst++;
}

}

```

【図1】

図1

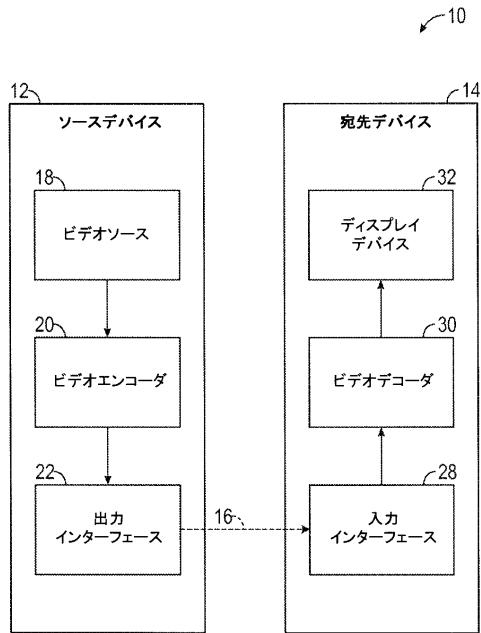


FIG. 1

【図2】

図2

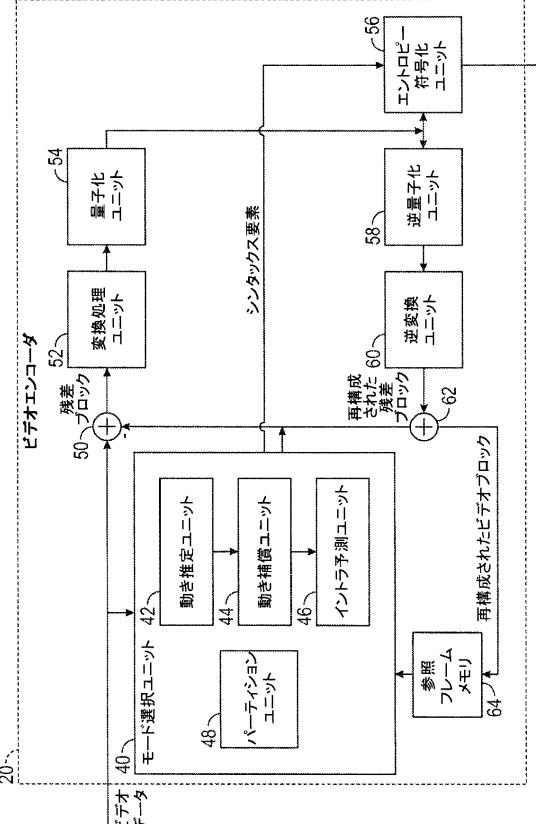


FIG. 2

【図3】

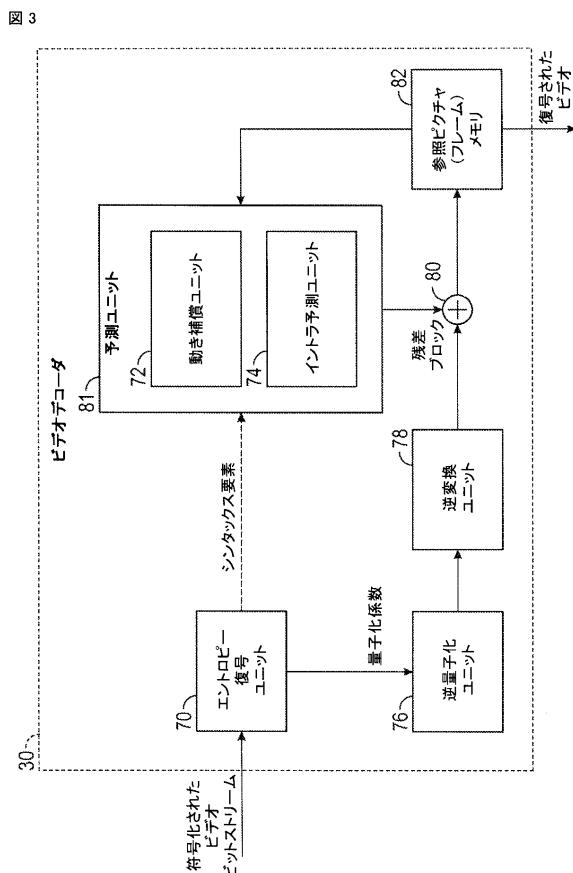


FIG. 3

【図4】

図4

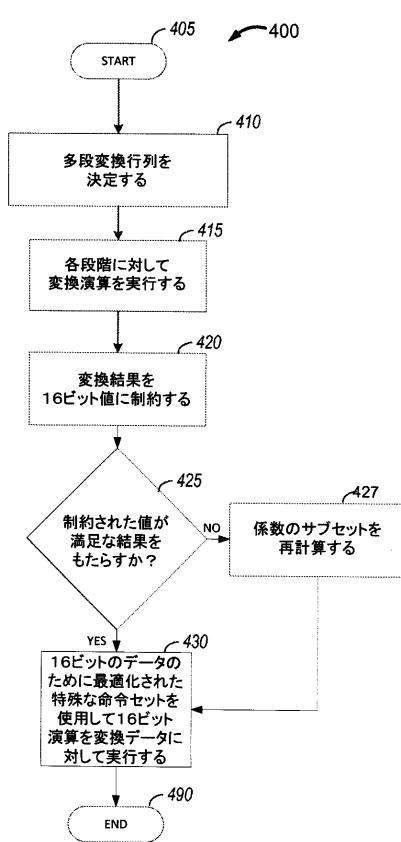


FIG. 4

【図5】

図5

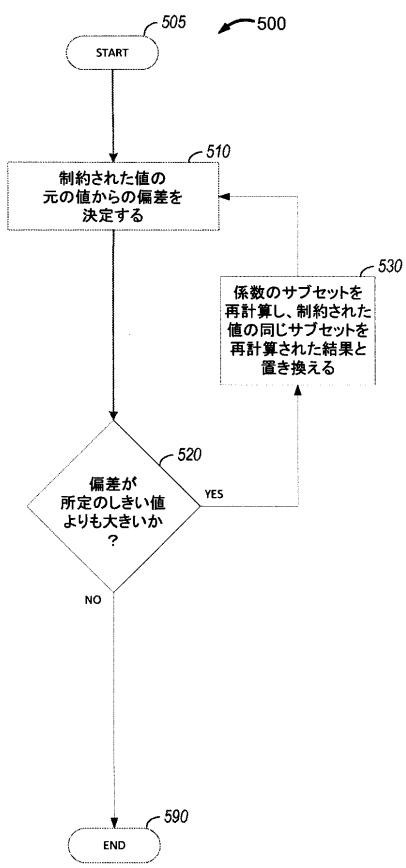


FIG. 5

## 【手続補正書】

【提出日】平成28年11月18日(2016.11.18)

## 【手続補正1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

## 【補正の内容】

## 【特許請求の範囲】

## 【請求項1】

ビデオデータのブロックを記憶するように構成されたメモリと、

前記メモリと通信しているプロセッサであって、

ピクセル領域から係数領域に前記ビデオデータのブロックを変換するための変換を複数の変換段階に分解することと、

前記複数の変換段階を前記ビデオデータのブロックに適用することによって、前記ピクセル領域から前記係数領域に前記ビデオデータのブロックを変換し、各変換段階における出力係数のセットを生成することと、

各変換段階における前記出力係数のセットを所定のビット深度に制約することと、

前記制約された係数のセットと各変換段階における前記出力係数のセットとの間の偏差を決定することと、

前記決定された偏差が所定のしきい値よりも大きいとき、変換段階の前記出力係数のセットのサブセットを再計算することと、

前記複数の変換段階の最後の段階の前記制約された出力係数に対して演算を実行することと、ここにおいて、前記演算の前記ビット深度が、前記所定のビット深度に等しい、

を行うように構成されたプロセッサと

を備える、ビデオエンコーダ。

**【請求項 2】**

前記プロセッサが、メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分解するようにさらに構成された、請求項1に記載のビデオエンコーダ。

**【請求項 3】**

前記プロセッサは、クリッピングされた係数が必ず所定の範囲内に入るよう前に記係数をクリッピングすることによって各変換段階における前記出力係数のセットを制約するようさらに構成された、請求項1に記載のビデオエンコーダ。

**【請求項 4】**

前記所定のビット深度が16ビットである、請求項1に記載のビデオエンコーダ。

**【請求項 5】**

前記偏差が、前記制約された係数のセットと前記出力係数のセットとの間の前記差を備える、請求項1に記載のビデオエンコーダ。

**【請求項 6】**

前記プロセッサが、少なくとも係数の前記再計算されたサブセットを使用して前記ビデオデータのブロックを変換するようにさらに構成された、請求項1に記載のビデオエンコーダ。

**【請求項 7】**

ビデオを符号化する方法であって、

ビデオデータのブロックを記憶することと、

ピクセル領域から係数領域に前記ビデオデータのブロックを変換するための変換を複数の変換段階に分解することと、

前記複数の変換段階を前記ビデオデータのブロックに適用することによって、前記ピクセル領域から前記係数領域に前記ビデオデータのブロックを変換し、各変換段階における出力係数のセットを生成することと、

各変換段階における前記係数出力のセットを所定のビット深度に制約することと、

前記制約された係数のセットと各変換段階における前記出力係数のセットとの間の偏差を決定することと、

前記決定された偏差が所定のしきい値よりも大きいとき、変換段階の前記出力係数のセットのサブセットを再計算することと、

前記複数の段階の最後の段階の前記制約された出力係数に対して演算を実行することと、ここにおいて、前記演算の前記ビット深度が、前記所定のビット深度に等しい、を備える方法。

**【請求項 8】**

メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分解することをさらに備える、請求項7に記載の方法。

**【請求項 9】**

クリッピングされた係数が必ず所定の範囲内に入るよう前に記係数をクリッピングすることによって各変換段階における前記出力係数のセットを制約することをさらに備える、請求項7に記載の方法。

**【請求項 10】**

前記所定のビット深度が16ビットである、請求項7に記載の方法。

**【請求項 11】**

前記偏差が、前記制約された係数のセットと前記出力係数のセットとの間の前記差を備える、請求項7に記載の方法。

**【請求項 12】**

少なくとも係数の前記再計算されたサブセットを使用して前記ビデオデータのブロックを変換することをさらに備える、請求項7に記載の方法。

**【請求項 13】**

実行されたとき、装置に、

ビデオデータのブロックを記憶することと、  
ピクセル領域から係数領域に前記ビデオデータのブロックを変換するための変換を複数の変換段階に分解することと、

前記複数の変換段階を前記ビデオデータのブロックに適用することによって、前記ピクセル領域から前記係数領域に前記ビデオデータのブロックを変換し、各変換段階における出力係数のセットを生成することと、

各変換段階における前記出力係数のセットを所定のビット深度に制約することと、

前記制約された係数のセットと各変換段階における前記出力係数のセットとの間の偏差を決定することと、

前記決定された偏差が所定のしきい値よりも大きいとき、変換段階の前記出力係数のセットのサブセットを再計算することと、

前記複数の変換段階の最後の段階の前記制約された出力係数に対して演算を実行することと、ここにおいて、前記演算の前記ビット深度が、前記所定のビット深度に等しい、を行わせるコードを備える、非一時的コンピュータ可読媒体。

#### 【請求項 1 4】

実行されたとき、前記装置に、メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分解することを行わせるコードをさらに備える、請求項 1 3 に記載のコンピュータ可読媒体。

#### 【請求項 1 5】

実行されたとき、前記装置に、クリッピングされた係数が必ず所定の範囲内に入るよう前記係数をクリッピングすることによって各変換段階における前記出力係数のセットを制約することを行わせるコードをさらに備える、請求項 1 3 に記載のコンピュータ可読媒体。

#### 【請求項 1 6】

前記偏差が、前記制約された係数のセットと前記出力係数のセットとの間の前記差を備える、請求項 1 3 に記載のコンピュータ可読媒体。

#### 【請求項 1 7】

実行されたとき、前記装置に、少なくとも係数の前記再計算されたサブセットを使用して前記ビデオデータのブロックを変換することを行わせるコードをさらに備える、請求項 1 3 に記載のコンピュータ可読媒体。

#### 【請求項 1 8】

ビデオを符号化するための装置であって、  
ビデオデータのブロックを記憶するための手段と、  
ピクセル領域から係数領域に前記ビデオデータのブロックを変換するための変換を複数の変換段階に分解するための手段と、

前記複数の変換段階を前記ビデオデータのブロックに適用することによって、前記ピクセル領域から前記係数領域に前記ビデオデータのブロックを変換し、各変換段階における出力係数のセットを生成するための手段と、

各変換段階における前記出力係数のセットを所定のビット深度に制約するための手段と、

前記制約された係数のセットと各変換段階における前記出力係数のセットとの間の偏差を決定するための手段と、

前記決定された偏差が所定のしきい値よりも大きいとき、変換段階の前記出力係数のセットのサブセットを再計算するための手段と、

前記複数の変換段階の最後の段階の前記制約された出力係数に対して演算を実行するための手段と、ここにおいて、前記演算の前記ビット深度が、前記所定のビット深度に等しい、

を備える、装置。

#### 【請求項 1 9】

メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分

解するための手段をさらに備える、請求項1 8に記載の装置。

【請求項 2 0】

クリッピングされた係数が必ず所定の範囲内に入るよう前に前記係数をクリッピングすることによって各変換段階における前記出力係数のセットを制約するための手段をさらに備える、請求項1 8に記載の装置。

【請求項 2 1】

前記偏差が、前記制約された係数のセットと前記出力係数のセットとの間の前記差を備える、請求項1 8に記載の装置。

【請求項 2 2】

少なくとも係数の前記再計算されたサブセットを使用して前記ビデオデータのブロックを変換するための手段をさらに備える、請求項1 8に記載の装置。

【手続補正 2】

【補正対象書類名】明細書

【補正対象項目名】0 1 0 7

【補正方法】変更

【補正の内容】

【0 1 0 7】

[00104] 様々な例について説明した。これらおよび他の例は、以下の特許請求の範囲に含まれる。

以下に、本願出願の当初の特許請求の範囲に記載された発明を付記する。

[ C 1 ]

ビデオ情報を記憶するように構成されたメモリと、

前記メモリと通信しているプロセッサであって、

変換を複数の変換段階に分解することと、

各変換段階における変換段階出力を決定するために前記複数の段階を使用して前記ビデオ情報を変換することと、

各変換段階における前記変換段階出力を所定のビット深度に制約することと、

前記複数の段階の最後の段階の前記制約された変換出力に対して演算を実行することと、ここにおいて、前記演算が、前記所定のビット深度を有するデータとともに使用するためにのみ利用可能である、

を行うように構成されたプロセッサと

を備える、ビデオエンコーダ。

[ C 2 ]

前記プロセッサが、メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分解するようにさらに構成された、C 1 に記載のビデオエンコーダ。

[ C 3 ]

前記プロセッサは、クリッピングされた変換段階出力が必ず所定の範囲内に入るよう前に前記変換段階出力をクリッピングすることによって前記変換段階出力を制約するようにさらに構成された、C 1 に記載のビデオエンコーダ。

[ C 4 ]

前記所定のビット深度が 1 6 ビットである、C 1 に記載のビデオエンコーダ。

[ C 5 ]

前記プロセッサが、前記制約された変換段階出力と前記変換段階出力との間の偏差を決定するようにさらに構成された、C 1 に記載のビデオエンコーダ。

[ C 6 ]

前記偏差が、前記制約された変換段階出力と前記変換段階出力との間の前記差を備える、C 5 に記載のビデオエンコーダ。

[ C 7 ]

前記プロセッサは、前記偏差が所定のしきい値よりも大きいとき、少なくとも 1 つの変換段階内で係数のサブセットを再計算するようにさらに構成された、C 5 に記載のビデオ

エンコーダ。

[ C 8 ]

前記プロセッサが、少なくとも係数の前記再計算されたサブセットを使用して前記ビデオ情報を変換するようにさらに構成された、C 7に記載のビデオエンコーダ。

[ C 9 ]

ビデオを符号化する方法であって、

ビデオ情報を記憶することと、

変換を複数の変換段階に分解することと、

各変換段階における変換段階出力を決定するために前記複数の段階を使用して前記ビデオ情報を変換することと、

各変換段階における前記変換段階出力を所定のビット深度に制約することと、

前記複数の段階の最後の段階の前記制約された変換出力に対して演算を実行することと、ここにおいて、前記演算が、前記所定のビット深度を有するデータとともに使用するためのみ利用可能である、

を備える方法。

[ C 10 ]

メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分解することをさらに備える、C 9に記載の方法。

[ C 11 ]

クリッピングされた変換段階出力が必ず所定の範囲内に入るよう前記変換段階出力をクリッピングすることによって前記変換段階出力を制約することをさらに備える、C 9に記載の方法。

[ C 12 ]

前記所定のビット深度が 16 ビットである、C 9に記載の方法。

[ C 13 ]

前記制約された変換段階出力と前記変換段階出力との間の偏差を決定することをさらに備える、C 9に記載の方法。

[ C 14 ]

前記偏差が、前記制約された変換段階出力と前記変換段階出力との間の前記差を備える、C 13に記載の方法。

[ C 15 ]

前記偏差が所定のしきい値よりも大きいとき、少なくとも 1 つの変換段階内で係数のサブセットを再計算することをさらに備える、C 13に記載の方法。

[ C 16 ]

少なくとも係数の前記再計算されたサブセットを使用して前記ビデオ情報を変換することをさらに備える、C 15に記載の方法。

[ C 17 ]

実行されたとき、装置に、

ビデオ情報を記憶することと、

変換を複数の変換段階に分解することと、

各変換段階における変換段階出力を決定するために前記複数の段階を使用して前記ビデオ情報を変換することと、

各変換段階における前記変換段階出力を所定のビット深度に制約することと、

前記複数の段階の最後の段階の前記制約された変換出力に対して演算を実行することと、ここにおいて、前記演算が、前記所定のビット深度を有するデータとともに使用するためのみ利用可能である、

を行わせるコードを備える、非一時的コンピュータ可読媒体。

[ C 18 ]

実行されたとき、前記装置に、メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分解することを行わせるコードをさらに備える、C 17に記

載のコンピュータ可読媒体。

[ C 1 9 ]

実行されたとき、前記装置に、クリッピングされた変換段階出力が必ず所定の範囲内に入るように前記変換段階出力をクリッピングすることによって前記変換段階出力を制約することを行わせるコードをさらに備える、C 1 7 に記載のコンピュータ可読媒体。

[ C 2 0 ]

実行されたとき、前記装置に、前記制約された変換段階出力と前記変換段階出力との間の偏差を決定することを行わせるコードをさらに備える、C 1 7 に記載のコンピュータ可読媒体。

[ C 2 1 ]

前記偏差が、前記制約された変換段階出力と前記変換段階出力との間の前記差を備える、C 2 0 に記載のコンピュータ可読媒体。

[ C 2 2 ]

実行されたとき、前記装置に、前記偏差が所定のしきい値よりも大きいとき、少なくとも1つの変換段階内で係数のサブセットを再計算することを行わせるコードをさらに備える、C 2 0 に記載のコンピュータ可読媒体。

[ C 2 3 ]

実行されたとき、前記装置に、少なくとも係数の前記再計算されたサブセットを使用して前記ビデオ情報を変換することを行わせるコードをさらに備える、C 2 2 に記載のコンピュータ可読媒体。

[ C 2 4 ]

ビデオを符号化するための装置であって、

ビデオ情報を記憶するための手段と、

変換を複数の変換段階に分解するための手段と、

各変換段階における変換段階出力を決定するために前記複数の段階を使用して前記ビデオ情報を変換するための手段と、

各変換段階における前記変換段階出力を所定のピット深度に制約するための手段と、

前記複数の段階の最後の段階の前記制約された変換出力に対して演算を実行するための手段と、ここにおいて、前記演算が、前記所定のピット深度を有するデータとともに使用するためにのみ利用可能である、

を備える、装置。

[ C 2 5 ]

メッシュベースの方法、バタフライ方法、または偶数奇数分解を使用して前記変換を分解するための手段をさらに備える、C 2 5 に記載の装置。

[ C 2 6 ]

クリッピングされた変換段階出力が必ず所定の範囲内に入るように前記変換段階出力をクリッピングすることによって前記変換段階出力を制約するための手段をさらに備える、C 2 5 に記載の装置。

[ C 2 7 ]

前記制約された変換段階出力と前記変換段階出力との間の偏差を決定するための手段をさらに備える、C 2 5 に記載の装置。

[ C 2 8 ]

前記偏差が、前記制約された変換段階出力と前記変換段階出力との間の前記差を備える、C 2 7 に記載の装置。

[ C 2 9 ]

前記偏差が所定のしきい値よりも大きいとき、少なくとも1つの変換段階内で係数のサブセットを再計算するための手段をさらに備える、C 2 7 に記載の装置。

[ C 3 0 ]

少なくとも係数の前記再計算されたサブセットを使用して前記ビデオ情報を変換するための手段をさらに備える、C 3 0 に記載の装置。

## 【国際調査報告】

INTERNATIONAL SEARCH REPORT		International application No PCT/US2015/020237
<b>A. CLASSIFICATION OF SUBJECT MATTER</b> INV. H04N19/42 H04N19/60 ADD.		
According to International Patent Classification (IPC) or to both national classification and IPC		
<b>B. FIELDS SEARCHED</b> Minimum documentation searched (classification system followed by classification symbols) H04N		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) EPO-Internal, WPI Data		
<b>C. DOCUMENTS CONSIDERED TO BE RELEVANT</b>		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2012/009237 A1 (QUALCOMM INC [US]; JOSHI RAJAN L [US]; CHIEN WEI-JUNG [US]; KARCZEWICZ) 19 January 2012 (2012-01-19) abstract paragraphs [0004], [0035], [0041], [0044] - [0055], [0063] figures 2, 3 ----- -/-/	1-30
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C.		<input checked="" type="checkbox"/> See patent family annex.
* Special categories of cited documents : "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed		
*T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family		
Date of the actual completion of the international search	Date of mailing of the international search report	
15 December 2015	21/12/2015	
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Authorized officer  Mayer, Claudia	

## INTERNATIONAL SEARCH REPORT

International application No PCT/US2015/020237
---

## C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	MEHER PRAMOD KUMAR ET AL: "Efficient Integer DCT Architectures for HEVC", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE SERVICE CENTER, PISCATAWAY, NJ, US, vol. 24, no. 1, 1 January 2014 (2014-01-01), pages 168-178, XP011536733, ISSN: 1051-8215, DOI: 10.1109/TCSVT.2013.2276862 [retrieved on 2014-01-03] abstract sections I-IV tables I, III figures 1-3, 7-8 -----	1-4, 9-12, 17-19, 24-26
X	WO 2008/002881 A2 (QUALCOMM INC [US]; GARUDADRI HARINATH [US]; REZNIK YURIY [US]) 3 January 2008 (2008-01-03) abstract paragraphs [0005] - [0021], [0114] - [0146] figures 9-12 -----	1-4, 9-12, 17-19, 24-26
A	Jie Liang ET AL: "Fast multiplierless approximations of the DCT with the lifting scheme", IEEE Transactions on Signal Processing, 1 January 2001 (2001-01-01), pages 3032-3044, XP055235999, New York DOI: 10.1109/78.969511 Retrieved from the Internet: URL: <a href="http://thanglong.ece.jhu.edu/Tran/Pub/binDCT.pdf">http://thanglong.ece.jhu.edu/Tran/Pub/binDCT.pdf</a> abstract sections I-V figures 1-5 table II -----	1-30
A	US 2013/243083 A1 (SEZER OSMAN GOKHAN [US]) 19 September 2013 (2013-09-19) abstract paragraphs [0050] - [0069], [0080], [0111] - [0114] figures 5, 21 -----	1-30
1		

**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No
PCT/US2015/020237

Patent document cited in search report		Publication date	Patent family member(s)	Publication date
WO 2012009237	A1	19-01-2012	CN 103004192 A EP 2594072 A1 JP 5727006 B2 JP 2013534795 A KR 20130054354 A US 2012014455 A1 WO 2012009237 A1	27-03-2013 22-05-2013 03-06-2015 05-09-2013 24-05-2013 19-01-2012 19-01-2012
WO 2008002881	A2	03-01-2008	BR P10712984 A2 CA 2653692 A1 CA 2653693 A1 CN 102007489 A CN 102016829 A EP 2035958 A2 EP 2035959 A2 JP 5086343 B2 JP 5269782 B2 JP 5661836 B2 JP 2010501911 A JP 2010505154 A JP 2013211011 A KR 20090016732 A KR 20090027670 A TW 200814738 A TW 200814792 A US 2007297504 A1 US 2007299897 A1 WO 2008002881 A2 WO 2008002897 A2	03-04-2012 03-01-2008 03-01-2008 06-04-2011 13-04-2011 18-03-2009 18-03-2009 28-11-2012 21-08-2013 28-01-2015 21-01-2010 18-02-2010 10-10-2013 17-02-2009 17-03-2009 16-03-2008 16-03-2008 27-12-2007 27-12-2007 03-01-2008 03-01-2008
US 2013243083	A1	19-09-2013	US 2013243083 A1 US 2015326871 A1	19-09-2013 12-11-2015

---

フロントページの続き

(81)指定国 AP(BW,GH,GM,KE,LR,LS,MW,MZ,NA,RW,SD,SL,ST,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,RU,TJ,TM),EP(AL,AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HR,HU,IE,IS,IT,LT,LU,LV,MC,MK,MT,NL,NO,PL,PT,R0,RS,SE,SI,SK,SM,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,KM,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AO,AT,AU,AZ,BA,BB,BG,BH,BN,BR,BW,BY,BZ,CA,CH,CL,CN,CO,CR,CU,CZ,DE,DK,DM,D0,DZ,EC,EE,EG,ES,FI,GB,GD,GE,GH,GM,GT,HN,HR,HU,ID,IL,IN,IR,IS,JP,KE,KG,KN,KP,KR,KZ,LA,LC,LK,LR,LS,LU,LY,MA,MD,ME,MG,MK,MN,MW,MX,MY,MZ,NA,NG,NI,NO,NZ,OM,PA,PE,PG,PH,PL,PT,QA,RO,RS,RU,RW,SA,SC,SD,SE,SG,SK,SL,SM,ST,SV,SY,TH,TJ,TM,TN,TR,TT,TZ,UA,UG,US

(72)発明者 チョン、イン・スク

アメリカ合衆国、カリフォルニア州 92121-1714、サン・ディエゴ、モアハウス・ドライブ 5775

(72)発明者 ユ、ヤン

アメリカ合衆国、カリフォルニア州 92121-1714、サン・ディエゴ、モアハウス・ドライブ 5775

(72)発明者 ワン、シャンリン

アメリカ合衆国、カリフォルニア州 92121-1714、サン・ディエゴ、モアハウス・ドライブ 5775

(72)発明者 カークゼウィックズ、マルタ

アメリカ合衆国、カリフォルニア州 92121-1714、サン・ディエゴ、モアハウス・ドライブ 5775

F ターム(参考) 5C159 KK61 MA21 PP04 TA35 TB08 TC08 TD12 UA02 UA05