

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号

特許第7106689号

(P7106689)

(45)発行日 令和4年7月26日(2022.7.26)

(24)登録日 令和4年7月15日(2022.7.15)

(51)国際特許分類

F I

H 0 4 N 19/12 (2014.01)

H 0 4 N 19/12

H 0 4 N 19/157 (2014.01)

H 0 4 N 19/157

H 0 4 N 19/176 (2014.01)

H 0 4 N 19/176

H 0 4 N 19/70 (2014.01)

H 0 4 N 19/70

請求項の数 6 (全54頁)

(21)出願番号 特願2020-571578(P2020-571578)

(86)(22)出願日 令和2年2月12日(2020.2.12)

(65)公表番号 特表2021-531680(P2021-531680
A)

(43)公表日 令和3年11月18日(2021.11.18)

(86)国際出願番号 PCT/CN2020/074780

(87)国際公開番号 WO2020/177509

(87)国際公開日 令和2年9月10日(2020.9.10)

審査請求日 令和3年1月14日(2021.1.14)

(31)優先権主張番号 62/813,136

(32)優先日 平成31年3月3日(2019.3.3)

(33)優先権主張国・地域又は機関

米国(US)

(31)優先権主張番号 62/815,293

(32)優先日 平成31年3月7日(2019.3.7)

最終頁に続く

(73)特許権者 504161984

ホアウェイ・テクノロジーズ・カンパニ
ー・リミテッド中華人民共和国・518129・グアン
ドン・シェンツェン・ロンガン・ディス
トリクト・バンティアン・(番地なし)
・ホアウェイ・アドミニストレーション
・ビルディング

(74)代理人 100110364

弁理士 実広 信哉

(74)代理人 100133569

弁理士 野村 進

(72)発明者 ハン・ガオ

ドイツ・80992・ミュンヘン・リー
スシュトラッセ・25

最終頁に続く

(54)【発明の名称】 変換プロセスのために使用されるエンコーダ、デコーダ、および対応する方法

(57)【特許請求の範囲】

【請求項1】

デコードデバイスによって実現される符号化の方法であって、
現在の符号化ブロックについて多重変換選択(MTS)方式が利用できず、現在の変換ブロッ
クの幅が第1のしきい値以上かつ第2のしきい値以下であり、
かつ前記現在の変換ブロックの高さが第3のしきい値以上かつ第4のしきい値以下であるとき、
前記現在の変換ブロックが前記現在の符号化ブロックに対応され、
前記現在の変換ブロックに対応するブロック幅および高さの比の値が第5のしきい値より
も大きいか否かを決定し、
ブロック幅および高さの前記比の前記値が前記第5のしきい値以下であるとき、
前記現在の変換ブロックの水平および垂直変換について変換コアDST 7が使用される、
ことを備える方法。

【請求項2】

前記方法は、

ブロック幅および高さの前記比の前記値が前記第5のしきい値よりも大きく、かつ前記現
在の変換ブロックの前記幅が前記現在の変換ブロックの前記高さよりも小さいとき、
前記現在の変換ブロックの水平変換について変換コアDST 7が使用され、前記現在の変換
ブロックの垂直変換について変換コアDCT 2が使用されることをさらに備える、請求項1
に記載の方法。

【請求項3】

前記方法は、

ブロック幅および高さの前記比の前記値が前記第5のしきい値よりも大きく、かつ前記現在の変換ブロックの前記幅が前記現在の変換ブロックの前記高さよりも大きいとき、前記現在の変換ブロックの水平変換について変換コアDCT 2が使用され、前記現在の変換ブロックの垂直変換について変換コアDST 7が使用されることをさらに備える、請求項1または2に記載の方法。

【請求項4】

請求項1から3のいずれか一項に記載の方法を実行するための処理回路を備えるデコーダ(30)。

【請求項5】

請求項1から3のいずれか一項に記載の方法を実行するためのプログラムコードを備えるコンピュータプログラム。

【請求項6】

デコーダであって、

1つまたは複数のプロセッサと、

前記プロセッサに結合され、前記プロセッサによる実行のためのプログラミングを記憶する非一時的コンピュータ可読記憶媒体とを備え、前記プログラミングが、前記プロセッサによって実行されたとき、請求項1から3のいずれか一項に記載の方法を実行するように前記デコーダを構成する、デコーダ。

【発明の詳細な説明】

【技術分野】

【0001】

本出願(開示)の実施形態は、一般に、ピクチャまたはビデオ処理の分野に、より詳細には、変換プロセスに関する。

【背景技術】

【0002】

ビデオ符号化(ビデオエンコードおよびビデオデコード)は、広い範囲のデジタルビデオアプリケーション、例えば、放送デジタルTV、インターネットおよびモバイルネットワーク上でのビデオ伝送、ビデオチャットのようなリアルタイム会話型アプリケーション、ビデオ会議、DVDおよびBlu-ray(登録商標)ディスク、ビデオコンテンツ収集および編集システム、およびセキュリティアプリケーションのカムコーダにおいて使用される。

【0003】

比較的短いビデオでさえ描写するために必要とされるビデオデータの量は、かなりである可能性があり、これは、限定された帯域幅容量を有する通信ネットワークを渡ってデータがストリーミングされ、またはそうでなく伝達されることになるときに、困難をもたらし得る。従って、ビデオデータは、一般に、現代の電気通信ネットワークを渡って伝達される前に圧縮される。メモリリソースが限定され得るので、ビデオが記憶デバイスにおいて記憶されるとき、ビデオのサイズも問題である可能性がある。ビデオ圧縮デバイスは、しばしば、伝送または記憶の前に、ソースにおいてソフトウェアおよび／またはハードウェアを使用してビデオデータを符号化し、それにより、デジタルビデオ画像を表現するために必要とされるデータの量を減らす。圧縮されたデータは、次いで、ビデオデータをデコードするビデオ復元デバイスによって宛先において受信される。限定されたネットワークリソース、およびより高いビデオ品質の増加さえしている需要を有して、ピクチャ品質においてほとんどないし全く犠牲なしで圧縮率を改善する、改善された圧縮および復元技法が望ましい。

【発明の概要】

【課題を解決するための手段】

【0004】

本出願の実施形態は、独立請求項による、エンコードおよびデコードのための装置および方法を提供する。

10

20

30

40

50

【0005】

上記および他の目的は、独立請求項の主題によって達成される。さらなる実装形式は、従属請求項、説明、および図から明らかである。

【0006】

発明の第1の態様は、デコードデバイスによって実現される符号化の方法であって、現在の符号化ブロックの予測モードがイントラ予測であり、かつ現在の符号化ブロックが、イントラ下位区分、ISP、符号化モードを使用することによって処理されるとき、現在の変換ブロックの幅が第1のしきい値以上であるか否か、および現在の変換ブロックの幅が第2のしきい値以下であるか否かを決定するステップであって、現在の変換ブロックが現在の符号化ブロックに対応される、ステップと、現在の変換ブロックの幅が第1のしきい値以上であり、かつ現在の変換ブロックの幅が第2のしきい値以下であるとき、現在の変換ブロックにおいて変換プロセスを実行するステップであって、水平方向変換において変換コアDST7が使用される、ステップとを備える方法を開示する。

10

【0007】

1つの可能な実装では、方法は、

現在の変換ブロックの幅が第1のしきい値よりも小さいとき、または現在の変換ブロックの幅が第2のしきい値よりも大きいとき、

現在の変換ブロックにおいて変換プロセスを実行するステップであって、水平変換において変換プロセス変換コアDCT2が使用される、ステップをさらに備える。

【0008】

20

1つの可能な実装では、方法は、

現在の符号化ブロックについてのシンタックスの値を取得するステップであって、現在の符号化ブロックについてのシンタックスの値は、現在の符号化ブロックの予測モードがイントラ予測であることを示すために使用される、ステップをさらに備える。

【0009】

1つの可能な実装では、現在の符号化ブロックについてのシンタックスの値は、現在の符号化ブロックの指定予測モードを示さない。

【0010】

発明の第2の態様は、デコードデバイスによって実現される符号化の方法であって、現在の符号化ブロックの予測モードがイントラ予測であり、かつ現在の符号化ブロックが、イントラ下位区分、ISP、符号化モードを使用することによって処理されるとき、現在の変換ブロックの高さが第3のしきい値以上であるか否か、および現在の変換ブロックの高さが第4のしきい値以下であるか否かを決定するステップであって、現在の変換ブロックが現在の符号化ブロックに対応される、ステップと、現在の変換ブロックの高さが第3のしきい値以上かつ第4のしきい値以下であるとき、現在の変換ブロックにおいて変換プロセスを実行するステップであって、垂直方向変換において変換コアDST 7が使用される、ステップとを備える方法を開示する。

30

【0011】

1つの可能な実装では、方法は、現在の変換ブロックの高さが第3のしきい値よりも小さいとき、または現在の変換ブロックの高さが第4のしきい値よりも大きいとき、

40

現在の変換ブロックにおいて変換プロセスを実行するステップであって、垂直方向変換において変換コアDCT 2が使用される、ステップをさらに備える。

【0012】

1つの可能な実装では、方法は、

現在の符号化ブロックについてのシンタックスの値を取得するステップであって、現在の符号化ブロックについてのシンタックスの値は、現在の符号化ブロックの予測モードがイントラ予測であることを示すために使用される、ステップをさらに備える。

【0013】

1つの可能な実装では、現在の符号化ブロックについてのシンタックスの値は、現在の符号化ブロックの指定予測モードを示さない。

50

【0014】

発明の第3の態様は、デコードデバイスによって実現される符号化の方法であって、現在の符号化ブロックについて多重変換選択、MTS、方式が利用できないとき、現在の変換ブロックの幅が第1のしきい値以上であるか否か、および現在の変換ブロックの幅が第2のしきい値以下であるか否かを決定するステップであって、現在の変換ブロックが現在の符号化ブロックに対応される、ステップと、現在の変換ブロックの幅が第1のしきい値以上かつ第2のしきい値以下であるとき、現在の変換ブロックにおいて変換プロセスを実行するステップであって、水平方向変換において変換コアDST 7が使用される、ステップとを備える方法を開示する。

【0015】

1つの可能な実装では、方法は、現在の変換ブロックの幅が第1のしきい値よりも小さいとき、または現在の変換ブロックの幅が第2のしきい値よりも大きいとき、現在の変換ブロックにおいて変換プロセスを実行するステップであって、水平方向変換において変換コアDCT 2が使用される、ステップをさらに備える。

【0016】

発明の第4の態様は、デコードデバイスによって実現される符号化の方法であって、現在の符号化ブロックについて多重変換選択、MTS、方式が利用できないとき、現在の変換ブロックの高さが第3のしきい値以上であるか否か、および現在の変換ブロックの高さが第4のしきい値以下であるか否かを決定するステップであって、現在の変換ブロックが現在の符号化ブロックに対応される、ステップと、現在の変換ブロックの高さが第3のしきい値以上かつ第4のしきい値以下であるとき、現在の変換ブロックにおいて変換プロセスを実行するステップであって、垂直方向変換において変換コアDST 7が使用される、ステップとを備える方法を開示する。

【0017】

1つの可能な実装では、方法は、現在の変換ブロックの高さが第3のしきい値よりも小さいとき、または現在の変換ブロックの高さが第4のしきい値よりも大きいとき、現在の変換ブロックにおいて変換プロセスを実行するステップであって、垂直方向変換において変換コアDCT 2が使用される、ステップをさらに備える。

【0018】

発明の第5の態様は、デコードデバイスによって実現される符号化の方法であって、現在の符号化ブロックについて多重変換選択、MTS、方式が利用できず、現在の変換ブロックの幅が第1のしきい値以上かつ第2のしきい値以下であり、かつ現在の変換ブロックの高さが第3のしきい値以上かつ第4のしきい値以下であるとき、現在の変換ブロックが現在の符号化ブロックに対応され、現在の変換ブロックに対応するブロック幅および高さの比の値が第5のしきい値よりも大きいか否かを決定し、ブロック幅および高さの比の値が第5のしきい値以下であるとき、現在の変換ブロックの水平および垂直変換について変換コアDST 7が使用されることをさらに備える方法を開示する。

【0019】

1つの可能な実装では、方法は、ブロック幅および高さの比の値が第5のしきい値よりも大きく、かつ現在の変換ブロックの幅が現在の変換ブロックの高さよりも小さいとき、現在の変換ブロックの水平変換について変換コアDST 7が使用され、現在の変換ブロックの垂直変換について変換コアDCT 2が使用されることをさらに備える。

【0020】

1つの可能な実装では、方法は、ブロック幅および高さの比の値が第5のしきい値よりも大きく、かつ現在の変換ブロックの幅が現在の変換ブロックの高さよりも大きいとき、

現在の変換ブロックの水平変換について変換コアDCT 2が使用され、現在の変換ブロックの垂直変換について変換コアDST 7が使用されることをさらに備える。

【0021】

発明の第6の態様は、上記の方法の実施形態のうちのいずれか1つを実行するための処理回路を備えるデコーダ(30)を開示する。

【0022】

発明の第7の態様は、上記の方法の実施形態のうちのいずれか1つによる方法を実行するためのプログラムコードを備えるコンピュータプログラム製品を開示する。

【0023】

発明の第8の態様は、

1つまたは複数のプロセッサと、プロセッサに結合され、プロセッサによる実行のためのプログラミングを記憶する非一時的コンピュータ可読記憶媒体とを備えるデコーダであって、プログラミングは、プロセッサによって実行されたとき、上記の方法の実施形態のうちのいずれか1つによる方法を実行するようにデコーダを構成する、デコーダを開示する。

【0024】

1つまたは複数の実施形態の詳細が、添付の図面および以下の説明において記載される。他の特徴、目的、および利点は、説明、図面、および請求項から明らかになるであろう。

【0025】

以下では、発明の実施形態が、添付の図および図面を参照して、より詳細に説明される。

【図面の簡単な説明】

【0026】

【図1A】発明の実施形態を実現するように構成されたビデオ符号化システムの一例を表すブロック図である。

【図1B】発明の実施形態を実現するように構成されたビデオ符号化システムの別の例を表すブロック図である。

【図2】発明の実施形態を実現するように構成されたビデオエンコーダの一例を表すブロック図である。

【図3】発明の実施形態を実現するように構成されたビデオデコーダの例示の構造を表すブロック図である。

【図4】エンコード装置またはデコード装置の一例を例示するブロック図である。

【図5】エンコード装置またはデコード装置の別の例を例示するブロック図である。

【図6】 4×8 および 8×4 ブロックの分割の一例である。

【図7】 4×8 、 8×4 、および 4×4 ブロックを除く、ブロックの分割の一例である。

【図8】コンテンツ配信サービスを実現するコンテンツ供給システム3100の例示の構造を表すブロック図である。

【図9】端末デバイスの一例の構造を表すブロック図である。

【発明を実施するための形態】

【0027】

以下では、同一の参照符号は、明示的にそうでなく指定されないならば、同一のまたは少なくとも機能的に等価な特徴を指す。

【0028】

以下の説明では、開示の部分を形成し、発明の実施形態の特定の態様または本発明の実施形態が使用され得る特定の態様を例示として表す、添付の図への参照が行われる。発明の実施形態が、他の態様において使用され、図の中に描写されていない構造的または論理的な変更を備え得ることが理解される。従って、以下の詳細な説明は限定する意味で受け取られるべきでなく、本発明の範囲は添付の請求項によって定義される。

【0029】

例えば、説明される方法に関する開示が、その方法を実行するように構成された対応するデバイスまたはシステムについても当てはまり得るとともに逆も同様であることが理解される。例えば、1つまたは複数の特定の方法のステップが説明されるならば、そのような1

10

20

30

40

50

つまたは複数のユニットが明示的に説明されない、または図の中に例示されないとしても、対応するデバイスは、説明される1つまたは複数の方法のステップを実行するための1つまたは複数のユニット、例えば、機能ユニット(例えば、1つまたは複数のステップを実行する1つのユニット、または複数のステップのうちの1つまたは複数の各々が実行する複数のユニット)を含んでよい。一方、例えば、1つまたは複数のユニット、例えば、機能ユニットに基づいて、特定の装置が説明されるならば、そのような1つまたは複数のステップが明示的に説明されない、または図の中に例示されないとしても、対応する方法は、1つまたは複数のユニットの機能を実行するための1つのステップ(例えば、1つまたは複数のユニットの機能を実行する1つのステップ、または複数のユニットのうちの1つまたは複数の機能を各々が実行する複数のステップ)を含んでよい。さらに、特にそうでなく注記されないならば、ここで説明される様々な例示的な実施形態および／または態様の特徴が互いに組み合わせられてよいことが理解される。

10

【0030】

ビデオ符号化は、典型的に、ビデオまたはビデオシーケンスを形成する、ピクチャのシーケンスの処理を指す。用語「ピクチャ」の代わりに、ビデオ符号化の分野では用語「フレーム」または「画像」が同義語として使用され得る。ビデオ符号化(または、一般に符号化)は、2つの部分、ビデオエンコードおよびビデオデコードを備える。ビデオエンコードは、ソース側において実行され、典型的に、ビデオピクチャを表現するために要求されるデータの量を(より効率的な記憶および／または伝送のために)減少させるために、(例えば、圧縮によって)元のビデオピクチャを処理することを備える。ビデオデコードは、宛先側において実行され、典型的に、ビデオピクチャを再構成するための、エンコーダと比較して逆の処理を備える。ビデオピクチャ(または、一般にピクチャ)の「符号化」に言及する実施形態は、ビデオピクチャまたはそれぞれのビデオシーケンスの「エンコード」または「デコード」に関すると理解されるものである。エンコード部分とデコード部分の組み合わせは、CODEC(Coding and Decoding(符号化およびデコード))とも呼ばれる。

20

【0031】

損失のないビデオ符号化の場合には、元のビデオピクチャは再構成されることが可能であり、すなわち、(記憶または伝送の間に伝送損失または他のデータ損失がないと仮定すると)再構成されたビデオピクチャは元のビデオピクチャと同じ品質を有する。損失のあるビデオ符号化の場合には、ビデオピクチャを表現するデータの量を減少させるために、例えば、量子化によって、さらなる圧縮が実行され、ビデオピクチャは、デコーダにおいて完全に再構成されることが可能でなく、すなわち、再構成されたビデオピクチャの品質は元のビデオピクチャの品質と比較して、より低い、またはより悪い。

30

【0032】

いくつかのビデオ符号化規格は、「損失のあるハイブリッドビデオコーデック」のグループに属する(すなわち、サンプル領域における空間および時間予測と、変換領域において量子化を適用するための2D変換符号化とを組み合わせる)。ビデオシーケンスの各ピクチャは、典型的に、重複しないブロックのセットに区分され、符号化は、典型的に、ブロックレベルにおいて実行される。言い換えれば、エンコーダにおいて、ビデオは、典型的に、例えば、空間(イントラピクチャ)予測および／または時間(インターピクチャ)予測を使用して予測ブロックを生成し、現在のブロック(現在処理されている／処理されるべきブロック)から予測ブロックを減算して残差ブロックを取得し、残差ブロックを変換し、変換領域において残差ブロックを量子化して伝送されるべきデータの量を減少させること(圧縮)によって、ブロック(ビデオブロック)レベルにおいて処理され、すなわちエンコードされ、一方、デコーダにおいて、エンコーダと比較して逆の処理が、エンコードされ、または圧縮されたブロックに適用されて表現のために現在のブロックを再構成する。さらに、エンコーダは、後続のブロックを処理する、すなわち符号化するために、両方が同一の予測(例えば、イントラおよびインター予測)および／または再構成を生成するように、デコーダ処理ループを二重化する。

40

【0033】

50

以下では、ビデオ符号化システム10、ビデオエンコーダ20、およびビデオデコーダ30の実施形態が、図1～図3に基づいて説明される。

【0034】

図1Aは、この本出願の技法を利用し得る例示の符号化システム10、例えば、ビデオ符号化システム10(または短縮して符号化システム10)を例示する概略ブロック図である。ビデオ符号化システム10のビデオエンコーダ20(または短縮してエンコーダ20)およびビデオデコーダ30(または短縮してデコーダ30)は、本出願において説明される様々な例による技法を実行するように構成され得るデバイスの例を表現する。

【0035】

図1Aに表されたように、符号化システム10は、例えば、エンコードされたピクチャデータ13をデコードするための宛先デバイス14に、エンコードされたピクチャデータ21を提供するように構成されたソースデバイス12を備える。

【0036】

ソースデバイス12は、エンコーダ20を備え、加えて、すなわち任意選択で、ピクチャソース16、プリプロセッサ(または、前処理ユニット)18、例えば、ピクチャプリプロセッサ18、および通信インターフェースまたは通信ユニット22を備えてよい。

【0037】

ピクチャソース16は、任意の種類のピクチャキャプチャデバイス、例えば、実世界ピクチャをキャプチャするためのカメラ、および／または任意の種類のピクチャ生成デバイス、例えば、コンピュータアニメーション化されたピクチャを生成するためのコンピュータグラフィックスプロセッサ、または実世界ピクチャ、コンピュータ生成されたピクチャ(例えば、スクリーンコンテンツ、仮想現実(virtual reality(VR))ピクチャ)、および／またはそれらの任意の組み合わせ(例えば、拡張現実(augmented reality(AR))ピクチャ)を取得および／または提供するための任意の種類の他のデバイスを備え、またはそれらであってよい。ピクチャソースは、上述のピクチャのうちのいずれかを記憶する任意の種類のメモリまたは記憶装置であってよい。

【0038】

プリプロセッサ18、および前処理ユニット18によって実行される処理と区別して、ピクチャまたはピクチャデータ17は、未処理ピクチャまたは未処理ピクチャデータ17とも呼ばれ得る。

【0039】

プリプロセッサ18は、(未処理)ピクチャデータ17を受信し、ピクチャデータ17において前処理を実行して、前処理されたピクチャ19または前処理されたピクチャデータ19を取得するように構成される。プリプロセッサ18によって実行される前処理は、例えば、トリミング、(例えば、RGBからYCbCrへの)カラーフォーマット変換、色補正、またはノイズ除去を備えてよい。前処理ユニット18が任意選択の構成要素であり得ることが理解されることが可能である。

【0040】

ビデオエンコーダ20は、前処理されたピクチャデータ19を受信し、エンコードされたピクチャデータ21を提供するように構成される(さらなる詳細が、例えば、図2に基づいて、以下で説明されるであろう)。

【0041】

ソースデバイス12の通信インターフェース22は、エンコードされたピクチャデータ21を受信し、記憶または直接の再構成のために、通信チャンネル13上で別のデバイス、例えば、宛先デバイス14または任意の他のデバイスへ、エンコードされたピクチャデータ21(または、それらのさらに処理された任意のバージョン)を伝送するように構成され得る。

【0042】

宛先デバイス14は、デコーダ30(例えば、ビデオデコーダ30)を備え、加えて、すなわち任意選択で、通信インターフェースまたは通信ユニット28、ポストプロセッサ32(または、後処理ユニット32)、およびディスプレイデバイス34を備えてよい。

10

20

30

40

50

【0043】

宛先デバイス14の通信インターフェース28は、例えば、ソースデバイス12から直接に、または任意の他のソース、例えば、記憶デバイス、例えば、エンコードされたピクチャデータ記憶デバイスから、エンコードされたピクチャデータ21(または、それらのさらに処理された任意のバージョン)を受信し、エンコードされたピクチャデータ21をデコーダ30に提供するように構成される。

【0044】

通信インターフェース22および通信インターフェース28は、ソースデバイス12と宛先デバイス14の間の直接の通信リンク、例えば、直接の有線または無線接続を介して、または任意の種類のネットワーク、例えば、有線または無線ネットワークまたはそれらの任意の組み合わせ、または任意の種類のプライベートおよびパブリックネットワーク、またはそれらの任意の種類の組み合わせを介して、エンコードされたピクチャデータ21またはエンコードされたデータ13を送信または受信するように構成され得る。

10

【0045】

通信インターフェース22は、例えば、適切なフォーマット、例えば、パケットの中に、エンコードされたピクチャデータ21をパッケージ化し、かつ／または通信リンクまたは通信ネットワーク上での伝送のために任意の種類の伝送エンコードまたは処理を使用してエンコードされたピクチャデータを処理するように構成され得る。

【0046】

通信インターフェース22の相手側を形成する通信インターフェース28は、例えば、伝送されたデータを受信し、任意の種類の対応する伝送デコードまたは処理および／またはパッケージ除去を使用して伝送データを処理してエンコードされたピクチャデータ21を取得するように構成され得る。

20

【0047】

通信インターフェース22と通信インターフェース28の両方が、ソースデバイス12から宛先デバイス14を指し示す、図1Aの中の通信チャネル13についての矢印によって示されるような単方向の通信インターフェース、または双方向の通信インターフェースとして構成されてよく、例えば、通信リンクおよび／またはデータ伝送、例えば、エンコードされたピクチャデータ伝送に関する、任意の他の情報を肯定応答および交換するために、例えば、接続をセットアップするために、メッセージを送信し、受信するように構成されてよい。

30

【0048】

デコーダ30は、エンコードされたピクチャデータ21を受信し、デコードされたピクチャデータ31またはデコードされたピクチャ31を提供するように構成される(さらなる詳細は、例えば、図3または図5に基づいて、以下で説明されるであろう)。

【0049】

宛先デバイス14のポストプロセッサ32は、デコードされたピクチャデータ31(再構成されたピクチャデータとも呼ばれる)、例えば、デコードされたピクチャ31を後処理して、後処理されたピクチャデータ33、例えば、後処理されたピクチャ33を取得するように構成される。後処理ユニット32によって実行される後処理は、例えば、ディスプレイデバイス34による、例えば、表示のためにデコードされたピクチャデータ31を準備するための、例えば、(例えば、YCbCrからRGBへの)カラーフォーマット変換、色補正、トリミング、または再サンプリング、または任意の他の処理を備えてよい。

40

【0050】

宛先デバイス14のディスプレイデバイス34は、例えば、ユーザまたは閲覧者に、ピクチャを表示するための後処理されたピクチャデータ33を受信するように構成される。ディスプレイデバイス34は、再構成されたピクチャを表現するための任意の種類のディスプレイ、例えば、統合型または外部のディスプレイまたはモニタであってよく、またはそれを備えてよい。ディスプレイは、例えば、液晶ディスプレイ(liquid crystal display(LCD))、有機発光ダイオード(organic light emitting diode(OLED))ディスプレイ、プラズマディスプレイ、プロジェクタ、マイクロLEDディスプレイ、液晶オンシリコン(liquid cry

50

stal on silicon(LCoS))、デジタル光プロセッサ(digital light processor(DLP))、または任意の種類の他のディスプレイを備えてよい。

【0051】

図1Aはソースデバイス12および宛先デバイス14を別個のデバイスとして描写するが、デバイスの実施形態はまた、ソースデバイス12または対応する機能と、宛先デバイス14または対応する機能の、両方または両方の機能を備えてよい。そのような実施形態では、ソースデバイス12または対応する機能、および宛先デバイス14または対応する機能は、同じハードウェアおよび／またはソフトウェアを使用して、または別個のハードウェアおよび／またはソフトウェアまたはそれらの任意の組み合わせによって、実現され得る。

【0052】

説明に基づいて当業者に対して明らかになることになるように、異なるユニットの機能または図1Aに表されたようなソースデバイス12および／または宛先デバイス14内の機能の存在および(正確な)分割は、実際のデバイスおよび適用に依存して変わり得る。

【0053】

エンコーダ20(例えば、ビデオエンコーダ20)またはデコーダ30(例えば、ビデオデコーダ30)、またはエンコーダ20とデコーダ30の両方は、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(digital signal processor(DSP))、特定用途向け集積回路(application-specific integrated circuit(ASIC))、フィールドプログラマブルゲートアレイ(field-programmable gate array(FPGA))、個別論理、ハードウェア、専用のビデオ符号化、またはそれらの任意の組み合わせのような、図1Bに表されたような処理回路を介して実現され得る。エンコーダ20は、図2のエンコーダ20に関して論じられるような様々なモジュールおよび／またはここで説明される任意の他のエンコーダシステムまたはサブシステムを具現するために、処理回路46を介して実現され得る。デコーダ30は、図3のデコーダ30に関して論じられるような様々なモジュールおよび／またはここで説明される任意の他のデコーダシステムまたはサブシステムを具現するために、処理回路46を介して実現され得る。処理回路は、後で論じられるような様々な演算を実行するように構成され得る。図5に表されたように、技法が部分的にソフトウェアで実現されるならば、デバイスは、ソフトウェアのための命令を、適した非一時的コンピュータ可読記憶媒体に記憶してよく、この開示の技法を実行するために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てよい。ビデオエンコーダ20およびビデオデコーダ30のいずれかは、例えば、図1Bに表されたように、組み合わせられたエンコーダ／デコーダ(CODEC)の部分として単一のデバイス内に統合され得る。

【0054】

ソースデバイス12および宛先デバイス14は、任意の種類のハンドヘルドまたは固定のデバイス、例えば、ノートブックまたはラップトップコンピュータ、スマートフォン、スマートフォン、タブレットまたはタブレットコンピュータ、カメラ、デスクトップコンピュータ、セットトップボックス、テレビ、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、(コンテンツサービスサーバまたはコンテンツ配信サーバのような)ビデオストリーミングデバイス、放送受信機デバイス、放送送信機デバイス、または同様のものを含む、広い範囲のデバイスのうちのいずれかを備えてよく、オペレーティングシステムを使用しなくてよく、または任意の種類のオペレーティングシステムを使用してもよい。いくつかの場合、ソースデバイス12および宛先デバイス14は無線通信のために装備されてよい。従って、ソースデバイス12および宛先デバイス14は無線通信デバイスであってよい。

【0055】

いくつかの場合、図1Aに例示されたビデオ符号化システム10は単に一例であり、本出願の技法は、エンコードおよびデコードデバイスの間の任意のデータ通信を必ずしも含まず、ビデオ符号化設定(例えば、ビデオエンコードまたはビデオデコード)に適用されてよい。他の例では、データは、ローカルメモリから取り出され、ネットワーク上でストリーミングされ、または同様である。ビデオエンコードデバイスは、データをエンコードしてメ

10

20

30

40

50

メモリに記憶してよく、かつ／またはビデオデコードデバイスは、データをメモリから取り出してデコードしてよい。いくつかの例では、互いに通信しないが、単にデータをメモリにエンコードし、かつ／またはデータをメモリから取り出してデコードするデバイスによって、エンコードおよびデコードが実行される。

【0056】

説明の便宜のため、例えば、高効率ビデオ符号化(High-Efficiency Video Coding(HEVC))への、またはITU-Tビデオ符号化エキスパートグループ(Video Coding Experts Group(VCEG))とISO/IECモーション・ピクチャ・エキスパート・グループ(Motion Picture Experts Group(MPEG))のビデオ符号化共同研究部会(Joint Collaboration Team on Video Coding(JCT-VC))によって策定された次世代ビデオ符号化規格である、多用途ビデオ符号化(Versatile Video Coding(VVC))の参照ソフトウェアへの参照によって、発明の実施形態がここで説明される。発明の実施形態がHEVCまたはVVCに限定されないことを、この技術分野の当業者は理解するであろう。

【0057】

エンコーダおよびエンコード方法

図2は、本出願の技法を実現するように構成される例示のビデオエンコーダ20の概略ブロック図を表す。図2の例では、ビデオエンコーダ20は、入力201(または入力インターフェース201)、残差計算ユニット204、変換処理ユニット206、量子化ユニット208、逆量子化ユニット210、および逆変換処理ユニット212、再構成ユニット214、ループフィルタユニット220、デコードされたピクチャバッファ(decoded picture buffer(DPB))230、モード選択ユニット260、エントロピーエンコードユニット270、および出力272(または出力インターフェース272)を備える。モード選択ユニット260は、インター予測ユニット244、イントラ予測ユニット254、および区分ユニット262を含んでよい。インター予測ユニット244は、動き推定ユニットおよび動き補償ユニット(表されていない)を含んでよい。図2に表されたようなビデオエンコーダ20は、ハイブリッドビデオエンコーダ、またはハイブリッドビデオコーデックによるビデオエンコーダとも呼ばれ得る。

【0058】

残差計算ユニット204、変換処理ユニット206、量子化ユニット208、モード選択ユニット260は、エンコーダ20の順方向信号経路を形成するとして言及されてよく、一方、逆量子化ユニット210、逆変換処理ユニット212、再構成ユニット214、バッファ216、ループフィルタ220、デコードされたピクチャバッファ(decoded picture buffer(DPB))230、インター予測ユニット244、およびイントラ予測ユニット254は、ビデオエンコーダ20の逆方向信号経路を形成するとして言及されてよく、ビデオエンコーダ20の逆方向信号経路はデコーダの信号経路に対応する(図3の中のビデオデコーダ30を見られたい)。逆量子化ユニット210、逆変換処理ユニット212、再構成ユニット214、ループフィルタ220、デコードされたピクチャバッファ(decoded picture buffer(DPB))230、インター予測ユニット244、およびイントラ予測ユニット254はまた、ビデオエンコーダ20の「ビルトインデコーダ」を形成することが言及される。

【0059】

ピクチャおよびピクチャ区分(ピクチャおよびブロック)

エンコーダ20は、ピクチャ17(またはピクチャデータ17)、例えば、ビデオまたはビデオシーケンスを形成するピクチャのシーケンスのピクチャを、例えば、入力201を介して受信するように構成され得る。受信されるピクチャまたはピクチャデータはまた、前処理されたピクチャ19(または前処理されたピクチャデータ19)であってよい。簡潔さの目的のために、以下の説明はピクチャ17に言及する。ピクチャ17は、現在のピクチャ、または(同じビデオシーケンス、すなわち、やはり現在のピクチャを備えるビデオシーケンスの他のピクチャ、例えば、前にエンコードされ、かつ／またはデコードされたピクチャから現在のピクチャを区別するために、特にビデオ符号化において)符号化されるべきピクチャとも呼ばれ得る。

【0060】

(デジタル)ピクチャは、強度値を有するサンプルの2次元のアレイまたは行列であるか、またはそのように見なされることが可能である。アレイ内のサンプルは、ピクセル(ピクチャ要素の短い形式)またはペルとも呼ばれ得る。アレイまたはピクチャの水平および垂直方向(または軸)におけるサンプルの数は、ピクチャのサイズおよび／または解像度を定義する。色の表現のために、典型的に、3つの色成分が採用され、すなわち、ピクチャは、3つのサンプルアレイが表現され、またはそれらを含んでよい。RGBフォーマットまたは色空間で、ピクチャは、対応する赤、緑、および青のサンプルアレイを備える。しかし、ビデオ符号化では、各ピクセルは、典型的に、ルミナンスおよびクロミナンスのフォーマットまたは色空間、例えば、YCbCrで表現され、YCbCrは、Y(時々、代わりにLも使用される)によって示されるルミナンス成分、およびCbおよびCrによって示される2つのクロミナンス成分を備える。ルミナンス(または短縮してルーマ)成分Yは、輝度または(例えば、グレースケールピクチャにおけるような)グレーレベル強度を表現し、一方、2つのクロミナンス(または短縮してクロマ)成分CbおよびCrは、色度または色情報成分を表現する。従って、YCbCrフォーマットでのピクチャは、ルミナンスサンプル値(Y)のルミナンスサンプルアレイ、およびクロミナンス値(CbおよびCr)の2つのクロミナンスサンプルアレイを備える。RGBフォーマットでのピクチャは、YCbCrフォーマットにコンバートされ、または変換されてよく、逆も同様であり、プロセスは色変換またはコンバートとしても知られる。ピクチャがモノクロであるならば、ピクチャはルミナンスサンプルアレイのみを備えてよい。従って、ピクチャは、例えば、モノクロフォーマットでのルーマサンプルのアレイ、または4:2:0、4:2:2、および4:4:4カラーフォーマットでの、ルーマサンプルのアレイおよびクロマサンプルの2つの対応するアレイであってよい。

【0061】

ビデオエンコーダ20の実施形態は、ピクチャ17を複数の(典型的に重複しない)ピクチャブロック203に区分するように構成されたピクチャ区分ユニット(図2に描写されない)を備えてよい。これらのブロックは、ルートブロック、マクロブロック(H.264/AVC)、または符号化ツリーブロック(coding tree block(CTB))または符号化ツリーユニット(coding tree unit(CTU))(H.265/HEVCおよびVVC)とも呼ばれ得る。ピクチャ区分ユニットは、ビデオシーケンスの全てのピクチャについて同じブロックサイズ、およびブロックサイズを定義する対応するグリッドを使用し、またはピクチャまたはピクチャのサブセットまたはグループの間でブロックサイズを変更し、各ピクチャを対応するブロックに区分するように構成され得る。

【0062】

さらなる実施形態では、ビデオエンコーダは、ピクチャ17のブロック203、例えば、ピクチャ17を形成する1つの、いくつかの、または全てのブロックを、直接に受信するように構成され得る。ピクチャブロック203は、現在のピクチャブロック、または符号化されるべきピクチャブロックとも呼ばれ得る。

【0063】

ピクチャ17のように、ピクチャブロック203は再び、強度値(サンプル値)を有するサンプルの、しかしピクチャ17よりも小さい寸法の、2次元のアレイまたは行列であるか、またはそのように見なされることが可能である。言い換えれば、ブロック203は、例えば、1つのサンプルアレイ(例えば、モノクロピクチャ17の場合にはルーマアレイ、またはカラーピクチャの場合にはルーマまたはクロマアレイ)、または3つのサンプルアレイ(例えば、カラーピクチャ17の場合にはルーマおよび2つのクロマアレイ)、または適用されるカラーフォーマットに依存して任意の他の数および／または種類のアレイを備えてよい。ブロック203の水平および垂直方向(または軸)におけるサンプルの数は、ブロック203のサイズを定義する。従って、ブロックは、例えば、サンプルのM×N(M列×N行)アレイ、または変換係数のM×Nアレイであってよい。

【0064】

図2に表されたようなビデオエンコーダ20の実施形態は、ピクチャ17をブロックごとにエンコードするように構成されてよく、例えば、エンコードおよび予測がブロック203ごと

10

20

30

40

50

に実行される。

【0065】

図2に表されたようなビデオエンコーダ20の実施形態は、スライス(ビデオスライスとも呼ばれる)を使用することによってピクチャを区分および／またはエンコードするようにさらに構成されてよく、ピクチャは、(典型的に重複しない)1つまたは複数のスライスに区分され、またはそのスライスを使用してエンコードされてよく、各スライスは、1つまたは複数のブロック(例えば、CTU)を備えてよい。

【0066】

図2に表されたようなビデオエンコーダ20の実施形態は、タイルグループ(ビデオタイルグループとも呼ばれる)および／またはタイル(ビデオタイルとも呼ばれる)を使用することによってピクチャを区分し、かつ／またはエンコードするようにさらに構成されてよく、ピクチャは、(典型的に重複しない)1つまたは複数のタイルグループに区分され、またはそのタイルグループを使用してエンコードされてよく、各タイルグループは、例えば、1つまたは複数のブロック(例えば、CTU)または1つまたは複数のタイルを備えてよく、各タイルは、例えば、長方形形状であってよく、1つまたは複数のブロック(例えば、CTU)、例えば、完全なまたは断片のブロックを備えてよい。

【0067】

残差計算

残差計算ユニット204は、例えば、サンプルごとに(ピクセルごとに)ピクチャブロック203のサンプル値から予測ブロック265のサンプル値を減算することによって、ピクチャブロック203および予測ブロック265(予測ブロック265についてのさらなる詳細は後で提供される)に基づいて残差ブロック205(残差205とも呼ばれる)を計算して、サンプル領域における残差ブロック205を取得するように構成され得る。

【0068】

変換

変換処理ユニット206は、残差ブロック205のサンプル値において変換、例えば、離散コサイン変換(discrete cosine transform(DCT))または離散サイン変換(discrete sine transform(DST))を適用して、変換領域における変換係数207を取得するように構成され得る。変換係数207は、変換残差係数とも呼ばれ、変換領域における残差ブロック205を表現してよい。

【0069】

変換処理ユニット206は、H.265/HEVCについて指定された変換のようなDCT/DSTの整数近似を適用するように構成され得る。直交DCT変換と比較して、そのような整数近似は、典型的に、ある係数によってスケールされる。順および逆変換によって処理される残差ブロックのノルムを維持するために、変換プロセスの部分として追加のスケール係数が適用される。スケール係数は、典型的に、スケール係数がシフト演算のために2の累乗であること、変換係数のビット深度、確度と実装コストの間のトレードオフなどのような、ある制約に基づいて選ばれる。例えば、逆変換処理ユニット212による、例えば、逆変換(および、例えば、ビデオデコーダ30における逆変換処理ユニット312による、対応する逆変換)について特定のスケール係数が指定され、エンコーダ20における、例えば、変換処理ユニット206による、順変換について対応するスケール係数が、それに応じて指定され得る。

【0070】

ビデオエンコーダ20の実施形態(それぞれ、変換処理ユニット206)は、例えば、直接に、またはエントロピーエンコードユニット270を介してエンコードされ、または圧縮された、変換パラメータ、例えば、1つまたは複数の変換のタイプを出力するように構成されてよく、それによって、例えば、ビデオデコーダ30は、デコードのために変換パラメータを受信して使用し得る。

【0071】

量子化

量子化ユニット208は、例えば、スカラー量子化またはベクトル量子化を適用することによって、変換係数207を量子化して量子化された係数209を取得するように構成され得る。量子化された係数209は、量子化された変換係数209または量子化された残差係数209とも呼ばれ得る。

【0072】

量子化プロセスは、変換係数207のいくつかまたは全てに関連付けられたビット深度を減少させ得る。例えば、 n ビットの変換係数は、量子化の間に m ビットの変換係数に切り捨てられてよく、ここで n は m よりも大きい。量子化の程度は、量子化パラメータ(Quantization Parameter(QP))を調整することによって修正され得る。例えば、スカラー量子化について、より細かいかまたはより粗い量子化を達成するために、異なるスケールリングが適用され得る。より小さい量子化ステップサイズは、より細かい量子化に対応し、一方、より大きい量子化ステップサイズは、より粗い量子化に対応する。適用可能な量子化ステップサイズは、量子化パラメータ(QP)によって示され得る。量子化パラメータは、例えば、適用可能な量子化ステップサイズの予め定義されたセットへのインデックスであってよい。例えば、小さい量子化パラメータは、細かい量子化(小さい量子化ステップサイズ)に対応してよく、大きい量子化パラメータは、粗い量子化(大きい量子化ステップサイズ)に対応してよく、またはその逆も同様である。量子化は、量子化ステップサイズによる除算を含んでよく、例えば、逆量子化ユニット210による、対応するおおよび／または逆の逆量子化は、量子化ステップサイズによる乗算を含んでよい。いくつかの規格、例えば、HEVCによる実施形態は、量子化パラメータを使用して量子化ステップサイズを決定するように構成され得る。一般に、量子化ステップサイズは、除算を含む式の固定点近似を使用して、量子化パラメータに基づいて計算され得る。量子化ステップサイズおよび量子化パラメータについての式の固定点近似において使用されるスケールリングの故に修正され得る、残差ブロックのノルムを復元するために、量子化および逆量子化について追加のスケールリング係数が導入されてよい。1つの例示の実装では、逆変換および逆量子化のスケールリングが組み合わされ得る。代替として、カスタマイズされた量子化テーブルが使用され、例えば、ビットストリーム内で、エンコーダからデコーダにシグナリングされてよい。量子化は損失のある演算であり、増加する量子化ステップサイズとともに損失が増加する。

【0073】

ビデオエンコーダ20の実施形態(それぞれ、量子化ユニット208)は、例えば、直接に、またはエントロピーエンコードユニット270を介してエンコードされた、量子化パラメータ(QP)を出力するように構成されてよく、それによって、例えば、ビデオデコーダ30は、デコードのために量子化パラメータを受信し、適用し得る。

【0074】

逆量子化

逆量子化ユニット210は、例えば、量子化ユニット208と同じ量子化ステップサイズに基づいて、またはそれを使用して、量子化ユニット208によって適用される量子化方式の逆を適用することによって、量子化された係数において量子化ユニット208の逆量子化を適用して逆量子化された係数211を取得するように構成される。逆量子化された係数211は、逆量子化された残差係数211とも呼ばれ、量子化による損失に起因して典型的には変換係数と同一でないが、変換係数207に対応し得る。

【0075】

逆変換

逆変換処理ユニット212は、変換処理ユニット206によって適用される変換の逆変換、例えば、逆離散コサイン変換(DCT)または逆離散サイン変換(DST)、または他の逆変換を適用して、サンプル領域における再構成された残差ブロック213(または、対応する逆量子化された係数213)を取得するように構成される。再構成された残差ブロック213は、変換ブロック213とも呼ばれ得る。

【0076】

再構成

10

20

30

40

50

再構成ユニット214(例えば、加算器または合算器214)は、例えば、再構成された残差ブロック213のサンプル値と予測ブロック265のサンプル値を、サンプルごとに、加算することによって、変換ブロック213(すなわち、再構成された残差ブロック213)を予測ブロック265に加算してサンプル領域における再構成されたブロック215を取得するように構成される。

【0077】

フィルタ処理

ループフィルタユニット220(または、短縮して「ループフィルタ」220)は、再構成されたブロック215をフィルタ処理してフィルタ処理されたブロック221を取得するように、または一般に、再構成されたサンプルをフィルタ処理してフィルタ処理されたサンプルを取得するように構成される。ループフィルタユニットは、例えば、ピクセル遷移を平滑化し、またはそうでなくビデオ品質を改善するように構成される。ループフィルタユニット220は、デブロッキングフィルタ、サンプル適応オフセット(Sample-Adaptive Offset(SAO))フィルタ、または1つまたは複数の他のフィルタ、例えば、バイラテラルフィルタ、適応ループフィルタ(Adaptive Loop Filter(ALF))、鮮鋭化、平滑化フィルタ、または協調フィルタ、またはそれらの任意の組み合わせのような1つまたは複数のループフィルタを備えてよい。ループフィルタユニット220はループ内フィルタであるとして図2に表されているが、他の構成では、ループフィルタユニット220は、ポストループフィルタとして実現されてよい。フィルタ処理されたブロック221は、フィルタ処理された再構成されたブロック221とも呼ばれ得る。

【0078】

ビデオエンコーダ20の実施形態(それぞれ、ループフィルタユニット220)は、例えば、直接に、またはエン트로ピーエンコードユニット270を介してエンコードされた、(サンプル適応オフセット情報のような)ループフィルタパラメータを出力するように構成されてよく、それによって、例えば、デコーダ30は、デコードのために同じループフィルタパラメータまたはそれぞれのループフィルタを受信し、適用し得る。

【0079】

デコードされたピクチャバッファ

デコードされたピクチャバッファ(DPB)230は、ビデオエンコーダ20によってビデオデータをエンコードするための、参照ピクチャまたは一般に参照ピクチャデータを記憶するメモリであってよい。DPB 230は、同期DRAM(synchronous DRAM(SDRAM))を含むダイナミックランダムアクセスメモリ(dynamic random access memory(DRAM))、磁気抵抗RAM(magnetoresistive RAM(MRAM))、抵抗性RAM(resistive RAM(RRAM(登録商標)))、または他のタイプのメモリデバイスのような、様々なメモリデバイスのうちのいずれかによって形成され得る。デコードされたピクチャバッファ(DPB)230は、1つまたは複数のフィルタ処理されたブロック221を記憶するように構成され得る。デコードされたピクチャバッファ230は、以前にフィルタ処理された他のブロック、例えば、同じ現在のピクチャの、または異なるピクチャの、以前に再構成され、フィルタ処理されたブロック221、例えば、以前に再構成されたピクチャを記憶するようにさらに構成されてよく、例えば、インター予測のために、以前に再構成され、すなわちデコードされた完全なピクチャ(および、対応する参照ブロックおよびサンプル)、および／または部分的に再構成された現在のピクチャ(および、対応する参照ブロックおよびサンプル)を提供し得る。例えば、再構成されたブロック215が、ループフィルタユニット220によってフィルタ処理されていない、または再構成されたブロックまたはサンプルのさらに処理された任意の他のバージョンであるならば、デコードされたピクチャバッファ(DPB)230はまた、1つまたは複数のフィルタ処理されていない再構成されたブロック215、または一般に、フィルタ処理されていない再構成されたサンプルを記憶するように構成されてもよい。

【0080】

モード選択(区分および予測)

モード選択ユニット260は、区分ユニット262、インター予測ユニット244、およびイン

10

20

30

40

50

トラ予測ユニット254を備え、元のピクチャデータ、例えば、元のブロック203(現在のピクチャ17の現在のブロック203)、および同じ(現在の)ピクチャの、かつ／または1つまたは複数の以前にデコードされたピクチャからの、例えば、デコードされたピクチャバッファ230または他のバッファ(例えば、表されていないラインバッファ)からの、再構成されたピクチャデータ、例えば、フィルタ処理された、および／またはフィルタ処理されていない再構成されたサンプルまたはブロックを受信または取得するように構成される。再構成されたピクチャデータは、予測ブロック265または予測子265を取得するために、予測、例えば、インター予測またはイントラ予測のための参照ピクチャデータとして使用される。

【0081】

モード選択ユニット260は、(区分を含まない)現在のブロック予測モードについて区分、および予測モード(例えば、イントラまたはインター予測モード)を決定または選択し、残差ブロック205の計算のために、かつ再構成されたブロック215の再構成のために使用される、対応する予測ブロック265を生成するように構成され得る。

【0082】

モード選択ユニット260の実施形態は、最も良い整合、または言い換えれば最小残差(最小残差は、伝送または記憶のためのより良い圧縮を意味する)、または最小シグナリングオーバーヘッド(最小シグナリングオーバーヘッドとは、伝送または記憶のためのより良い圧縮を意味する)を提供し、または両方を考慮し、または釣り合わせる、区分および予測モードを(例えば、モード選択ユニット260によってサポートされ、またはモード選択ユニット260に対して利用可能なものから)選択するように構成され得る。モード選択ユニット260は、レート歪み最適化(Rate Distortion Optimization(RDO))に基づいて区分および予測モードを決定する、すなわち、最小のレート歪みを提供する予測モードを選択するように構成され得る。この文脈における「最も良い」、「最小の」、「最適な」などのような用語は、全体的な「最も良い」、「最小の」、「最適な」などを必ずしも指さず、しきい値または他の制約を上回るまたは下回る値のような、終了または選択の基準の充足を指してもよく、潜在的に「準最適な選択」に導くが、複雑さおよび処理時間を減少させる。

【0083】

言い換えれば、区分ユニット262は、例えば、4分木区分(quad-tree-partitioning(QT))、バイナリ区分(binary partitioning(BT))、またはトリプルツリー区分(triple-tree-partitioning(TT))、またはそれらの任意の組み合わせを反復的に使用して、ブロック203をより小さいブロック区分または(再びブロックを形成する)サブブロックに区分し、例えば、ブロック区分またはサブブロックの各々について予測を実行するように構成されてよく、モード選択は、区分されるブロック203のツリー構造の選択を備え、予測モードは、ブロック区分またはサブブロックの各々に適用される。

【0084】

以下では、例示のビデオエンコーダ20によって実行される(例えば、区分ユニット260による)区分および(インター予測ユニット244およびイントラ予測ユニット254による)予測処理が、より詳細に説明されるであろう。

【0085】

区分

区分ユニット262は、現在のブロック203をより小さい区分、例えば、正方形または長方形のサイズのより小さいブロックに、区分(または分割)し得る。これらのより小さいブロック(サブブロックとも呼ばれ得る)は、いっそう小さい区分にさらに区分され得る。これは、ツリー区分または階層的ツリー区分とも呼ばれ、例えば、ルートツリーレベル0(階層レベル0、深度0)におけるルートブロックは、再帰的に区分され、例えば、次に低いツリーレベルの2つ以上のブロック、例えば、ツリーレベル1(階層レベル1、深度1)におけるノードに区分されてよく、これらのブロックは再び、例えば、終了基準が充足され、例えば、最大ツリー深度または最小ブロックサイズが到達されたので区分が終了されるまで、次に低いレベル、例えば、ツリーレベル2(階層レベル2、深度2)などの2つ以上のブロックに

10

20

30

40

50

区分されてよい。さらに区分されないブロックは、ツリーのリーフブロックまたはリーフノードとも呼ばれる。2つの区分への区分を使用するツリーは2分木(Binary-Tree(BT))と呼ばれ、3つの区分への区分を使用するツリーは3分木(Ternary-Tree(TT))と呼ばれ、4つの区分への区分を使用するツリーは4分木(Quad-Tree(QT))と呼ばれる。

【0086】

前に述べたように、ここで使用される用語「ブロック」は、ピクチャの一部分、特に正方形または長方形の部分であってよい。例えば、HEVCおよびVVCを参照すると、ブロックは、符号化ツリーユニット(coding tree unit(CTU))、符号化ユニット(coding unit(CU))、予測ユニット(prediction unit(PU))、および変換ユニット(transform unit(TU))、および／または対応するブロック、例えば、符号化ツリーブロック(coding tree block(CTB))、符号化ブロック(coding block(CB))、変換ブロック(transform block(TB))、または予測ブロック(prediction block(PB))であってよく、またはそれらに対応し得る。

【0087】

例えば、符号化ツリーユニット(CTU)は、3つのサンプルアレイを有するピクチャのルーマサンプルのCTB、クロマサンプルの2つの対応するCTB、またはモノクロピクチャまたは3つの別個の色平面を使用して符号化されるピクチャのサンプルのCTB、およびサンプルを符号化するために使用されるシンタックス構造であってよく、またはそれらを備えてもよい。それに対応して、符号化ツリーブロック(CTB)は、CTBへの成分の分割が区分であるようなNのいくつかの値についてサンプルのN×Nブロックであってよい。符号化ユニット(CU)は、3つのサンプルアレイを有するピクチャのルーマサンプルの符号化ブロック、クロマサンプルの2つの対応する符号化ブロック、またはモノクロピクチャまたは3つの別個の色平面を使用して符号化されるピクチャのサンプルの符号化ブロック、およびサンプルを符号化するために使用されるシンタックス構造であってよく、またはそれらを備えてもよい。それに対応して、符号化ブロック(CB)は、符号化ブロックへのCTBの分割が区分であるようなMおよびNのいくつかの値についてサンプルのM×Nブロックであってよい。

【0088】

例えば、HEVCによる実施形態では、符号化ツリーユニット(CTU)は、符号化ツリーとして表記される4分木構造を使用することによってCUに分割され得る。ピクチャエリアを、(時間的な)インターピクチャ予測を使用して符号化すべきか、または(空間的な)イントラピクチャ予測を使用して符号化すべきかの決定は、CUレベルにおいて行われる。各CUは、PU分割タイプに従って1つ、2つ、または4つのPUにさらに分割されることが可能である。1つのPUの内部では、同じ予測プロセスが適用され、関連する情報がPUごとにデコーダへ伝送される。PU分割タイプに基づいて予測プロセスを適用することによって残差ブロックを取得した後、CUは、CUについての符号化ツリーと類似の別の4分木構造に従って変換ユニット(TU)に区分されることが可能である。

【0089】

例えば、多用途ビデオ符号化(VVC)と呼ばれる、現在策定中の最新のビデオ符号化規格による実施形態では、組み合わせられた4分木および2分木(Quad-Tree and Binary Tree(QTBT))区分が、例えば、符号化ブロックを区分するために使用される。QTBTブロック構造において、CUは、正方形または長方形のいずれかの形状を有することができる。例えば、符号化ツリーユニット(CTU)は、4分木構造によって最初に区分される。4分木リーフノードは、2分木または3分木(またはトリプルツリー)構造によってさらに区分される。区分するツリーリーフノードは、符号化ユニット(CU)と呼ばれ、そのセグメント化は、さらなる区分なしで予測および変換処理のために使用される。これは、CU、PU、およびTUがQTBT符号化ブロック構造において同じブロックサイズを有することを意味する。並行して、複数の区分、例えば、トリプルツリー区分が、QTBTブロック構造と一緒に使用され得る。

【0090】

一例では、ビデオエンコーダ20のモード選択ユニット260は、ここで説明される区分技法の任意の組み合わせを実行するように構成され得る。

【0091】

上記で説明されたように、ビデオエンコーダ20は、(例えば、予め決定された)予測モードのセットから、最も良いまたは最適な予測モードを決定または選択するように構成される。予測モードのセットは、例えば、イントラ予測モードおよび／またはインター予測モードを備えてよい。

【0092】

イントラ予測

イントラ予測モードのセットは、例えば、HEVCにおいて定義されるように、35個の異なるイントラ予測モード、例えば、DC(または平均)モードおよび平面モードのような無方向性モード、または方向性モードを備えてよく、または、例えば、VVCについて定義されるように、67個の異なるイントラ予測モード、例えば、DC(または平均)モードおよび平面モードのような無方向性モード、または方向性モードを備えてよい。

10

【0093】

イントラ予測ユニット254は、イントラ予測モードのセットのイントラ予測モードに従ってイントラ予測ブロック265を生成するために、同じ現在のピクチャの隣接ブロックの再構成されたサンプルを使用するように構成される。

【0094】

イントラ予測ユニット254(または一般にモード選択ユニット260)は、エンコードされたピクチャデータ21への包含のためにシンタックス要素266の形式でイントラ予測パラメータ(または一般に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троピーエンコードユニット270に出力するようにさらに構成され、それによって、例えば、ビデオデコーダ30は、デコードのために予測パラメータを受信および使用し得る。

20

【0095】

インター予測

インター予測モードのセット(または可能なインター予測モード)は、利用可能な参照ピクチャ(すなわち、例えば、DBP 230に記憶されている、少なくとも部分的にデコードされた以前のピクチャ)、および他のインター予測パラメータ、例えば、最も良く整合する参照ブロックを探索するために参照ピクチャ全体が使用されるか、または参照ピクチャの一部のみ、例えば、現在のブロックのエリアの周囲の探索ウィンドウエリアが使用されるか、および／または、例えば、ピクセル補間、例えば、ハーフ／セミペルおよび／またはクォーターペル補間が適用されるか否かに依存する。

30

【0096】

上記の予測モードに加えて、スキップモードおよび／または直接モードが適用されてよい。

【0097】

インター予測ユニット244は、動き推定(motion estimation(ME))ユニットおよび動き補償(motion compensation(MC))ユニット(両方とも図2に表されていない)を含んでよい。動き推定ユニットは、動き推定のために、ピクチャブロック203(現在のピクチャ17の現在のピクチャブロック203)およびデコードされたピクチャ231、または少なくとも1つまたは複数の以前に再構成されたブロック、例えば、1つまたは複数の他の／異なる以前にデコードされたピクチャ231の再構成されたブロックを、受信または取得するように構成され得る。例えば、ビデオシーケンスは、現在のピクチャおよび以前にデコードされたピクチャ231を備えてよく、または言い換えれば、現在のピクチャおよび以前にデコードされたピクチャ231は、ビデオシーケンスを形成するピクチャのシーケンスの一部であってよく、またはそれらを形成してよい。

40

【0098】

エンコーダ20は、例えば、複数の他のピクチャのうちの同じまたは異なるピクチャの複数の参照ブロックから参照ブロックを選択し、参照ピクチャ(または参照ピクチャインデックス)、および／または参照ブロックの位置(x、y座標)と現在のブロックの位置との間のオフセット(空間オフセット)を、インター予測パラメータとして動き推定ユニットに提供するように構成され得る。このオフセットは動きベクトル(motion vector(MV))とも呼ばれる。

。

50

【0099】

動き補償ユニットは、インター予測パラメータを取得し、例えば受信し、インター予測パラメータに基づいて、またはそれを使用して、インター予測を実行してインター予測ブロック265を取得するように構成される。動き補償ユニットによって実行される動き補償は、動き推定によって決定された動き／ブロック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、おそらくサブピクセル精度への補間を実行することを伴ってよい。補間フィルタ処理は、知られているピクセルサンプルから追加のピクセルサンプルを生成してよく、従って、ピクチャブロックを符号化するために使用され得る候補予測ブロックの数を潜在的に増加させる。現在のピクチャブロックのPUについての動きベクトルを受信すると、動き補償ユニットは、参照ピクチャリストのうちの1つの中で動きベクトルが指し示す予測ブロックを位置付け得る。

10

【0100】

動き補償ユニットはまた、ビデオスライスのピクチャブロックをデコードする際のビデオデコーダ30による使用のために、ブロックおよびビデオスライスに関連付けられたシンタックス要素を生成し得る。スライスおよびそれぞれのシンタックス要素に加えて、またはそれらの代替として、タイルグループおよび／またはタイルおよびそれぞれのシンタックス要素が生成または使用されてよい。

【0101】

エントロピー符号化

エントロピーエンコードユニット270は、量子化された係数209、インター予測パラメータ、イントラ予測パラメータ、ループフィルタパラメータ、および／または他のシンタックス要素において、例えば、エントロピーエンコードアルゴリズムまたは方式(例えば、可変長符号化(variable length coding(VLC))方式、コンテキスト適応VLC方式(context adaptive VLC scheme(CAVLC))、算術符号化方式、2値化、コンテキスト適応バイナリ算術符号化(context adaptive binary arithmetic coding(CABAC))、シンタックスベースのコンテキスト適応バイナリ算術符号化(syntax-based context-adaptive binary arithmetic coding(SBAC))、確率区間区分エントロピー(probability interval partitioning entropy(PIPE))符号化、または別のエントロピーエンコード方法論または技法)、またはバイパス(圧縮なし)を適用して、例えば、エンコードされたビットストリーム21の形式で、出力272を介して出力されることが可能であるエンコードされたピクチャデータ21を取得するように構成され、それによって、例えば、ビデオデコーダ30は、デコードのためにパラメータを受信および使用し得る。エンコードされたビットストリーム21は、ビデオデコーダ30へ伝送され、またはビデオデコーダ30による後の伝送または取り出しのためにメモリに記憶され得る。

20

30

【0102】

ビデオエンコーダ20の他の構造的変形が、ビデオストリームをエンコードするために使用されることが可能である。例えば、非変換ベースのエンコーダ20が、あるブロックまたはフレームについて変換処理ユニット206なしで直接に残差信号を量子化することができる。別の実装では、エンコーダ20は、単一のユニットの中に組み合わせられた量子化ユニット208および逆量子化ユニット210を有することができる。

40

【0103】

デコーダおよびデコード方法

図3は、この本出願の技法を実現するように構成されるビデオデコーダ30の一例を表す。ビデオデコーダ30は、例えば、エンコーダ20によってエンコードされた、エンコードされたピクチャデータ21(例えば、エンコードされたビットストリーム21)を受信して、デコードされたピクチャ331を取得するように構成される。エンコードされたピクチャデータまたはビットストリームは、エンコードされたピクチャデータをデコードするための情報、例えば、エンコードされたビデオスライス(および／または、タイルグループまたはタイル)のピクチャブロックを表現するデータ、および関連付けられたシンタックス要素を備える。

50

【0104】

図3の例では、デコーダ30は、エントロピーデコードユニット304、逆量子化ユニット310、逆変換処理ユニット312、再構成ユニット314(例えば、合算器314)、ループフィルタ320、デコードされたピクチャバッファ(DBP)330、モード適用ユニット360、インター予測ユニット344、およびイントラ予測ユニット354を備える。インター予測ユニット344は、動き補償ユニットであってよく、またはそれを含んでもよい。ビデオデコーダ30は、いくつかの例では、図2からのビデオエンコーダ100に関して説明されたエンコードパスとは一般に相補的なデコードパスを実行し得る。

【0105】

エンコーダ20に関して説明されたように、逆量子化ユニット210、逆変換処理ユニット212、再構成ユニット214、ループフィルタ220、デコードされたピクチャバッファ(DPB)230、インター予測ユニット344、およびイントラ予測ユニット354は、ビデオエンコーダ20の「ビルトインデコーダ」を形成するとしても言及される。従って、逆量子化ユニット310は逆量子化ユニット110と機能において同一であってよく、逆変換処理ユニット312は逆変換処理ユニット212と機能において同一であってよく、再構成ユニット314は再構成ユニット214と機能において同一であってよく、ループフィルタ320はループフィルタ220と機能において同一であってよく、デコードされたピクチャバッファ330はデコードされたピクチャバッファ230と機能において同一であってよい。従って、ビデオ20エンコーダのそれぞれのユニットおよび機能について提供された説明は、それに対応してビデオデコーダ30のそれぞれのユニットおよび機能に適用される。

【0106】

エントロピーデコード

エントロピーデコードユニット304は、ビットストリーム21(または一般にエンコードされたピクチャデータ21)を構文解析し、例えば、エンコードされたピクチャデータ21へのエントロピーデコードを実行して、例えば、量子化された係数309および／またはデコードされた符号化パラメータ(図3に表されていない)、例えば、インター予測パラメータ(例えば、参照ピクチャインデックスおよび動きベクトル)、イントラ予測パラメータ(例えば、イントラ予測モードまたはインデックス)、変換パラメータ、量子化パラメータ、ループフィルタパラメータ、および／または他のシンタックス要素のうちのいずれかまたは全てを取得するように構成される。エントロピーデコードユニット304は、エンコーダ20のエントロピーエンコードユニット270に関して説明されたようなエンコード方式に対応するデコードアルゴリズムまたは方式を適用するように構成され得る。エントロピーデコードユニット304は、モード適用ユニット360にインター予測パラメータ、イントラ予測パラメータ、および／または他のシンタックス要素を、デコーダ30の他のユニットに他のパラメータを提供するようにさらに構成され得る。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においてシンタックス要素を受信し得る。スライスおよびそれぞれのシンタックス要素に加えて、またはそれらの代替として、タイルグループおよび／またはタイルおよびそれぞれのシンタックス要素が受信および／または使用され得る。

【0107】

逆量子化

逆量子化ユニット310は、(例えば、エントロピーデコードユニット304によって、例えば、構文解析および／またはデコードすることによって)エンコードされたピクチャデータ21から量子化パラメータ(quantization parameter(QP))(または一般に逆量子化に関する情報)および量子化された係数を受信し、量子化パラメータに基づいて、デコードされた量子化された係数309において逆量子化を適用して、変換係数311とも呼ばれ得る逆量子化された係数311を取得するように構成され得る。逆量子化プロセスは、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(またはタイルまたはタイルグループ)の中のビデオブロックごとにビデオエンコーダ20によって決定された量子化パラメータの使用を含んでよい。

【0108】

逆変換

逆変換処理ユニット312は、変換係数311とも呼ばれる逆量子化された係数311を受信し、サンプル領域において再構成された残差ブロック213を取得するために逆量子化された係数311に変換を適用するように構成され得る。再構成された残差ブロック213は、変換ブロック313とも呼ばれ得る。変換は、逆変換、例えば、逆DCT、逆DST、逆整数変換、または概念的に類似の逆変換プロセスであってよい。逆変換処理ユニット312は、逆量子化された係数311に適用されるべき変換を決定するために、(例えば、エントロピーデコードユニット304によって、例えば、構文解析および／またはデコードすることによって)エンコードされたピクチャデータ21から変換パラメータまたは対応する情報を受信するようにさらに構成され得る。

10

【0109】

再構成

再構成ユニット314(例えば、加算器または合算器314)は、例えば、再構成された残差ブロック313のサンプル値と予測ブロック365のサンプル値とを加算することによって、予測ブロック365に再構成された残差ブロック313を加算して、サンプル領域において再構成されたブロック315を取得するように構成され得る。

【0110】

フィルタ処理

(符号化ループ内または符号化ループの後のいずれかの)ループフィルタユニット320は、例えば、ピクセル遷移を平滑化し、またはビデオ品質をそうでなく改善するために、再構成されたブロック315をフィルタ処理してフィルタ処理されたブロック321を取得するように構成される。ループフィルタユニット320は、デブロッキングフィルタ、サンプル適応オフセット(sample-adaptive offset(SAO))フィルタ、または1つまたは複数の他のフィルタ、例えば、バイラテラルフィルタ、適応ループフィルタ(adaptive loop filter(ALF))、鮮鋭化、平滑化フィルタ、または協調フィルタ、またはそれらの任意の組み合わせのような1つまたは複数のループフィルタを備えてよい。ループフィルタユニット320はループ内フィルタであるとして図3に表されているが、他の構成では、ループフィルタユニット320はポストループフィルタとして実現されてもよい。

20

【0111】

デコードされたピクチャバッファ

ピクチャのデコードされたビデオブロック321は、次いで、他のピクチャについての後続の動き補償のために、かつ／または出力されるそれぞれ表示のために、参照ピクチャとしてデコードされたピクチャ331を記憶するデコードされたピクチャバッファ330に記憶される。

30

【0112】

デコーダ30は、ユーザへの提示または閲覧のために、例えば、出力312を介してデコードされたピクチャ311を出力するように構成される。

【0113】

予測

インター予測ユニット344は、インター予測ユニット244と(特に動き補償ユニットと)同一であってよく、イントラ予測ユニット354は、機能においてインター予測ユニット254と同一であってよく、区分および／または予測パラメータ、またはエンコードされたピクチャデータ21から(例えば、エントロピーデコードユニット304によって、例えば、構文解析および／またはデコードすることによって)受信されたそれぞれの情報に基づいて、分割または区分決定および予測を実行する。モード適用ユニット360は、(フィルタ処理された、またはフィルタ処理されていない)再構成されたピクチャ、ブロック、またはそれぞれのサンプルに基づいて、ブロックごとに予測(イントラまたはインター予測)を実行して、予測ブロック365を取得するように構成され得る。

40

【0114】

50

イントラ符号化された(I)スライスとしてビデオスライスが符号化されるとき、モード適用ユニット360のイントラ予測ユニット354は、シグナリングされたイントラ予測モード、および現在のピクチャの以前にデコードされたブロックからのデータに基づいて、現在のビデオスライスのピクチャブロックについて予測ブロック365を生成するように構成される。インター符号化された(すなわち、BまたはP)スライスとしてビデオピクチャが符号化されるとき、モード適用ユニット360のインター予測ユニット344(例えば、動き補償ユニット)は、動きベクトル、およびエントロピーデコードユニット304から受信された他のシンタックス要素に基づいて、現在のビデオスライスのビデオブロックについて予測ブロック365を作り出すように構成される。インター予測について、予測ブロックは、参照ピクチャリストのうちの1つの中の参照ピクチャのうちの1つから作り出されてよい。ビデオデコーダ30は、DPB 330に記憶された参照ピクチャに基づいてデフォルトの構成技法を使用して、参照フレームリスト、リスト0およびリスト1を構成し得る。スライス(例えば、ビデオスライス)に加えて、またはその代替として、タイルグループ(例えば、ビデオタイルグループ)および／またはタイル(例えば、ビデオタイル)を使用する実施形態について、またはその実施形態によって、同じまたは類似のことが適用されてよく、例えば、ビデオは、I、P、またはBタイルグループおよび／またはタイルを使用して符号化されてよい。

【0115】

モード適用ユニット360は、動きベクトルまたは関連する情報および他のシンタックス要素を構文解析することによって、現在のビデオスライスのビデオブロックについて予測情報を決定し、予測情報を使用して、デコードされている現在のビデオブロックについて予測ブロックを作り出すように構成される。例えば、モード適用ユニット360は、受信されたシンタックス要素のうちのいくつかを使用して、現在のビデオスライス内のビデオブロックをデコードするために、ビデオスライスのビデオブロックを符号化するために使用された予測モード(例えば、イントラまたはインター予測)、インター予測スライスタイプ(例えば、Bスライス、Pスライス、またはGPBスライス)、スライスについての参照ピクチャリストのうちの1つまたは複数についての構成情報、スライスのインターエンコードされたビデオブロックごとの動きベクトル、スライスのインター符号化されたビデオブロックごとのインター予測ステータス、および他の情報を決定する。スライス(例えば、ビデオスライス)に加えて、またはその代替として、タイルグループ(例えば、ビデオタイルグループ)および／またはタイル(例えば、ビデオタイル)を使用する実施形態について、またはその実施形態によって、同じまたは類似のことが適用されてよく、例えば、ビデオは、I、P、またはBタイルグループおよび／またはタイルを使用して符号化されてよい。

【0116】

図3に表されたようなビデオデコーダ30の実施形態は、スライス(ビデオスライスとも呼ばれる)を使用することによってピクチャを区分および／またはデコードするように構成されてよく、ピクチャは、(典型的に、重複しない)1つまたは複数のスライスに区分され、またはそれを使用してデコードされてよく、各スライスは、1つまたは複数のブロック(例えば、CTU)を備えてよい。

【0117】

図3に表されたようなビデオデコーダ30の実施形態は、タイルグループ(ビデオタイルグループとも呼ばれる)および／またはタイル(ビデオタイルとも呼ばれる)を使用することによってピクチャを区分および／またはデコードするように構成されてよく、ピクチャは、(典型的に、重複しない)1つまたは複数のタイルグループに区分され、またはそれを使用してデコードされてよく、各タイルグループは、例えば、1つまたは複数のブロック(例えば、CTU)または1つまたは複数のタイルを備えてよく、各タイルは、例えば、長方形形状であってよく、1つまたは複数のブロック(例えば、CTU)、例えば、完全なまたは断片のブロックを備えてよい。

【0118】

ビデオデコーダ30の他の変形が、エンコードされたピクチャデータ21をデコードするために使用されることが可能である。例えば、デコーダ30は、ループフィルタ処理ユニット

10

20

30

40

50

320なしで出力ビデオストリームを作り出すことができる。例えば、非変換ベースのデコーダ30が、あるブロックまたはフレームについて逆変換処理ユニット312なしで直接に残差信号を逆量子化することができる。別の実装では、ビデオデコーダ30は、単一のユニットの中に組み合わせられた逆量子化ユニット310および逆変換処理ユニット312を有することができる。

【0119】

エンコーダ20およびデコーダ30において、現在のステップの処理結果がさらに処理され、次いで、次のステップに出力されてよいことが理解されるべきである。例えば、補間フィルタ処理、動きベクトル導出、またはループフィルタ処理の後、補間フィルタ処理、動きベクトル導出、またはループフィルタ処理の処理結果において、クリップまたはシフト

10

【0120】

現在のブロックの導出された動きベクトル(アフィンモードの制御点動きベクトル、アフィン、平面、ATMVPモードにおけるサブブロック動きベクトル、時間動きベクトルなどを含むが、それらに限定されない)に、さらなる演算が適用され得ることが注記されるべきである。例えば、動きベクトルの値は、その表現ビットに従って予め定義された範囲に制約される。動きベクトルの表現ビットがbitDepthであるならば、範囲は $-2^{(bitDepth-1)} \sim 2^{(bitDepth-1)}-1$ であり、ここで「 $\wedge$ 」はべき乗を意味する。例えば、bitDepthが16に等しく設定されるならば、範囲は-32768~32767であり、bitDepthが18に等しく設定されるならば、範囲は-131072~131071である。例えば、導出される動きベクトル(例えば、1つの8×8ブロック内の4つの4×4サブブロックのMV)の値は、4つの4×4サブブロックMVの整数部分の間の最大差が、1ピクセルより大きくない、のような、Nピクセルより大きくないように制約される。ここで、bitDepthに従って動きベクトルを制約するための2つの方法を提供する。

20

【0121】

方法1:フロー演算によってオーバーフローMSB(最上位ビット)を除去する。

$$ux = (mvx + 2^{bitDepth}) \% 2^{bitDepth} \quad (1)$$

$$mvx = (ux > 2^{bitDepth-1}) ? (ux - 2^{bitDepth}) : ux \quad (2)$$

$$uy = (mvy + 2^{bitDepth}) \% 2^{bitDepth} \quad (3)$$

$$mvy = (uy > 2^{bitDepth-1}) ? (uy - 2^{bitDepth}) : uy \quad (4)$$

30

ここで、mvxは画像ブロックまたはサブブロックの動きベクトルの水平成分であり、mvyは画像ブロックまたはサブブロックの動きベクトルの垂直成分であり、uxおよびuyは中間値を示す。

【0122】

例えば、mvxの値が-32769であるならば、式(1)および(2)を適用した後、結果として生じる値は32767である。コンピュータシステムでは、10進数は2の補数として記憶される。-32769の2の補数は1,0111,1111,1111,1111(17ビット)であり、次いで、MSBが廃棄され、そのため、結果として生じる2の補数は、式(1)および(2)を適用することによる出力と同じである0111,1111,1111,1111である(10進数は32767である)。

$$ux = (mvp_x + mvd_x + 2^{bitDepth}) \% 2^{bitDepth} \quad (5)$$

40

$$mvx = (ux > 2^{bitDepth-1}) ? (ux - 2^{bitDepth}) : ux \quad (6)$$

$$uy = (mvp_y + mvd_y + 2^{bitDepth}) \% 2^{bitDepth} \quad (7)$$

$$mvy = (uy > 2^{bitDepth-1}) ? (uy - 2^{bitDepth}) : uy \quad (8)$$

【0123】

式(5)~(8)に表されたように、その演算は、mvpとmvdの合算の間に適用され得る。

【0124】

方法2:値をクリッピングすることによってオーバーフローMSBを除去する。

$$vx = \text{Clip3}(-2^{bitDepth-1}, 2^{bitDepth-1}-1, vx)$$

$$vy = \text{Clip3}(-2^{bitDepth-1}, 2^{bitDepth-1}-1, vy)$$

ここで、vxは画像ブロックまたはサブブロックの動きベクトルの水平成分であり、vyは画

50

像ブロックまたはサブブロックの動きベクトルの垂直成分であり、 x 、 y 、および z はそれぞれ、MVクリッピングプロセスの3つの入力値に対応し、関数Clip3の定義は以下の通りである。

【0125】

【数1】

$$\text{Clip3}(x, y, z) = \begin{cases} x & ; \quad z < x \\ y & ; \quad z > y \\ z & ; \quad \text{その他} \end{cases}$$

10

【0126】

図4は、開示の一実施形態によるビデオ符号化デバイス400の概略図である。ビデオ符号化デバイス400は、ここで説明されるような開示される実施形態を実現するために適している。一実施形態では、ビデオ符号化デバイス400は、図1Aのビデオデコーダ30のようなデコーダ、または図1Aのビデオエンコーダ20のようなエンコーダであってよい。

【0127】

ビデオ符号化デバイス400は、データを受信するための入口ポート410(または入力ポート410)および受信機ユニット(Rx)420、データ処理するためのプロセッサ、論理ユニット、または中央処理ユニット(CPU)430、データを伝送するための送信機ユニット(Tx)440および出口ポート450(または出力ポート450)、およびデータを記憶するためのメモリ460を備える。ビデオ符号化デバイス400はまた、光信号または電気信号の出口または入口のために入口ポート410、受信機ユニット420、送信機ユニット440、および出口ポート450に結合された、光電気(optical-to-electrical(OE))構成要素および電気光(electrical-to-optical(EO))構成要素を備えてよい。

20

【0128】

プロセッサ430は、ハードウェアおよびソフトウェアによって実現される。プロセッサ430は、1つまたは複数のCPUチップ、(例えば、マルチコアプロセッサとしての)コア、FPGA、ASIC、およびDSPとして実現され得る。プロセッサ430は、入口ポート410、受信機ユニット420、送信機ユニット440、出口ポート450、およびメモリ460と通信している。プロセッサ430は、符号化モジュール470を備える。符号化モジュール470は、上記で説明された開示された実施形態を実現する。例えば、符号化モジュール470は、様々な符号化演算を実現、処理、準備、または提供する。従って、符号化モジュール470の包含は、ビデオ符号化デバイス400の機能にかなりの改善を提供し、ビデオ符号化デバイス400の異なる状態への変換をもたらす。代替として、符号化モジュール470は、メモリ460に記憶され、プロセッサ430によって実行される命令として実現される。

30

【0129】

メモリ460は、1つまたは複数のディスク、テープドライブ、およびソリッドステートドライブを備えてよく、実行のためにそのようなプログラムが選択されるときにプログラムを記憶するために、かつプログラム実行の間に読み取られる命令およびデータを記憶するために、オーバーフローデータ記憶デバイスとして使用されてよい。メモリ460は、例えば、揮発性および／または不揮発性であってよく、リードオンリメモリ(read-only memory(ROM))、ランダムアクセスメモリ(random access memory(RAM))、3値連想メモリ(ternary content-addressable memory(TCAM))、および／またはスタティックランダムアクセスメモリ(static random-access memory(SRAM))であってよい。

40

【0130】

図5は、例示的な実施形態による、図1からのソースデバイス12および宛先デバイス14のいずれかまたは両方として使用され得る装置500の簡略化されたブロック図である。

【0131】

50

装置500内のプロセッサ502は中央処理ユニットであることが可能である。代替として、プロセッサ502は、現存する、または今後開発される、情報を操作または処理することが可能な任意の他のタイプのデバイスまたは複数のデバイスであることが可能である。開示される実装は、表されているように単一のプロセッサ、例えば、プロセッサ502を用いて実施されることが可能であるが、速度および効率における利点は、1つより多くのプロセッサを使用して達成されることが可能である。

【0132】

装置500内のメモリ504は、一実装ではリードオンリメモリ(ROM)デバイスまたはランダムアクセスメモリ(RAM)デバイスであることが可能である。任意の他の適したタイプの記憶デバイスが、メモリ504として使用されることが可能である。メモリ504は、バス512を使用してプロセッサ502によってアクセスされるコードおよびデータ506を含むことができる。メモリ504は、オペレーティングシステム508およびアプリケーションプログラム510をさらに含むことができ、アプリケーションプログラム510は、プロセッサ502がここで説明される方法を実行することを可能にする少なくとも1つのプログラムを含む。例えば、アプリケーションプログラム510は、アプリケーション1からNを含むことができ、アプリケーション1からNは、ここで説明される方法を実行するビデオ符号化アプリケーションをさらに含む。

【0133】

装置500はまた、ディスプレイ518のような1つまたは複数の出力デバイスを含むことができる。ディスプレイ518は、一例では、ディスプレイを、タッチ入力感知するように動作可能なタッチ感応性要素と組み合わせる、タッチ感応性ディスプレイであってよい。ディスプレイ518は、バス512を介してプロセッサ502に結合されることが可能である。

【0134】

ここでは単一のバスとして描写されているが、装置500のバス512は複数のバスからなることが可能である。さらに、2次記憶装置514が、装置500の他の構成要素に直接に結合されることが可能であり、またはネットワークを介してアクセスされることが可能であり、メモ리카ードのような単一の統合されたユニット、または複数のメモ리카ードのような複数のユニットを備えることができる。従って、装置500は広く様々な構成で実現されることが可能である。

【0135】

VVCドラフト4.0(JVET-M1001)では、変換コアDST-VII(DST 7)、DCT-VIII(DCT 8)、およびDCT-II(DCT 2)が、変換プロセスについて開示されている。

【0136】

HEVCにおいて採用されてきたDCT-IIに加えて、インターおよびイントラ符号化ブロックの両方についての残差符号化のために、多重変換選択(Multiple Transform Selection(MTS))方式が使用される。MTS方式では多重選択された変換DCT8またはDST7が使用される。HEVCと比較して、DST-VIIおよびDCT-VIIIは新たに導入された変換コアである。表1(Table 1)は、選択される変換コアの基本関数を表す。通常、変換コアは、変換プロセスのために使用される係数のセットを備える。変換コアは、公式化によって示されてよく、または変換コアは、数学的に定義された関数からの数学的に丸められた係数を備えるテーブルまたは行列によって示されてよい。丸めは、ハードウェア設計を簡略化することを目的としている(例えば、丸めの後、より小さいサイズ(例えば、 4×4)の変換コアは、より大きいサイズの変換コア(例えば、 16×16)から直接にダウンサンプリングすることができる)。丸めの後、係数は依然として変換の数学的神格化に従う。

【0137】

10

20

30

40

50

【表 1】

表 1 - N ポイント入力についての DCT-II/VIII および DST-VII の変換基底関数

変換コア	基底関数 $T_i(j)$, $i, j = 0, 1, \dots, N-1$
DCT-II	$T_i(j) = \omega_0 \cdot \sqrt{\frac{2}{N}} \cdot \cos\left(\frac{\pi \cdot i \cdot (2j + 1)}{2N}\right)$ ここで、$\omega_0 = \begin{cases} \sqrt{\frac{2}{N}} & i = 0 \\ 1 & i \neq 0 \end{cases}$
DCT-VIII	$T_i(j) = \sqrt{\frac{4}{2N + 1}} \cdot \cos\left(\frac{\pi \cdot (2i + 1) \cdot (2j + 1)}{4N + 2}\right)$
DST-VII	$T_i(j) = \sqrt{\frac{4}{2N + 1}} \cdot \sin\left(\frac{\pi \cdot (2i + 1) \cdot (j + 1)}{2N + 1}\right)$

10

20

【 0 1 3 8 】

別の例では、これらの変換コアは変換行列によって示される。これらの例では、変数 nTbs が、スケーリングされた変換係数の変換ブロックサイズまたは水平サンプルサイズを指定する。

DST-VII:

- nTbs が 4 に等しいならば、以下を適用する。

transMatrix[m][n] = (1206)

```
{
{ 29  55  74  84 }
{ 74  74   0 -74 }
{ 84 -29 -74  55 }
{ 55 -84  74 -29 }
},
```

30

- そうでなく、nTbs が 8 に等しいならば、以下を適用する。

transMatrix[m][n] = (1207)

```
{
{ 17  32  46  60  71  78  85  86 }
{ 46  78  86  71  32 -17 -60 -85 }
{ 71  85  32 -46 -86 -60  17  78 }
{ 85  46 -60 -78  17  86  32 -71 }
{ 86 -17 -85  32  78 -46 -71  60 }
{ 78 -71 -17  85 -60 -32  86 -46 }
{ 60 -86  71 -17 -46  85 -78  32 }
{ 32 -60  78 -86  85 -71  46 -17 }
},
```

40

- そうでなく、nTbs が 16 に等しいならば、以下を適用する。

transMatrix[m][n] = (1208)

```
{
{ 8  17  25  33  40  48  55  62  68  73  77  81  85  87  88  88 }
{ 25  48  68  81  88  88  81  68  48  25  0 -25 -48 -68 -81 -88 }
```

50

```

{ 40 73 88 85 62 25 -17 -55 -81 -88 -77 -48 -8 33 68 87 }
{ 55 87 81 40 -17 -68 -88 -73 -25 33 77 88 62 8 -48 -85 }
{ 68 88 48 -25 -81 -81 -25 48 88 68 0 -68 -88 -48 25 81 }
{ 77 77 0 -77 -77 0 77 77 0 -77 -77 0 77 77 0 -77 }
{ 85 55 -48 -87 -8 81 62 -40 -88 -17 77 68 -33 -88 -25 73 }
{ 88 25 -81 -48 68 68 -48 -81 25 88 0 -88 -25 81 48 -68 }
{ 88 -8 -88 17 87 -25 -85 33 81 -40 -77 48 73 -55 -68 62 }
{ 87 -40 -68 73 33 -88 8 85 -48 -62 77 25 -88 17 81 -55 }
{ 81 -68 -25 88 -48 -48 88 -25 -68 81 0 -81 68 25 -88 48 }
{ 73 -85 25 55 -88 48 33 -87 68 8 -77 81 -17 -62 88 -40 }
{ 62 -88 68 -8 -55 88 -73 17 48 -87 77 -25 -40 85 -81 33 }
{ 48 -81 88 -68 25 25 -68 88 -81 48 0 -48 81 -88 68 -25 }
{ 33 -62 81 -88 85 -68 40 -8 -25 55 -77 88 -87 73 -48 17 }
{ 17 -33 48 -62 73 -81 87 -88 88 -85 77 -68 55 -40 25 -8 }
},

```

10

- そうでなく、nTbsが32に等しいならば、以下を適用する。

transMatrix[m][n] = transMatrixCol0to15[m][n] ここでm=0..15、n=0..15
(1209)

transMatrixCol0to15 =

(1210)

```

{
{ 4 9 13 17 21 26 30 34 38 42 46 50 53 56 60 63 }
{ 13 26 38 50 60 68 77 82 86 89 90 88 85 80 74 66 }
{ 21 42 60 74 84 89 89 84 74 60 42 21 0 -21 -42 -60 }
{ 30 56 77 87 89 80 63 38 9 -21 -50 -72 -85 -90 -84 -68 }
{ 38 68 86 88 74 46 9 -30 -63 -84 -90 -78 -53 -17 21 56 }
{ 46 78 90 77 42 -4 -50 -80 -90 -74 -38 9 53 82 89 72 }
{ 53 85 85 53 0 -53 -85 -85 -53 0 53 85 85 53 0 -53 }
{ 60 89 74 21 -42 -84 -84 -42 21 74 89 60 0 -60 -89 -74 }
{ 66 90 56 -13 -74 -87 -46 26 80 84 34 -38 -85 -78 -21 50 }
{ 72 86 34 -46 -89 -63 13 78 82 21 -56 -90 -53 26 84 77 }
{ 77 80 9 -72 -84 -17 66 86 26 -60 -88 -34 53 90 42 -46 }
{ 80 72 -17 -86 -60 34 90 46 -50 -89 -30 63 85 13 -74 -78 }
{ 84 60 -42 -89 -21 74 74 -21 -89 -42 60 84 0 -84 -60 42 }
{ 86 46 -63 -78 21 90 26 -77 -66 42 87 4 -85 -50 60 80 }
{ 88 30 -78 -56 60 77 -34 -87 4 89 26 -80 -53 63 74 -38 }
{ 90 13 -87 -26 84 38 -78 -50 72 60 -63 -68 53 77 -42 -82 }
},

```

20

30

transMatrix[m][n] = transMatrixCol16to31[m - 16][n] ここでm=16..31、
n=0..15 (1211)

transMatrixCol16to31 =

(1212)

```

{
{ 66 68 72 74 77 78 80 82 84 85 86 87 88 89 90 90 }
{ 56 46 34 21 9 -4 -17 -30 -42 -53 -63 -72 -78 -84 -87 -90 }
{ -74 -84 -89 -89 -84 -74 -60 -42 -21 0 21 42 60 74 84 89 }
{ -46 -17 13 42 66 82 90 86 74 53 26 -4 -34 -60 -78 -88 }
{ 80 90 82 60 26 -13 -50 -77 -89 -85 -66 -34 4 42 72 87 }
{ 34 -13 -56 -84 -88 -68 -30 17 60 85 87 66 26 -21 -63 -86 }
{ -85 -85 -53 0 53 85 85 53 0 -53 -85 -85 -53 0 53 85 }
{ -21 42 84 84 42 -21 -74 -89 -60 0 60 89 74 21 -42 -84 }
{ 88 72 9 -60 -90 -63 4 68 89 53 -17 -77 -86 -42 30 82 }
}

```

40

50

```

{ 9 -66 -88 -42 38 87 68 -4 -74 -85 -30 50 90 60 -17 -80 }
{ -90 -50 38 89 56 -30 -87 -63 21 85 68 -13 -82 -74 4 78 }
{ 4 82 68 -21 -87 -56 38 90 42 -53 -88 -26 66 84 9 -77 }
{ 89 21 -74 -74 21 89 42 -60 -84 0 84 60 -42 -89 -21 74 }
{ -17 -90 -30 74 68 -38 -88 -9 84 53 -56 -82 13 89 34 -72 }
{ -86 9 90 21 -82 -50 66 72 -42 -85 13 90 17 -84 -46 68 }
{ 30 86 -17 -89 4 90 9 -88 -21 85 34 -80 -46 74 56 -66 }
},

```

ここで、nTbsは変換ブロックサイズである。VVCは最大32ポイントのDST-VIIのみをサポートするので、必要な最大の行列は32×32である。

10

DCT-II:

transMatrix[m][n] = transMatrixCol0to15[m][n] ここでm=0..15、n=0..63 (1200)

transMatrixCol0to15 =

(1201)

```

{
{ 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 }
{ 91 90 90 90 88 87 86 84 83 81 79 77 73 71 69 65 }
{ 90 90 88 85 82 78 73 67 61 54 46 38 31 22 13 4 }
{ 90 88 84 79 71 62 52 41 28 15 2 -11 -24 -37 -48 -59 }
{ 90 87 80 70 57 43 25 9 -9 -25 -43 -57 -70 -80 -87 -90 }
{ 90 84 73 59 41 20 -2 -24 -44 -62 -77 -86 -90 -90 -83 -71 }
{ 90 82 67 46 22 -4 -31 -54 -73 -85 -90 -88 -78 -61 -38 -13 }
{ 90 79 59 33 2 -28 -56 -77 -88 -90 -81 -62 -37 -7 24 52 }
{ 89 75 50 18 -18 -50 -75 -89 -89 -75 -50 -18 18 50 75 89 }
{ 88 71 41 2 -37 -69 -87 -90 -73 -44 -7 33 65 86 90 77 }
{ 88 67 31 -13 -54 -82 -90 -78 -46 -4 38 73 90 85 61 22 }
{ 87 62 20 -28 -69 -90 -84 -56 -11 37 73 90 81 48 2 -44 }
{ 87 57 9 -43 -80 -90 -70 -25 25 70 90 80 43 -9 -57 -87 }
{ 86 52 -2 -56 -87 -84 -48 7 59 88 83 44 -11 -62 -90 -81 }
{ 85 46 -13 -67 -90 -73 -22 38 82 88 54 -4 -61 -90 -78 -31 }
{ 84 41 -24 -77 -90 -56 7 65 91 69 11 -52 -88 -79 -28 37 }
{ 83 36 -36 -83 -83 -36 36 83 83 36 -36 -83 -83 -36 36 83 }
{ 83 28 -44 -88 -73 -11 59 91 62 -7 -71 -90 -48 24 81 84 }
{ 82 22 -54 -90 -61 13 78 85 31 -46 -90 -67 4 73 88 38 }
{ 81 15 -62 -90 -44 37 88 69 -7 -77 -84 -24 56 91 52 -28 }
{ 80 9 -70 -87 -25 57 90 43 -43 -90 -57 25 87 70 -9 -80 }
{ 79 2 -77 -81 -7 73 83 11 -71 -84 -15 69 86 20 -65 -87 }
{ 78 -4 -82 -73 13 85 67 -22 -88 -61 31 90 54 -38 -90 -46 }
{ 77 -11 -86 -62 33 90 44 -52 -90 -24 69 83 2 -81 -71 20 }
{ 75 -18 -89 -50 50 89 18 -75 -75 18 89 50 -50 -89 -18 75 }
{ 73 -24 -90 -37 65 81 -11 -88 -48 56 86 2 -84 -59 44 90 }
{ 73 -31 -90 -22 78 67 -38 -90 -13 82 61 -46 -88 -4 85 54 }
{ 71 -37 -90 -7 86 48 -62 -79 24 91 20 -81 -59 52 84 -11 }
{ 70 -43 -87 9 90 25 -80 -57 57 80 -25 -90 -9 87 43 -70 }
{ 69 -48 -83 24 90 2 -90 -28 81 52 -65 -71 44 84 -20 -90 }
{ 67 -54 -78 38 85 -22 -90 4 90 13 -88 -31 82 46 -73 -61 }
{ 65 -59 -71 52 77 -44 -81 37 84 -28 -87 20 90 -11 -90 2 }
{ 64 -64 -64 64 64 -64 -64 64 64 -64 -64 64 64 -64 -64 64 }
{ 62 -69 -56 73 48 -79 -41 83 33 -86 -24 88 15 -90 -7 91 }
{ 61 -73 -46 82 31 -88 -13 90 -4 -90 22 85 -38 -78 54 67 }
}

```

20

30

40

50

```

{ 59 -77 -37 87 11 -91 15 86 -41 -73 62 56 -79 -33 88 7 }
{ 57 -80 -25 90 -9 -87 43 70 -70 -43 87 9 -90 25 80 -57 }
{ 56 -83 -15 90 -28 -77 65 44 -87 -2 88 -41 -69 73 33 -90 }
{ 54 -85 -4 88 -46 -61 82 13 -90 38 67 -78 -22 90 -31 -73 }
{ 52 -87 7 83 -62 -41 90 -20 -77 71 28 -91 33 69 -79 -15 }
{ 50 -89 18 75 -75 -18 89 -50 -50 89 -18 -75 75 18 -89 50 }
{ 48 -90 28 65 -84 7 79 -73 -15 87 -59 -37 91 -41 -56 88 }
{ 46 -90 38 54 -90 31 61 -88 22 67 -85 13 73 -82 4 78 }
{ 44 -91 48 41 -90 52 37 -90 56 33 -90 59 28 -88 62 24 }
{ 43 -90 57 25 -87 70 9 -80 80 -9 -70 87 -25 -57 90 -43 }
{ 41 -90 65 11 -79 83 -20 -59 90 -48 -33 87 -71 -2 73 -86 }
{ 38 -88 73 -4 -67 90 -46 -31 85 -78 13 61 -90 54 22 -82 }
{ 37 -86 79 -20 -52 90 -69 2 65 -90 56 15 -77 87 -41 -33 }
{ 36 -83 83 -36 -36 83 -83 36 36 -83 83 -36 -36 83 -83 36 }
{ 33 -81 87 -48 -15 71 -90 62 -2 -59 90 -73 20 44 -86 83 }
{ 31 -78 90 -61 4 54 -88 82 -38 -22 73 -90 67 -13 -46 85 }
{ 28 -73 91 -71 24 33 -77 90 -69 20 37 -79 90 -65 15 41 }
{ 25 -70 90 -80 43 9 -57 87 -87 57 -9 -43 80 -90 70 -25 }
{ 24 -65 88 -86 59 -15 -33 71 -90 83 -52 7 41 -77 91 -79 }
{ 22 -61 85 -90 73 -38 -4 46 -78 90 -82 54 -13 -31 67 -88 }
{ 20 -56 81 -91 83 -59 24 15 -52 79 -90 84 -62 28 11 -48 }
{ 18 -50 75 -89 89 -75 50 -18 -18 50 -75 89 -89 75 -50 18 }
{ 15 -44 69 -84 91 -86 71 -48 20 11 -41 65 -83 90 -87 73 }
{ 13 -38 61 -78 88 -90 85 -73 54 -31 4 22 -46 67 -82 90 }
{ 11 -33 52 -69 81 -88 91 -87 79 -65 48 -28 7 15 -37 56 }
{ 9 -25 43 -57 70 -80 87 -90 90 -87 80 -70 57 -43 25 -9 }
{ 7 -20 33 -44 56 -65 73 -81 86 -90 91 -90 87 -83 77 -69 }
{ 4 -13 22 -31 38 -46 54 -61 67 -73 78 -82 85 -88 90 -90 }
{ 2 -7 11 -15 20 -24 28 -33 37 -41 44 -48 52 -56 59 -62 }
},

```

10

20

30

```

transMatrix[ m ][ n ] = transMatrixCol16to31[ m - 16 ][ n ] ここでm=16..31、
n=0..63

```

(1202)

```

transMatrixCol16to31 =

```

(1203)

```

{
{ 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 64 }
{ 62 59 56 52 48 44 41 37 33 28 24 20 15 11 7 2 }
{ -4 -13 -22 -31 -38 -46 -54 -61 -67 -73 -78 -82 -85 -88 -90 -90 }
{ -69 -77 -83 -87 -90 -91 -90 -86 -81 -73 -65 -56 -44 -33 -20 -7 }
{ -90 -87 -80 -70 -57 -43 -25 -9 9 25 43 57 70 80 87 90 }
{ -56 -37 -15 7 28 48 65 79 87 91 88 81 69 52 33 11 }
{ 13 38 61 78 88 90 85 73 54 31 4 -22 -46 -67 -82 -90 }
{ 73 87 90 83 65 41 11 -20 -48 -71 -86 -91 -84 -69 -44 -15 }
{ 89 75 50 18 -18 -50 -75 -89 -89 -75 -50 -18 18 50 75 89 }
{ 48 11 -28 -62 -84 -90 -79 -52 -15 24 59 83 91 81 56 20 }
{ -22 -61 -85 -90 -73 -38 4 46 78 90 82 54 13 -31 -67 -88 }
{ -79 -91 -77 -41 7 52 83 90 71 33 -15 -59 -86 -88 -65 -24 }
{ -87 -57 -9 43 80 90 70 25 -25 -70 -90 -80 -43 9 57 87 }
{ -41 15 65 90 79 37 -20 -69 -90 -77 -33 24 71 91 73 28 }
{ 31 78 90 61 4 -54 -88 -82 -38 22 73 90 67 13 -46 -85 }
{ 83 86 44 -20 -73 -90 -59 2 62 90 71 15 -48 -87 -81 -33 }
}

```

40

50

```

{ 83 36 -36 -83 -83 -36 36 83 83 36 -36 -83 -83 -36 36 83 }
{ 33 -41 -87 -77 -15 56 90 65 -2 -69 -90 -52 20 79 86 37 }
{ -38 -88 -73 -4 67 90 46 -31 -85 -78 -13 61 90 54 -22 -82 }
{ -86 -73 -2 71 87 33 -48 -90 -59 20 83 79 11 -65 -90 -41 }
{ -80 -9 70 87 25 -57 -90 -43 43 90 57 -25 -87 -70 9 80 }
{ -24 62 88 28 -59 -90 -33 56 90 37 -52 -90 -41 48 91 44 }
{ 46 90 38 -54 -90 -31 61 88 22 -67 -85 -13 73 82 4 -78 }
{ 88 56 -41 -91 -37 59 87 15 -73 -79 7 84 65 -28 -90 -48 }
{ 75 -18 -89 -50 50 89 18 -75 -75 18 89 50 -50 -89 -18 75 }
{ 15 -79 -69 33 91 28 -71 -77 20 90 41 -62 -83 7 87 52 }
{ -54 -85 4 88 46 -61 -82 13 90 38 -67 -78 22 90 31 -73 }
{ -90 -33 73 69 -41 -88 -2 87 44 -65 -77 28 90 15 -83 -56 }
{ -70 43 87 -9 -90 -25 80 57 -57 -80 25 90 9 -87 -43 70 }
{ -7 88 33 -79 -56 62 73 -41 -86 15 91 11 -87 -37 77 59 }
{ 61 73 -46 -82 31 88 -13 -90 -4 90 22 -85 -38 78 54 -67 }
{ 91 7 -90 -15 88 24 -86 -33 83 41 -79 -48 73 56 -69 -62 }
{ 64 -64 -64 64 64 -64 -64 64 64 -64 -64 64 64 -64 -64 64 }
{ -2 -90 11 90 -20 -87 28 84 -37 -81 44 77 -52 -71 59 65 }
{ -67 -54 78 38 -85 -22 90 4 -90 13 88 -31 -82 46 73 -61 }
{ -90 20 84 -44 -71 65 52 -81 -28 90 2 -90 24 83 -48 -69 }
{ -57 80 25 -90 9 87 -43 -70 70 43 -87 -9 90 -25 -80 57 }
{ 11 84 -52 -59 81 20 -91 24 79 -62 -48 86 7 -90 37 71 }
{ 73 31 -90 22 78 -67 -38 90 -13 -82 61 46 -88 4 85 -54 }
{ 90 -44 -59 84 2 -86 56 48 -88 11 81 -65 -37 90 -24 -73 }
{ 50 -89 18 75 -75 -18 89 -50 -50 89 -18 -75 75 18 -89 50 }
{ -20 -71 81 2 -83 69 24 -90 52 44 -90 33 62 -86 11 77 }
{ -78 -4 82 -73 -13 85 -67 -22 88 -61 -31 90 -54 -38 90 -46 }
{ -87 65 20 -86 69 15 -84 71 11 -83 73 7 -81 77 2 -79 }
{ -43 90 -57 -25 87 -70 -9 80 -80 9 70 -87 25 57 -90 43 }
{ 28 52 -91 56 24 -84 77 -7 -69 88 -37 -44 90 -62 -15 81 }
{ 82 -22 -54 90 -61 -13 78 -85 31 46 -90 67 4 -73 88 -38 }
{ 84 -81 24 48 -90 71 -7 -62 91 -59 -11 73 -88 44 28 -83 }
{ 36 -83 83 -36 -36 83 -83 36 36 -83 83 -36 -36 83 -83 36 }
{ -37 -28 79 -88 52 11 -69 91 -65 7 56 -90 77 -24 -41 84 }
{ -85 46 13 -67 90 -73 22 38 -82 88 -54 -4 61 -90 78 -31 }
{ -81 90 -62 11 44 -83 88 -59 7 48 -84 87 -56 2 52 -86 }
{ -25 70 -90 80 -43 -9 57 -87 87 -57 9 43 -80 90 -70 25 }
{ 44 2 -48 81 -90 73 -37 -11 56 -84 90 -69 28 20 -62 87 }
{ 88 -67 31 13 -54 82 -90 78 -46 4 38 -73 90 -85 61 -22 }
{ 77 -90 86 -65 33 7 -44 73 -90 87 -69 37 2 -41 71 -88 }
{ 18 -50 75 -89 89 -75 50 -18 -18 50 -75 89 -89 75 -50 18 }
{ -52 24 7 -37 62 -81 90 -88 77 -56 28 2 -33 59 -79 90 }
{ -90 82 -67 46 -22 -4 31 -54 73 -85 90 -88 78 -61 38 -13 }
{ -71 83 -90 90 -86 77 -62 44 -24 2 20 -41 59 -73 84 -90 }
{ -9 25 -43 57 -70 80 -87 90 -90 87 -80 70 -57 43 -25 9 }
{ 59 -48 37 -24 11 2 -15 28 -41 52 -62 71 -79 84 -88 90 }
{ 90 -90 88 -85 82 -78 73 -67 61 -54 46 -38 31 -22 13 -4 }
{ 65 -69 71 -73 77 -79 81 -83 84 -86 87 -88 90 -90 90 -91 }
},

```

```
transMatrix[ m ][n] = (n & 1 ? -1 : 1) * transMatrixCol16to31[ 47 - m ][ n ]
```

10

20

30

40

50

(1204)

ここでm=32..47、n=0..63

transMatrix[m][n] = (n & 1 ? -1 : 1) * transMatrixCol0to15[63 - m][n]

(1205)

ここでm=48..63、n=0..63

【0139】

変換行列の直交性を保つために、これらの変換行列はHEVCにおける変換行列よりも正確に量子化される。いくつかの例では、変換された係数の中間値を16ビットの範囲内に保つために、水平変換の後、かつ垂直変換の後、全ての係数は10ビットを有すべきである。

【0140】

MTS方式をシグナリングするために、それぞれ、イントラまたはインターについて、SPSレベルにおいて別個の有効化フラグが指定される。SPSレベルにおいてMTS方式が有効化されるとき、ビットストリーム内でCUレベルフラグがシグナリングされ、CUレベルフラグの値は、現在のCUにMTSが適用されるか否かを示すために使用される。一例では、MTSは、ルーマサンプルについてのみ適用される。一例では、以下の条件が満たされるときにCUレベルフラグがシグナリングされる。

【0141】

現在のCUの幅と現在のCUの高さの両方が32以下であり、
かつ現在のCUの符号化ブロックフラグ(coded block flag)CBFフラグの値が1に等しい。

【0142】

CUレベルフラグの値が0に等しいならば、現在のCUの水平方向および現在のCUの垂直方向においてDCT2が適用される。CUレベルフラグの値が1に等しいならば、それぞれ、水平方向および垂直方向について変換タイプを示すために追加の2つのフラグがシグナリングされる。以下の表2に表されているような、変換タイプおよびフラグ値についてのマッピング関係。変換行列精度に言及すると、8ビットの1次変換コアが使用される。一例では、変換コアは、HEVCにおいて使用される変換コアと同じである。一例では、これらの変換コアは、4ポイントのDCT-2およびDST-7、8ポイント、16ポイント、および32ポイントのDCT-2を備える。また、他の変換コアは、64ポイントのDCT-2、4ポイントのDCT-8、8ポイント、16ポイント、32ポイントのDST-7およびDCT-8を備え、8ビットの1次変換コアを使用する。

【0143】

【表2】

表2 - 変換およびシグナリングマッピングテーブル

MTS_CU_flag	MTS_Hor_flag	MTS_Ver_flag	イントラ/インター	
			水平	垂直
0			DCT2	
1	0	0	DST7	DST7
	0	1	DCT8	DST7
	1	0	DST7	DCT8
	1	1	DCT8	DCT8

【0144】

HEVCにおけるように、ブロックの残差は変換スキップモードを用いて符号化されることが可能である。シンタックス符号化の冗長性を回避するために、CUレベルフラグMTS_CU_flagの値が0に等しくないとき、変換スキップフラグはシグナリングされない。ブロック

幅とブロック高さの両方が4以下であるとき、変換スキップモードが有効化される。

【0145】

イントラ下位区分(Intra Sub-Partition(ISP))符号化モードは、VVCドラフト4.0(JVET-M1001)において採用されたツールである。ISP符号化モードが適用されるとき、ルーマイントラ予測されたブロックは垂直または水平に、表3に表されたように、ブロックサイズ寸法に依存して2つまたは4つの下位区分に分割される。図6および図7は2つの可能性の例を表す。各下位区分は少なくとも16個のサンプルを備える。

【0146】

【表3】

表3: ブロックサイズに依存した下位区分の数

ブロックサイズ	下位区分の数
4×4	分割されない
4×8および8×4	2
他の場合	4

【0147】

現在のブロックについてイントラ下位区分符号化モードが有効化されるとき(現在のブロックについてMTSが有効化されるか、または無効化されるかにかかわらず)、変換コアはMTSインデックスに従って選択されないが、変換コアは現在のブロックのイントラモードに基づいて選択される。一例では、以下の表4に表されたように、trTypeHorは、ISPモードを有する現在のブロックの水平方向についての変換コアを示し、trTypeVerは、ISPモードを有する現在のブロックの垂直方向についての変換コアを示す。

【0148】

表4によれば、trTypeVerの値が0に等しいか、またはtrTypeHorの値が0に等しいとき、現在のブロックについて変換コアDCT2が選択され、trTypeVerの値が1に等しいか、またはtrTypeHorの値が1に等しいとき、現在のブロックについて変換コアDST7が選択される。

【0149】

10

20

30

40

50

【表 4】

表4 - predModeIntraに依存したtrTypeHorおよびtrTypeVerの仕様

predModeIntra (ブロックについてのイントラ予測モード)	trTypeHor	trTypeVer
INTRA_PLANAR, INTRA_ANGULAR31, INTRA_ANGULAR32, INTRA_ANGULAR34, INTRA_ANGULAR36, INTRA_ANGULAR37	(nTbW (変換ブロックの幅) >= 4 && nTbW <= 16) ? 1 : 0	(nTbH (変換ブロックの高さ) >= 4 && nTbH <= 16) ? 1 : 0
INTRA_ANGULAR33, INTRA_ANGULAR35	0	0
INTRA_ANGULAR2, INTRA_ANGULAR4,...,INTRA_ANGULAR28, INTRA_ANGULAR30, INTRA_ANGULAR39, INTRA_ANGULAR41,...,INTRA_ANGULAR63, INTRA_ANGULAR65	(nTbW >= 4 && nTbW <= 16) ? 1 : 0	0
INTRA_ANGULAR3, INTRA_ANGULAR5,..., INTRA_ANGULAR27, INTRA_ANGULAR29, INTRA_ANGULAR38, INTRA_ANGULAR40,...,INTRA_ANGULAR64, INTRA_ANGULAR66	0	(nTbH >= 4 && nTbH <= 16) ? 1 : 0

10

20

【0150】

イントラ形状適応黙示的変換コア選択が、JVET-M0303によって開示される。

【0151】

イントラ形状適応黙示的変換コア選択方法では、変換ブロックのより短い側についてDST7を使用し、変換ブロックのより長い側についてDCT2を使用することが提案されている。正方形ブロックについては、シーケンスについてMTSが無効化されているならば、DST7を使用することが提案されている。形状適応は、空間予測モードを用いて予測されるイントラブロックについて使用される(かつ、交差成分予測されたブロックについて無効化される)。

30

【0152】

一例では、MTSが無効化されているとき、以下の式に基づいてイントラブロック変換コア選択が実行される。trTypeHorは、ISPモードを有する現在のブロックの水平方向についての変換コアを示し、trTypeVerは、ISPモードを有する現在のブロックの垂直方向についての変換コアを示し、nTbWは変換ブロック幅であり、nTbHは変換ブロック高さである。

【0153】

特定の符号化モードを用いてブロックが最初に符号化され、ブロックが符号化ブロックと名付けられ、符号化ブロックが特定の予測方法を使用して予測されてよく、元のサンプルと予測されたサンプルの間の差が残差信号と名付けられ、残差信号がさらに変換されることが注記される。変換されたブロックは変換ブロックとして定義される。通常の場合では、符号化ブロックサイズ(すなわち、符号化ブロック幅、符号化ブロック高さ)は、変換ブロックサイズ(すなわち、変換ブロック幅、変換ブロック高さ)に等しい。符号化ブロックサイズが変換ブロックサイズに等しくない、以下の例示の場合がある。

40

1.符号化ブロックサイズが128×128であり、それがサイズ64×64を有する4個の変換ブロックに黙示的に分割される。

2.符号化ブロックがISPモードによって符号化されるならば、対応する符号化ブロックが

50

ら複数の変換ブロックが分割される。

3.符号化ブロックがSBTモードによって符号化されるならば、符号化ブロックの部分のみが変換され、そのため、変換ブロックサイズは符号化ブロックサイズよりも小さい。

【0154】

式によれば、trTypeVerの値が0に等しい、またはtrTypeHorの値が0に等しいとき、現在のブロックについて変換コアDCT2が選択され、trTypeVerの値が1に等しい、またはtrTypeHorの値が1に等しいとき、現在のブロックについて変換コアDST7が選択される。

$$\text{trTypeHor} = (\text{nTbW} > 4 \ \&\& \ \text{nTbW} \leq 16 \ \&\& \ \text{nTbW} \leq \text{nTbH}) ? 1 : 0 \quad \text{EQ-1}$$

$$\text{trTypeVer} = (\text{nTbH} > 4 \ \&\& \ \text{nTbH} \leq 16 \ \&\& \ \text{nTbH} \leq \text{nTbW}) ? 1 : 0 \quad \text{EQ-2}$$

【0155】

EQ-1およびEQ-2において、nTbWは、現在の変換ユニットTUの幅を示し、nTbHは、現在のTUの高さを示し、nSbTbWは、ISP分割から結果として生じるサブTUの幅を示し、nSbTbHは、ISP分割から結果として生じるサブTUの高さを示す。

【0156】

ここで、ルーマサンプルについてサイズ制限(4および16)が使用され、ブロックは、ルーマ成分およびクロマ成分を有することができる。ビデオシーケンスがクロマサブサンプリングされていないならば、ルーマサンプルサイズ、クロマサンプルサイズは等しい。ビデオシーケンスがクロマサブサンプリングされているならば(例えば、4:2:0ビデオ)、クロマサンプルサイズは2で除算されたルーマサンプルサイズに等しい。通常、述べられたしきい値は常にルーマサンプルサイズ内にある。

【0157】

いくつかの例では、ISPの黙示的変換コア選択は、たいへん複雑なテーブルを有するイントラモードに基づく。テーブルはまた、変換コア選択とイントラモードの間の依存性を導入した。加えて、ISPの黙示的変換コア選択およびイントラ形状適応方法は、調和されていない。

【0158】

以下の提案される解決策は、ISP黙示的変換コア選択の複雑さ、イントラモードとのISP変換コア選択の依存性、ISPとイントラ形状適応黙示的変換コア選択の調和の問題に対処している。いくつかの実施形態では、これらの解決策はDST7に基づき、DCT8変換コアは、高レベルフラグの場合(sps_mts_enabled_flagの値が1に等しい)によって有効化される。

【0159】

一実施形態では、

解決策1:

解決策1によれば、形状適応黙示的変換コア選択は、ブロック幅および高さの制約を有するISPモードを使用することによって処理されたイントラブロックに依存して適用される。

【0160】

現在のイントラブロックがISPモードを使用することによって処理されるならば、現在のブロックの選択された変換コアを決定するために、式EQ-1および式EQ-2が使用される。ここでのEQ-1およびEQ-2の利用は、MTSモードが有効化されているか、または無効化されているかに依存しない。

【0161】

ISPモードを使用することによって処理されない現在のイントラブロックについて、シーケンスレベルにおいてMTSが有効化されているならば、変換コアはMTSインデックスに基づいて選択される。そうでなければ(シーケンスレベルにおいてMTSが無効化されているならば)、変換コアはEQ-1およびEQ-2に基づいて選択される。

【0162】

この提案される解決策は、ISPとイントラ形状適応黙示的変換コア選択を調和させる。さらに、提案される解決策は、ISPモード黙示的変換コア選択の複雑さを減少させ、最も悪い場合の変更なしでイントラモード依存性を除去した。

【0163】

10

20

30

40

50

一例では、DST7の計算がDCT2の計算よりも複雑であるので、従って、最も悪い場合は、全てのISPブロックが水平および垂直変換コアの両方について使用されたDST7であることである。提案される方法は、ISP黙示的変換コア選択の最も悪い場合を変更しない。

【0164】

別の実施形態では、

解決策2:

解決策2によれば、形状適応黙示的変換コア選択は、サブブロック幅および高さの制約を有するISPモードを使用することによって処理されたイントラブロックに依存して適用され、サブブロックはISP分割から結果として生じる。

【0165】

現在のイントラブロックがISPモードを使用することによって処理されるならば、現在のブロックについて変換コアを決定するために、式EQ-3および式EQ-4が使用される。ここでのEQ-3およびEQ-4の利用は、MTSモードが有効化されているか、または無効化されているかに依存しない。EQ-3およびEQ-4において、nTbWは、現在のTUの幅を示し、nTbHは、現在のTUの高さを示し、nSbTbWは、ISP分割から結果として生じるサブTUの幅を示し、nSbTbHは、ISP分割から結果として生じるサブTUの高さを示す。

$$\text{trTypeHor} = (\text{nTbW} > 4 \ \&\& \ \text{nTbW} <= 16 \ \&\& \ \text{nSbTbW} <= \text{nSbTbH}) ? 1 : 0 \quad \text{EQ-3}$$

$$\text{trTypeVer} = (\text{nTbH} > 4 \ \&\& \ \text{nTbH} <= 16 \ \&\& \ \text{nSbTbH} <= \text{nSbTbW}) ? 1 : 0 \quad \text{EQ-4}$$

【0166】

ISPモードを使用することによって処理されない現在のイントラブロックについて、シーケンスレベルにおいてMTSが有効化されているならば、変換コアはMTSインデックスに基づいて選択される。シーケンスレベルにおいてMTSが無効化されているならば、変換コアはEQ-1およびEQ-2に基づいて選択される。

【0167】

この提案される解決策は、ISPとイントラ形状適応黙示的変換コア選択を調和させる。さらに、提案される解決策は、ISPモード黙示的変換コア選択の複雑さを減少させ、最も悪い場合の変更なしでイントラモード依存性を除去した。

【0168】

一例では、DST7の計算がDCT2の計算よりも複雑であるので、従って、最も悪い場合は、全てのISPブロックが水平および垂直変換コアの両方について使用されたDST7であることである。提案される方法は、ISP黙示的変換コア選択の最も悪い場合を変更しない。

【0169】

別の実施形態では、

解決策3

解決策3によれば、サイズ制約を有するISPモードを使用することによって処理されるイントラブロックについて変換コアDST7を使用することが提案される。

【0170】

現在のイントラブロックがISPモードを使用することによって処理されるならば、現在のブロックの変換コアを決定するために、式EQ-5および式EQ-6が使用される。ここでのEQ-5およびEQ-6の利用は、MTSモードが有効化されているか、または無効化されているかに依存しない。

$$\text{trTypeHor} = (\text{nTbW} > 4 \ \&\& \ \text{nTbW} <= 16) ? 1 : 0 \quad \text{EQ-5}$$

$$\text{trTypeVer} = (\text{nTbH} > 4 \ \&\& \ \text{nTbH} <= 16) ? 1 : 0 \quad \text{EQ-6}$$

【0171】

言い換えれば、現在のブロックがISPモードを使用することによって処理される場合には、一例では、TU幅がルーマサンプルにおいて4以上であり、かつTU幅が16サンプル以下であるならば、水平方向についてDST7が使用される。そうでなければ、水平方向についてDCT2が使用される。一例では、ビデオのピクチャはルーマ成分(Y)およびクロマ成分(Cb、Cr)を有する。クロマがサブサンプリングされているならば(例えば、4:2:0ビデオ)、2個のクロマサンプルに対応する4個のルーマサンプル。

10

20

30

40

50

【0172】

一例では、TU高さがルーマサンプルにおいて4以上であり、かつTU高さがルーマサンプルにおいて16以下であるならば、垂直方向についてDST7が使用される。そうでなければ、垂直方向についてDCT2が使用される。

【0173】

ISPモードを使用することによって処理されない現在のイントラブロックについて、シーケンスレベルにおいてMTSが有効化されているならば、変換コアはMTSインデックスに基づいて選択される。シーケンスレベルにおいてMTSが無効化されているならば、変換コアはEQ-1およびEQ-2に基づいて選択される。

【0174】

この提案される解決策は、ISPモード黙示的変換コア選択の複雑さを減少させ、最も悪い場合の変更なしでイントラモード依存性を除去した。

【0175】

一例では、DST7の計算がDCT 2の計算よりも複雑であるので、従って、最も悪い場合は、全てのISPブロックが水平および垂直変換コアの両方について使用されたDST7であることである。提案される方法は、ISP黙示的変換コア選択の最も悪い場合を変更しない。

【0176】

別の実施形態では、解決策4:

解決策4によれば、ブロックのイントラ予測が平面モードを使用していないとき、ISPモードを使用することによって処理されるイントラブロックについてのTU幅および高さに依存して、形状適応黙示的変換コア選択が適用される。平面イントラ予測モードを使用してブロックが予測されるならば、水平および垂直変換コアの両方は、サイズ制約を有するDST7を使用している。

【0177】

一例では、現在のイントラブロックがISPモードを使用することによって処理されるならば、現在のイントラブロックのイントラ予測モードが平面モードでないとき、現在のブロックの変換コアを決定するために、式EQ-1および式EQ-2が使用される。EQ-1およびEQ-2の利用は、MTSモードが有効化されているか、または無効化されているかに依存しない。

【0178】

一例では、現在のイントラブロックがISPモードを使用することによって処理されるならば、現在のイントラブロックのイントラ予測モードが平面モードであるとき、現在のブロックの変換コアを決定するために、式EQ-5および式EQ-6が使用される。ここでのEQ-5およびEQ-6の利用は、MTSモードが有効化されているか、または無効化されているかに依存しない。

【0179】

ISPモードを使用することによって処理されない現在のイントラブロックについて、シーケンスレベルにおいてMTSが有効化されているならば、変換コアはMTSインデックスに基づいて選択される。シーケンスレベルにおいてMTSが無効化されているならば、変換コアはEQ-1およびEQ-2に基づいて選択される。

【0180】

この提案される解決策は、ISPとイントラ形状適応黙示的変換コア選択を調和させる。さらに、解決策は、最も悪い場合の変更なしでISPモード黙示的変換コア選択の複雑さを減少させた。

【0181】

一例では、DST7の計算がDCT 2の計算よりも複雑であるので、従って、最も悪い場合は、全てのISPブロックが水平および垂直変換コアの両方について使用されたDST7であることである。提案される方法は、ISP黙示的変換コア選択の最も悪い場合を変更しない。

【0182】

別の実施形態では、
解決策5:

10

20

30

40

50

解決策5によれば、ISP分割方向に基づいて変換コアを選択することが提案される。

【0183】

一例では、現在のイントラブロックがISPモードを使用することによって処理されるならば、現在のイントラブロックのイントラ予測モードが平面モードでないとき、現在のブロックの変換コアを決定するために、式EQ-7および式EQ-8が使用される。ここでのEQ-7およびEQ-8の利用は、MTSモードが有効化されているか、または無効化されているかに依存しない。

```
trTypeHor = ( nTbW >= 4 && nTbW <= 16 && IntraSubPartitionsSplitType =
= ISP_VER_SPLIT ) ? 1 : 0      EQ-7
```

```
trTypeVer = ( nTbH >= 4 && nTbH <= 16 && IntraSubPartitionsSplitType =
= ISP_HOR_SPLIT ) ? 1 : 0      EQ-8
```

10

【0184】

IntraSubPartitionsSplitTypeは、ISPモードのタイプを示すために使用される。いくつかの例では、ISPモードは、4個のサブTUへの分割、2個のサブTUへの分割、水平分割、垂直分割、または非分割のタイプを備えてよい。

【0185】

ISP_VER_SPLITは、垂直ISP分割モードを示すために使用され、ISP_HOR_SPLITは、水平ISP分割モードを示すために使用される。

【0186】

現在のイントラブロックが垂直ISP分割を使用しており、ブロック幅がルーマサンプルにおいて4以上であり、かつブロック高さがルーマサンプルにおいて16以下であるとき、水平変換コア(水平方向について使用される変換コア)についてDST7が選択される。そうでなければ、水平変換コアについてDCT2が選択される。

20

【0187】

一例では、現在のイントラブロックが水平ISP分割を使用しており、ブロック高さがルーマサンプルにおいて4以上であり、かつブロック高さがルーマサンプルにおいて16以下であるならば、垂直変換コア(垂直方向について使用される変換コア)についてDST7が選択される。そうでなければ、垂直変換コアについてDCT2が選択される。

【0188】

ISPモードを使用することによって処理されない現在のイントラブロックについて、シーケンスレベルにおいてMTSが有効化されているならば、変換コアはMTSインデックスに基づいて選択される。シーケンスレベルにおいてMTSが無効化されているならば、変換コアはEQ-1およびEQ-2に基づいて選択される。

30

【0189】

この提案される解決策は、ISPモード黙示的変換コア選択の複雑さを減少させ、最も悪い場合の変更なしでイントラモード依存性を除去した。

【0190】

一例では、DST7の計算がDCT2の計算よりも複雑であるので、従って、先行技術の最も悪い場合は、全てのISPブロックが水平および垂直変換コアの両方について使用されたDST7であることである。提案される方法は、ISP黙示的変換コア選択の最も悪い場合を変更しない。

40

【0191】

別の実施形態では、

解決策6

解決策6によれば、インターおよびイントラスライスの両方についてシーケンスレベルにおいてMTSが無効化されているとき、イントラブロックについてDST7コアのみを使用することが提案される。

【0192】

一例では、インターおよびイントラスライスの両方についてシーケンスレベルにおいてMTSが無効化されているならば、現在のイントラブロックの幅が4サンプル以上かつ16サンプル

50

ル以下であるとき、水平方向についてDST7が使用され、そうでなければ、EQ-9において記述されるようにDCT2が選択される。

【0193】

一例では、インターおよびイントラスライスの両方についてシーケンスレベルにおいてMTSが無効化されているならば、現在のイントラブロックの高さが4サンプル以上かつ16サンプル以下であるとき、垂直方向についてDST7が使用され、そうでなければ、EQ-10において記述されるようにDCT2が選択される。

$$\text{trTypeHor} = (\text{nTbW} > 4 \ \&\& \ \text{nTbW} \leq 16) ? 1 : 0 \quad \text{EQ-9}$$

$$\text{trTypeVer} = (\text{nTbH} > 4 \ \&\& \ \text{nTbH} \leq 16) ? 1 : 0 \quad \text{EQ-10}$$

【0194】

一実施形態では、提案される解決策6は解決策3と調和され、追加の解決策6は、追加の複雑さなしで追加の符号化利得をもたらす。

【0195】

別の実施形態では、

解決策7

解決策7によれば、インターおよびイントラスライスの両方についてシーケンスレベルにおいてMTSが無効化されているとき、イントラブロックについての変換コアを選択するために、形状および比適応方法を使用することが提案される。

【0196】

一例では、インターおよびイントラスライスの両方についてシーケンスレベルにおいてMTSが無効化されているならば、高さおよび幅の両方がルーマサンプルにおいて4以上かつルーマサンプルにおいて16以下である(例えば、 $\text{nTbW} \leq 16 \ \&\& \ \text{nTbW} > 4 \ \&\& \ \text{nTbH} \leq 16 \ \&\& \ \text{nTbH} > 4$)イントラブロックについて、

現在のブロックの(EQ-11において計算される)ブロック幅と高さの比の値が2以下であるならば、水平および垂直方向の両方についてDST7が使用され、そうでないならば(比の値が2より大きい)、より短い縁部について変換コアDST7が使用され、長い縁部について変換コアDCT2が使用される。例えば、幅16および高さ4を有するブロックならば、より短い縁部は4であり、より長い縁部は16である。この例では、高さ4(垂直方向)を有するより短い縁部について変換コアDST7が使用され、幅16(水平方向)を有するより長い縁部について変換コアDCT2が使用される。

【0197】

方法はEQ-11およびEQ-12に従って記述され、比はEQ-13に従って計算され、ここで、 $\log_2()$ は2を底とする対数スケーリングであり、 $\text{abs}()$ は絶対値演算子である。

$$\text{trTypeHor} = (\text{nTbW} > 4 \ \&\& \ \text{nTbW} \leq 16 \ \&\& \ (\text{nTbW} < \text{nTbH} \mid \mid \text{Ratio} \leq 2)) ? 1 : 0 \quad \text{EQ-11}$$

$$\text{trTypeVer} = (\text{nTbH} > 4 \ \&\& \ \text{nTbH} \leq 16 \ \&\& \ (\text{nTbH} < \text{nTbW} \mid \mid \text{Ratio} \leq 2)) ? 1 : 0 \quad \text{EQ-12}$$

$$\text{Ratio} = \log_2(\text{abs}(\text{width} - \text{height})) \quad \text{EQ-13}$$

【0198】

一実施形態では、提案される解決策6は、追加の複雑さなしで追加の符号化利得をもたらす。

【0199】

別の実施形態では、

解決策8

解決策8によれば、解決策7におけるブロック幅と高さの比が、EQ-14に基づいて計算されることが可能である。

$$\text{Ratio} = 1 \ll (\text{abs}(\log_2(\text{width}) - \log_2(\text{height}))) \quad \text{EQ-14.}$$

ここで、「 $\ll$ 」は左ビットシフト演算であり、 $\log_2()$ は2に基づく対数スケーリングであり、 $\text{abs}()$ は絶対値演算である。

【0200】

10

20

30

40

50

言い換えれば、幅が高さよりも大きいならば、比は幅／高さであり、そうでなければ(幅が高さ以下である)、比は高さ／幅である。

【0201】

解決策9

解決策9によれば、形状適応部分を解決策7または解決策8の提案された形状比適応方法と置き換えるために、解決策7または解決策8が解決策1、2、および4と組み合わせられることが可能である。

【0202】

例1.デコードデバイスまたはエンコードデバイスによって実現される符号化の方法であって、

現在のブロックの予測モードがイントラ予測であり、かつ現在のブロックが、イントラ下位区分、ISP、符号化モードを使用することによって処理されるとき、

現在のブロックの幅が第1のしきい値(一例では、第1のしきい値は4であり、第1のしきい値についての単位はルーマサンプルである)以上であるか否か、および現在のブロックの幅が第2のしきい値(一例では、第2のしきい値は16であり、第2のしきい値についての単位はルーマサンプルである)以下であるか否かを決定するステップを備え、

現在のブロックの幅が第1のしきい値以上かつ第2のしきい値以下であるとき、現在のブロックの水平変換について変換コアDST 7が使用される、方法。

【0203】

例2.現在のブロックの幅が第1のしきい値よりも小さいとき、または現在のブロックの幅が第2のしきい値よりも大きいとき、

現在のブロックの水平変換について変換コアDCT 2が使用される、例1の方法。

【0204】

例3.デコードデバイスまたはエンコードデバイスによって実現される符号化の方法であって、

現在のブロックの予測モードがイントラ予測であり、かつ現在のブロックが、イントラ下位区分、ISP、符号化モードを使用することによって処理されるとき、

現在のブロックの高さが第3のしきい値(一例では、第3のしきい値は4であり、第3のしきい値についての単位はルーマサンプルである)以上であるか否か、および現在のブロックの高さが第4のしきい値(一例では、第4のしきい値は16であり、第4のしきい値についての単位はルーマサンプルである)以下であるか否かを決定するステップを備え、

現在のブロックの高さが第3のしきい値以上かつ第4のしきい値以下であるとき、現在のブロックの垂直変換について変換コアDST 7が使用される、方法。

【0205】

例4.現在のブロックの高さが第3のしきい値よりも小さいとき、または現在のブロックの高さが第4のしきい値よりも大きいとき、

現在のブロックの垂直変換について変換コアDCT 2が使用される、例3の方法。

【0206】

例5.デコードデバイスまたはエンコードデバイスによって実現される符号化の方法であって、

現在のブロックに対して多重変換選択、MTS、方式が利用できない(例えば、インターおよびイントラスライスの両方についてシーケンスレベルにおいてMTSがオフにされている)とき、

現在のブロックの幅が第1のしきい値(一例では、第1のしきい値は4であり、第1のしきい値についての単位はルーマサンプルである)以上であるか否か、および現在のブロックの幅が第2のしきい値(一例では、第2のしきい値は16であり、第2のしきい値についての単位はルーマサンプルである)以下であるか否かを決定するステップを備え、

現在のブロックの幅が第1のしきい値以上かつ第2のしきい値以下であるとき、現在のブロックの水平変換について変換コアDST 7が使用される、方法。

【0207】

例6.現在のブロックの幅が第1のしきい値よりも小さいとき、または現在のブロックの幅が第2のしきい値よりも大きいとき、

現在のブロックの水平変換について変換コアDCT 2が使用される、例5の方法。

【0208】

例7.デコードデバイスまたはエンコードデバイスによって実現される符号化の方法であって、

現在のブロックに対して多重変換選択、MTS、方式が利用できない(例えば、インターおよびイントラスライスの両方についてシーケンスレベルにおいてMTSがオフにされている)とき、

現在のブロックの高さが第3のしきい値(一例では、第3のしきい値は4であり、第3のしきい値についての単位はルーマサンプルである)以上であるか否か、および現在のブロックの高さが第4のしきい値(一例では、第4のしきい値は16であり、第4のしきい値についての単位はルーマサンプルである)以下であるか否かを決定するステップを備え、

現在のブロックの高さが第3のしきい値以上かつ第4のしきい値以下であるとき、現在のブロックの垂直変換について変換コアDST 7が使用される、方法。

【0209】

例8.現在のブロックの高さが第3のしきい値よりも小さいとき、または現在のブロックの高さが第4のしきい値よりも大きいとき、

現在のブロックの垂直変換について変換コアDCT 2が使用される、例7の方法。

【0210】

例9.デコードデバイスまたはエンコードデバイスによって実現される符号化の方法であって、

現在のブロックに対して多重変換選択、MTS、方式が利用できず(例えば、インターおよびイントラスライスの両方についてシーケンスレベルにおいてMTSがオフにされている)、現在のブロックの幅が第1のしきい値(一例では、第1のしきい値は4であり、第1のしきい値についての単位はルーマサンプルである)以上かつ第2のしきい値(一例では、第2のしきい値は16であり、第2のしきい値についての単位はルーマサンプルである)以下であり、かつ現在のブロックの高さが第3のしきい値(一例では、第3のしきい値は4であり、第3のしきい値についての単位はルーマサンプルである)以上かつ第4のしきい値(一例では、第4のしきい値は16であり、第4のしきい値についての単位はルーマサンプルである)以下であるとき、

現在のブロックに対応するブロック幅と高さの比の値が第5のしきい値(一例では、第5のしきい値は2である)よりも大きいのか否かを決定するステップを備え、

ブロック幅と高さの比の値が第5のしきい値以下であるとき、

現在のブロックの水平および垂直変換について変換コアDST 7が使用される、方法。

【0211】

例10.ブロック幅と高さの比の値が第5のしきい値よりも大きく、かつブロックの幅がブロックの高さよりも小さいとき、

現在のブロックの水平変換について変換コアDST 7が使用され、現在のブロックの垂直変換について変換コアDCT 2が使用される、例9の方法。

【0212】

例11.ブロック幅と高さの比の値が第5のしきい値よりも大きく、かつブロックの幅がブロックの高さよりも大きいとき、

現在のブロックの水平変換について変換コアDCT 2が使用され、現在のブロックの垂直変換について変換コアDST 7が使用される、例9または10の方法。

【0213】

例12.例1から11のいずれか1つによる方法を実行するための処理回路を備えるエンコーダ(20)。

【0214】

例13.例1から11のいずれか1つによる方法を実行するための処理回路を備えるデコーダ(3

10

20

30

40

50

0)。

【0215】

例14.例1から11のいずれか1つによる方法を実行するためのプログラムコードを備えるコンピュータプログラム製品。

【0216】

例15.デコーダまたはエンコーダであって、

1つまたは複数のプロセッサと、

プロセッサに結合され、プロセッサによる実行のためのプログラミングを記憶する非一時的コンピュータ可読記憶媒体とを備え、プログラミングが、プロセッサによって実行されたとき、例1～11のいずれか1つによる方法を実行するようにデコーダを構成する、デコーダまたはエンコーダ。

10

【0217】

以下は、エンコード方法、並びに上記で述べた実施形態に表されたようなデコード方法、およびそれらを使用するシステムの適用の説明である。

【0218】

図8は、コンテンツ配信サービスを実現するためのコンテンツ供給システム3100を表すブロック図である。このコンテンツ供給システム3100は、キャプチャデバイス3102、端末デバイス3106を含み、任意選択でディスプレイ3126を含む。キャプチャデバイス3102は、通信リンク3104上で端末デバイス3106と通信する。通信リンクは、上記で説明された通信チャンネル13を含んでよい。通信リンク3104は、WIFI、Ethernet、ケーブル、無線(3G/4G/5G)、USB、またはそれらの任意の種類の組み合わせ、または同様のものを含むが、それらに限定されない。

20

【0219】

キャプチャデバイス3102は、データを生成し、上記の実施形態に表されたようなエンコード方法によってデータをエンコードし得る。代替として、キャプチャデバイス3102は、ストリーミングサーバ(図に表されていない)にデータを配信してよく、サーバは、データをエンコードし、エンコードされたデータを端末デバイス3106へ伝送する。キャプチャデバイス3102は、カメラ、スマートフォンまたはパッド、コンピュータまたはラップトップ、ビデオ会議システム、PDA、車両搭載型デバイス、またはそれらのいずれかの組み合わせ、または同様のものを含むが、それらに限定されない。例えば、キャプチャデバイス3102は、上記で説明されたようなソースデバイス12を含んでよい。データがビデオを含むとき、キャプチャデバイス3102内に含まれるビデオエンコーダ20が、実際にビデオエンコード処理を実行し得る。データがオーディオ(すなわち、音声)を含むとき、キャプチャデバイス3102内に含まれるオーディオエンコーダが、実際にオーディオエンコード処理を実行し得る。いくつかの実際的なシナリオについて、キャプチャデバイス3102は、エンコードされたビデオおよびオーディオデータを、それらを一緒に多重化することによって配信する。他の実際的なシナリオについて、例えば、ビデオ会議システムでは、エンコードされたオーディオデータおよびエンコードされたビデオデータは多重化されない。キャプチャデバイス3102は、エンコードされたオーディオデータおよびエンコードされたビデオデータを端末デバイス3106に別々に配信する。

30

40

【0220】

コンテンツ供給システム3100において、端末デバイス310は、エンコードされたデータを受信および再生する。端末デバイス3106は、上記で述べたエンコードされたデータをデコードすることが可能な、スマートフォンまたはパッド3108、コンピュータまたはラップトップ3110、ネットワークビデオレコーダ(network video recorder(NVR))／デジタルビデオレコーダ(digital video recorder(DVR))3112、TV 3114、セットトップボックス(set top box(STB))3116、ビデオ会議システム3118、ビデオ監視システム3120、パーソナルデジタルアシスタント(personal digital assistant(PDA))3122、車両搭載型デバイス3124、またはそれらのいずれかの組み合わせ、または同様のもののような、データ受信および復元能力を有するデバイスであることが可能である。例えば、端末

50

デバイス3106は、上記で説明されたような宛先デバイス14を含んでよい。エンコードされたデータがビデオを含むとき、端末デバイス内に含まれるビデオデコーダ30は、ビデオデコードを実行するように優先順位付けされる。エンコードされたデータがオーディオを含むとき、端末デバイス内に含まれるオーディオデコーダは、オーディオデコード処理を実行するように優先順位付けされる。

【0221】

そのディスプレイを有する端末デバイス、例えば、スマートフォンまたはパッド3108、コンピュータまたはラップトップ3110、ネットワークビデオレコーダ(NVR)/デジタルビデオレコーダ(DVR)3112、TV 3114、パーソナルデジタルアシスタント(PDA)3122、または車両搭載型デバイス3124に対して、端末デバイスは、そのディスプレイにデコードされたデータを供給することができる。ディスプレイが装備されていない端末デバイス、例えば、STB 3116、ビデオ会議システム3118、またはビデオ監視システム3120に対して、デコードされたデータを受信および表示するために外部ディスプレイ3126がそこに接触される。

10

【0222】

このシステム内の各デバイスがエンコードまたはデコードを実行するとき、上記で述べた実施形態に表されたように、ピクチャエンコードデバイスまたはピクチャデコードデバイスが使用されることが可能である。

【0223】

図9は、端末デバイス3106の一例の構造を表す図である。端末デバイス3106がキャプチャデバイス3102からストリームを受信した後、プロトコル進行ユニット3202がストリームの伝送プロトコルを分析する。プロトコルは、リアルタイム・ストリーミング・プロトコル(Real Time Streaming Protocol(RTSP))、ハイパーテキスト転送プロトコル(Hyper Text Transfer Protocol(HTTP))、HTTPライブ・ストリーミング・プロトコル(HTTP Live streaming protocol(HLS))、MPEG-DASH、リアルタイム・トランスポート・プロトコル(Real-time Transport protocol(RTP))、リアルタイム・メッセージング・プロトコル(Real Time Messaging Protocol(RTMP))、またはそれらの任意の種類の組み合わせ、または同様のものを含むが、それらに限定されない。

20

【0224】

プロトコル進行ユニット3202がストリームを処理した後、ストリームファイルが生成される。ファイルは逆多重化ユニット3204に出力される。逆多重化ユニット3204は、多重化されたデータをエンコードされたオーディオデータおよびエンコードされたビデオデータに分離することができる。上記で説明されたように、いくつかの実際的なシナリオについて、例えば、ビデオ会議システムでは、エンコードされたオーディオデータおよびエンコードされたビデオデータは多重化されない。この状況では、エンコードされたデータは、逆多重化ユニット3204を通すことなくビデオデコーダ3206およびオーディオデコーダ3208へ伝送される。

30

【0225】

逆多重化処理を介して、ビデオエレメンタリストリーム(elementary stream(ES))、オーディオES、および任意選択でサブタイトルが生成される。上記で述べた実施形態で説明されたようなビデオデコーダ30を含むビデオデコーダ3206は、上記で述べた実施形態に表されたようなデコード方法によってビデオESをデコードしてビデオフレームを生成し、このデータを同期ユニット3212に供給する。オーディオデコーダ3208は、オーディオESをデコードしてオーディオフレームを生成し、このデータを同期ユニット3212に供給する。代替として、ビデオフレームは、それを同期ユニット3212に供給する前にバッファ(図9に表されていない)に記憶し得る。同様に、オーディオフレームは、それを同期ユニット3212に供給する前にバッファ(図9に表されていない)に記憶し得る。

40

【0226】

同期ユニット3212は、ビデオフレームとオーディオフレームを同期させ、ビデオ/オーディオをビデオ/オーディオディスプレイ3214に供給する。例えば、同期ユニット3212

50

は、ビデオおよびオーディオ情報の提示を同期させる。情報は、符号化されたオーディオおよびビジュアルデータの提示に関するタイムスタンプ、およびデータストリーム自体の配信に関するタイムスタンプを使用してシンタックス内に符号化し得る。

【0227】

ストリーム内にサブタイトルが含まれるならば、サブタイトルデコーダ3210が、サブタイトルをデコードし、それをビデオフレームおよびオーディオフレームと同期させ、ビデオ／オーディオ／サブタイトルをビデオ／オーディオ／サブタイトルディスプレイ3216に供給する。

【0228】

本発明は上記で述べたシステムに限定されず、上記で述べた実施形態におけるピクチャエンコードデバイスまたはピクチャデコードデバイスのいずれかは、他のシステム、例えば、自動車システムの中に組み込まれることが可能である。

10

【0229】

数学演算子

この出願において使用される数学演算子は、Cプログラミング言語において使用されるものと類似である。しかし、整数除算および算術シフト演算の結果がより精密に定義され、べき乗および実数値除算のような追加の演算が定義される。番号付けおよび計数の規約は、一般に、0から始まり、例えば、「第1」が0番目と等価であり、「第2」が1番目と等価である、などである。

【0230】

20

算術演算子

以下の算術演算子は以下のように定義される。

+ 加算。

- 減算(2つの引数の演算子として)または符号反転(単項前置演算子として)。

* 乗算、行列乗算を含む。

x^y べき乗。 x の y 乗を指定する。他の文脈では、そのような表記法は、べき乗としての解釈のために意図されない上付き文字にするために使用される。

/ 結果の0への切り捨てを伴う整数除算。例えば、 $7/4$ および $-7/-4$ は1に切り捨てられ、 $-7/4$ および $7/-4$ は-1に切り捨てられる。

÷ 切り捨てまたは丸めが意図されない、数式における除算を表記するために使用される。

30

【0231】

【数2】

$$\frac{x}{y}$$

【0232】

切り捨てまたは丸めが意図されない、数式における除算を表記するために使用される。

【0233】

40

【数3】

$$\sum_{i=x}^y f(i)$$

【0234】

i が x から y まで、かつ y を含む全ての整数値をとる、 $f(i)$ の合算。

$x \% y$ 法。 $x \geq 0$ かつ $y > 0$ となる整数 x および y のみについて定義される、 x を y で除算し

50

た剰余。

【0235】

論理演算子

以下の論理演算子は以下のように定義される。

$x \ \&\& \ y$ x と y のブール論理の「論理積」。

$x \ || \ y$ x と y のブール論理の「論理和」。

$!$ ブール論理の「否定」。

$x \ ? \ y \ : \ z$ x がTRUE、すなわち0に等しくないならば、 y の値に評価し、そうでなければ、 z の値に評価する。

【0236】

関係演算子

以下の関係演算子は以下のように定義される。

$>$ よりも大きい。

$>=$ 以上。

$<$ よりも小さい。

$<=$ 以下。

$==$ 等しい。

$!=$ 等しくない。

【0237】

値「na」(適用可能でない)が割り当てられているシンタックス要素または変数に関係演算子が適用されるとき、値「na」は、そのシンタックス要素または変数について特別な値として扱われる。値「na」は、いかなる他の値にも等しくないと見なされる。

【0238】

ビット単位演算子

以下のビット単位演算子は以下のように定義される。

$\&$ ビット単位の「論理積」。整数の引数に対して演算するとき、整数値の2の補数表現に対して演算する。別の引数よりも少ないビットを含む2進数の引数に対して演算するとき、より短い引数は、0に等しいより上位のビットを追加することによって拡張される。

$|$ ビット単位の「論理和」。整数の引数に対して演算するとき、整数値の2の補数表現に対して演算する。別の引数よりも少ないビットを含む2進数の引数に対して演算するとき、より短い引数は、0に等しいより上位のビットを追加することによって拡張される。

$\wedge$ ビット単位の「排他的論理和」。整数の引数に対して演算するとき、整数値の2の補数表現に対して演算する。別の引数よりも少ないビットを含む2進数の引数に対して演算するとき、より短い引数は、0に等しいより上位のビットを追加することによって拡張される。

$x \gg y$ x の2の補数整数表現の、2進数の y 桁だけの算術右シフト。この関数は、 y の非負の整数値に対してのみ定義される。右シフトの結果として最上位ビット(most significant bit(MSB))にシフトされるビットは、そのシフト演算の前の x のMSBに等しい値を有する。

$x \ll y$ x の2の補数整数表現の、2進数の y 桁だけの算術左シフト。この関数は、 y の非負の整数値に対してのみ定義される。左シフトの結果として最下位ビット(least significant bit(LSB))にシフトされるビットは、0に等しい値を有する。

【0239】

割当て演算子

以下の算術演算子は以下のように定義される。

$=$ 割当て演算子。

$++$ インクリメント、すなわち、 $x++$ は $x = x + 1$ と等価であり、アレイインデックスにおいて使用されるとき、インクリメント演算の前の変数の値に評価する。

$--$ デクリメント、すなわち、 $x--$ は $x = x - 1$ と等価であり、アレイインデックスにおいて使用されるとき、デクリメント演算の前の変数の値に評価する。

10

20

30

40

50

+= 指定された量だけのインクリメント、すなわち、 $x += 3$ は $x = x + 3$ と等価であり $x += (-3)$ は $x = x + (-3)$ と等価である。

-= 指定された量だけのデクリメント、すなわち、 $x -= 3$ は $x = x - 3$ と等価であり、 $x -= (-3)$ は $x = x - (-3)$ と等価である。

【0240】

範囲表記法

値の範囲を指定するために以下の表記法が使用される。

$x=y..z$ x は、 y から始まり z までの全てを含む整数値をとり、 x 、 y 、および z は整数であり、 z は y よりも大きい。

【0241】

数学関数

以下の数学関数が定義される。

【0242】

【数4】

$$\text{Abs}(x) = \begin{cases} x & ; x \geq 0 \\ -x & ; x < 0 \end{cases}$$

【0243】

Asin(x) $-1.0 \sim 1.0$ の全てを含む範囲内にある引数 x に対して演算し、ラジアン単位での $-\pi \div 2$ から $\pi \div 2$ の全てを含む範囲内の出力値を有する、三角法の逆正弦関数。

Atan(x) 引数 x に対して演算し、ラジアン単位での $-\pi \div 2$ から $\pi \div 2$ の全てを含む範囲内の出力値を有する、三角法の逆正接関数。

【0244】

【数5】

$$\text{Atan2}(y, x) = \begin{cases} \text{Atan}\left(\frac{y}{x}\right) & ; x > 0 \\ \text{Atan}\left(\frac{y}{x}\right) + \pi & ; x < 0 \ \&\& \ y \geq 0 \\ \text{Atan}\left(\frac{y}{x}\right) - \pi & ; x < 0 \ \&\& \ y < 0 \\ +\frac{\pi}{2} & ; x = 0 \ \&\& \ y \geq 0 \\ -\frac{\pi}{2} & ; \text{その他} \end{cases}$$

【0245】

Ceil(x) x 以上の最小の整数。

Clip1Y(x) = Clip3(0, (1 << BitDepth_Y) - 1, x)

Clip1C(x) = Clip3(0, (1 << BitDepth_C) - 1, x)

【0246】

【数6】

$$\text{Clip3}(x, y, z) = \begin{cases} x & ; z < x \\ y & ; z > y \\ z & ; \text{その他} \end{cases}$$

【0247】

Cos(x) ラジアン単位での引数 x に対して演算する、三角法の余弦関数。

Floor(x) x 以下の最大の整数。

10

20

30

40

50

【 0 2 4 8 】

【数 7】

$$\text{GetCurrMsb}(a, b, c, d) = \begin{cases} c + d & ; \quad b - a \geq d/2 \\ c - d & ; \quad a - b > d/2 \\ c & ; \quad \text{その他} \end{cases}$$

【 0 2 4 9 】

$\text{Ln}(x)$ x の自然対数(e を底とする対数、ここで、 e は自然対数の底定数2.718 281 828...である)。

10

$\text{Log2}(x)$ 2を底とする x の対数。

$\text{Log10}(x)$ 10を底とする x の対数。

【 0 2 5 0 】

【数 8】

$$\text{Min}(x, y) = \begin{cases} x & ; \quad x \leq y \\ y & ; \quad x > y \end{cases}$$

$$\text{Max}(x, y) = \begin{cases} x & ; \quad x \geq y \\ y & ; \quad x < y \end{cases}$$

20

【 0 2 5 1 】

$\text{Round}(x) = \text{Sign}(x) * \text{Floor}(\text{Abs}(x) + 0.5)$

【 0 2 5 2 】

【数 9】

$$\text{Sign}(x) = \begin{cases} 1 & ; \quad x > 0 \\ 0 & ; \quad x = 0 \\ -1 & ; \quad x < 0 \end{cases}$$

30

【 0 2 5 3 】

$\text{Sin}(x)$ ラジアン単位での引数 x に対して演算する、三角法の正弦関数。

【 0 2 5 4 】

【数 10】

$$\text{Sqrt}(x) = \sqrt{x}$$

40

【 0 2 5 5 】

$\text{Swap}(x, y) = (y, x)$

$\text{Tan}(x)$ ラジアン単位での引数 x に対して演算する、三角法の正接関数。

【 0 2 5 6 】

演算優先順位の順序

式における優先順位の順序が括弧の使用によって明示的には示されないとき、以下の規則が適用される。

- より高い優先順位の演算は、より低い優先順位の任意の演算の前に評価される。
- 同じ優先順位の演算は、左から右へ順次に評価される。

50

【 0 2 5 7 】

以下の表は、最高から最低までの演算の優先順位を指定し、表内のより高い位置がより高い優先順位を示す。

【 0 2 5 8 】

Cプログラミング言語においても使用されるそれらの演算子について、この明細書において使用される優先順位の順序は、Cプログラミング言語において使用されるのと同じである。

【 0 2 5 9 】

【表 5】

10

表：(表の最上部における)最高から(表の最下部における)最低までの演算優先順位

演算(オペランド x、y、および z を有する)
"x++", "x--"
"!x", "-x" (単項前置演算子として)
x^y
"x * y", "x / y", "x ÷ y", " $\frac{x}{y}$ ", "x % y"
"x + y", "x - y" (2 つの引数の演算子として), " $\sum_{i=x}^y f(i)$ "
"x << y", "x >> y"
"x < y", "x <= y", "x > y", "x >= y"
"x == y", "x != y"
"x & y"
"x y"
"x && y"
"x y"
"x ? y : z"
"x.y"
"x = y", "x += y", "x -= y"

20

30

40

【 0 2 6 0 】

論理演算の本文記述

本文の中で、以下の形式で数学的に記述されることになるような論理演算のステートメント、すなわち、

if(条件0)

ステートメント0

50

```
else if(条件1)
ステートメント1
```

```
...
```

```
else /* 残りの条件における説明的な注釈 */
ステートメントn
```

は、以下の形態で説明され得る。

...以下のように／...以下が適用される

- 条件0ならば、ステートメント0
- そうでなく、条件1ならば、ステートメント1
- ...
- そうでなければ(残りの条件における説明的な注釈)、ステートメントn

【0261】

本文の中の各々の「...ならば、そうでなく...ならば、そうでなければ...」のステートメントは、その直後に「...ならば」が来る、「...以下のように」または「...以下が適用される」を用いて導入される。「...ならば、そうでなく...ならば、そうでなければ...」の最後の条件は、常に「そうでなければ...」である。交互に配置された「...ならば、そうでなく...ならば、そうでなければ...」のステートメントは、「...以下のように」または「...以下が適用される」を、終わりの「そうでなければ...」に整合させることによって識別されることが可能である。

【0262】

本文の中で、以下の形式で数学的に記述されることになるような論理演算のステートメント、すなわち、

```
if(条件0a && 条件0b)
ステートメント0
else if(条件1a || 条件1b)
ステートメント1
```

```
...
```

```
else
ステートメントn
```

は、以下の形態で説明され得る。

...以下のように／...以下が適用される

- 以下の条件の全てが真であるならば、ステートメント0:
 - 条件0a
 - 条件0b
- そうでなく、以下の条件のうちの1つまたは複数が真であるならば、ステートメント1:
 - 条件1a
 - 条件1b
- ...
- そうでなければ、ステートメントn

【0263】

本文の中で、以下の形式で数学的に記述されることになるような論理演算のステートメント、すなわち、

```
if(条件0)
ステートメント0
if(条件1)
ステートメント1
```

は、以下の形態で説明され得る。

条件0のとき、ステートメント0

条件1のとき、ステートメント1

【0264】

発明の実施形態はビデオ符号化に基づいて主に説明されているが、符号化システム10、エンコーダ20、およびデコーダ30(およびそれに対応してシステム10)の実施形態、およびここで説明される他の実施形態がまた、静止ピクチャ処理または符号化、すなわち、ビデオ符号化におけるような任意の先行するまたは連続したピクチャから独立した、個々のピクチャの処理または符号化のために構成され得ることが注記されるべきである。一般に、ピクチャ処理符号化が単一のピクチャ17に限定される場合には、インター予測ユニット244(エンコーダ)および344(デコーダ)のみが利用可能でなくてよい。ビデオエンコーダ20およびビデオデコーダ30の全ての他の機能(ツールまたは技術とも呼ばれる)は、静止ピクチャ処理、例えば、残差計算204/304、変換206、量子化208、逆量子化210/310、(逆)変換212/312、区分262/362、イントラ予測254/354、および／またはループフィルタ処理220、320、およびエントロピー符号化270およびエントロピーデコード304のために等しく使用され得る。

【0265】

例えば、エンコーダ20およびデコーダ30の実施形態、および、例えば、エンコーダ20およびデコーダ30を参照してここで説明される機能は、ハードウェア、ソフトウェア、ファームウェア、またはそれらの任意の組み合わせで実現され得る。ソフトウェアで実現されるならば、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において記憶され、または通信媒体上で伝送され、ハードウェアベースの処理ユニットによって実行されてよい。コンピュータ可読媒体は、データ記憶媒体のような有形媒体に対応するコンピュータ可読記憶媒体、または、例えば、通信プロトコルに従って、1つの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容易にする任意の媒体を含む通信媒体を含んでよい。この形態において、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、この開示において説明される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または1つまたは複数のプロセッサによってアクセスされることが可能である任意の利用可能な媒体であってよい。コンピュータプログラム製品がコンピュータ可読媒体を含んでよい。

【0266】

例として、かつ限定せず、そのようなコンピュータ可読記憶媒体は、RAM、ROM、EEPROM、CD-ROM、または他の光ディスク記憶、磁気ディスク記憶、または他の磁気記憶デバイス、フラッシュメモリ、または命令またはデータ構造の形式で所望のプログラムコードを記憶するために使用されることが可能であり、コンピュータによってアクセスされることが可能である任意の他の媒体を備えることができる。また、任意の接続が、コンピュータ可読媒体と適切に呼ばれる。例えば、命令が、同軸ケーブル、光ファイバケーブル、ツイストペア、デジタル加入者回線(digital subscriber line(DSL))、または赤外線、無線、およびマイクロ波のような無線技術を使用して、ウェブサイト、サーバ、または他のリモートソースから伝送されるならば、同軸ケーブル、光ファイバケーブル、ツイストペア、DSL、または赤外線、無線、およびマイクロ波のような無線技術は、媒体の定義の中に含まれる。しかし、コンピュータ可読記憶媒体およびデータ記憶媒体が、接続、搬送波、信号、または他の一時的媒体を含まないが、代わりに非一時的有形記憶媒体を対象とすることが理解されるべきである。ここで使用されるようなディスク(disk)およびディスク(disc)は、コンパクトディスク(disc)(CD)、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)(DVD)、フロッピーディスク(disk)、およびBlu-ray(登録商標)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、一方、ディスク(disc)は、レーザーを用いてデータを光学的に再生する。上記のものの組み合わせも、コンピュータ可読媒体の範囲内に含まれるべきである。

【0267】

命令は、1つまたは複数のデジタル信号プロセッサ(DSP)、汎用マイクロプロセッサ、特定用途向け集積回路(ASIC)、フィールドプログラマブル論理アレイ(FPGA)、または他の等価な集積論理回路または個別論理回路のような1つまたは複数のプロセッサによって実行

10

20

30

40

50

され得る。従って、ここで使用されるような用語「プロセッサ」は、上記の構造またはここで説明される技法の実装のために適した任意の他の構造のいずれかを指し得る。加えて、いくつかの態様では、ここで説明される機能は、エンコードおよびデコードするために構成された専用のハードウェアおよび／またはソフトウェアモジュール内で提供され、または組み合わせられたコーデック内に組み込まれ得る。また、技法は、1つまたは複数の回路または論理要素内で十分に実現されることが可能である。

【0268】

この開示の技法は、無線ハンドセット、集積回路(IC)、またはICのセット(例えば、チップセット)を含む、広く様々なデバイスまたは装置内で実現され得る。様々な構成要素、モジュール、またはユニットは、開示され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この開示において説明されているが、異なるハードウェアユニットによる実現を必ずしも要求しない。むしろ、上記で説明されたように、様々なユニットが、コーデックハードウェアユニット内で組み合わせられ、または上記で説明されたような1つまたは複数のプロセッサを含む、相互動作可能なハードウェアユニットの集合によって、適したソフトウェアおよび／またはファームウェアとともに提供されてよい。

【符号の説明】

【0269】

- 10 ビデオ符号化システム
- 12 ソースデバイス
- 13 通信チャネル
- 14 宛先デバイス
- 16 ピクチャソース
- 17 ピクチャ、ピクチャデータ、未処理ピクチャ、未処理ピクチャデータ
- 18 プリプロセッサ、前処理ユニット
- 19 前処理されたピクチャ、前処理されたピクチャデータ
- 20 ビデオエンコーダ
- 21 エンコードされたピクチャデータ
- 22 通信インターフェース、通信ユニット
- 28 通信インターフェース、通信ユニット
- 30 ビデオデコーダ
- 31 デコードされたピクチャ、デコードされたピクチャデータ
- 32 ポストプロセッサ、後処理ユニット
- 33 後処理されたピクチャ、後処理されたピクチャデータ
- 34 ディスプレイデバイス
- 46 処理回路
- 201 入力、入力インターフェース
- 203 ピクチャブロック
- 204 残差計算ユニット
- 205 残差ブロック、残差
- 206 変換処理ユニット
- 207 変換係数
- 208 量子化ユニット
- 209 量子化された係数、量子化された変換係数、量子化された残差係数
- 210 逆量子化ユニット
- 211 逆量子化された係数、逆量子化された残差係数
- 212 逆変換処理ユニット
- 213 再構成された残差ブロック、対応する逆量子化された係数、変換ブロック
- 214 再構成ユニット
- 215 再構成されたブロック
- 220 ループフィルタユニット

10

20

30

40

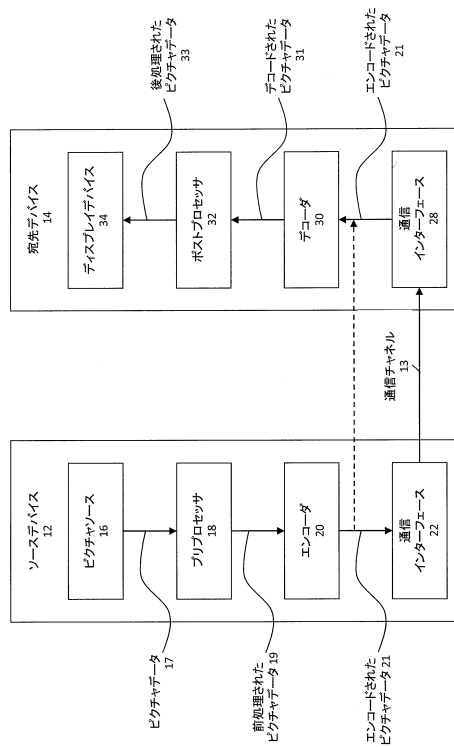
50

221	フィルタ処理されたブロック、フィルタ処理された再構成されたブロック	
230	デコードされたピクチャバッファ	
231	デコードされたピクチャ	
244	インター予測ユニット	
254	イントラ予測ユニット	
260	モード選択ユニット	
262	区分ユニット	
265	予測ブロック、予測子	
266	シンタックス要素	
270	エントロピーエンコードユニット	10
272	出力、出力インターフェース	
304	エントロピーデコードユニット	
309	量子化された係数	
310	逆量子化ユニット	
311	変換係数、逆量子化された係数	
312	逆変換処理ユニット	
313	再構成された残差ブロック、変換ブロック	
314	再構成ユニット	
315	再構成されたブロック	
320	ループフィルタユニット	20
321	フィルタ処理されたブロック	
330	デコードされたピクチャバッファ	
331	デコードされたピクチャ	
344	インター予測ユニット	
354	イントラ予測ユニット	
360	モード適用ユニット	
365	予測ブロック	
400	ビデオ符号化デバイス	
410	入口ポート、入力ポート	
420	受信機ユニット	30
430	プロセッサ、論理ユニット、中央処理ユニット	
440	送信機ユニット	
450	出口ポート、出力ポート	
460	メモリ	
470	符号化モジュール	
500	装置	
502	プロセッサ	
504	メモリ	
506	コードおよびデータ	
508	オペレーティングシステム	40
510	アプリケーションプログラム	
512	バス	
514	2次記憶装置	
518	ディスプレイ	
3100	コンテンツ供給システム	
3102	キャプチャデバイス	
3104	通信リンク	
3106	端末デバイス	
3108	スマートフォン／パッド	
3110	コンピュータ／ラップトップ	50

- 3112 ネットワークビデオレコーダ／デジタルビデオレコーダ
- 3114 TV
- 3116 セットトップボックス
- 3118 ビデオ会議システム
- 3120 ビデオ監視システム
- 3122 携帯情報端末
- 3124 車両搭載型デバイス
- 3126 ディスプレイ
- 3202 プロトコル進行ユニット
- 3204 逆多重化ユニット
- 3206 ビデオデコーダ
- 3208 オーディオデコーダ
- 3210 サブタイトルデコーダ
- 3212 同期ユニット
- 3214 ビデオ／オーディオディスプレイ
- 3216 ビデオ／オーディオ／サブタイトルディスプレイ

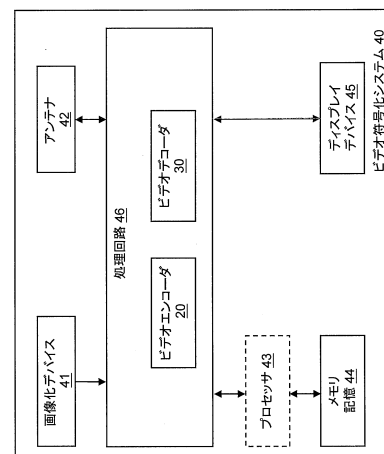
【図面】

【図 1 A】



10

【図 1 B】



10

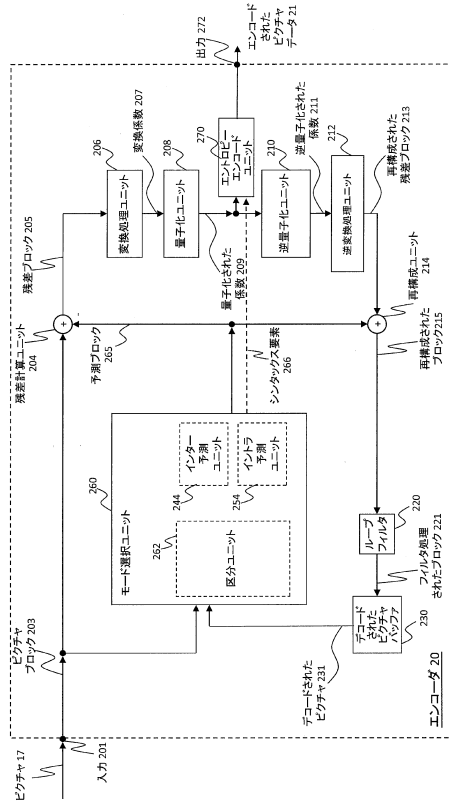
20

30

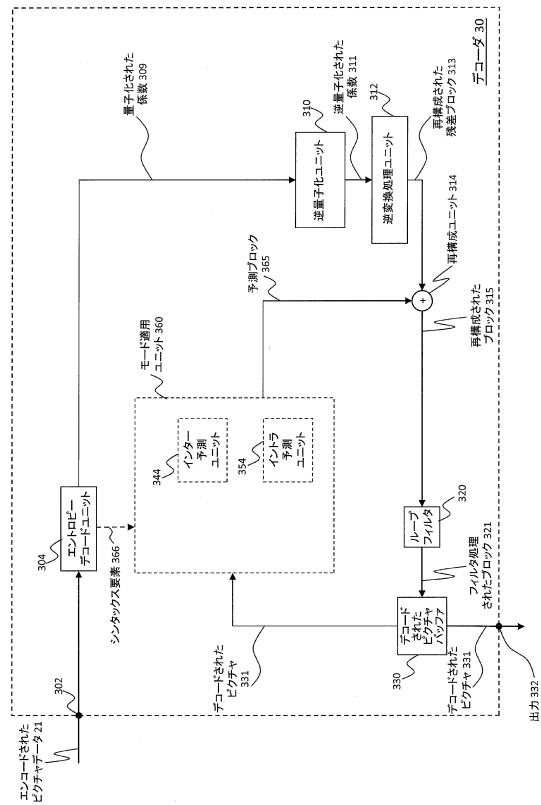
40

50

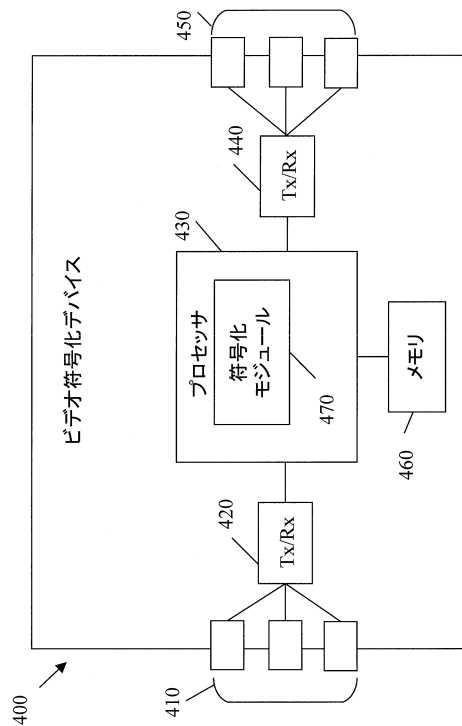
【図 2】



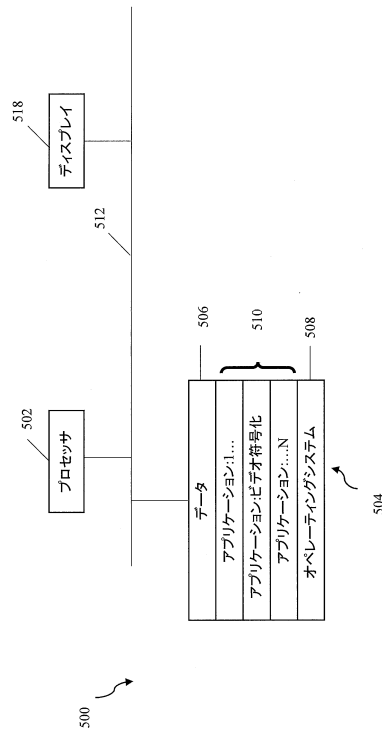
【図 3】



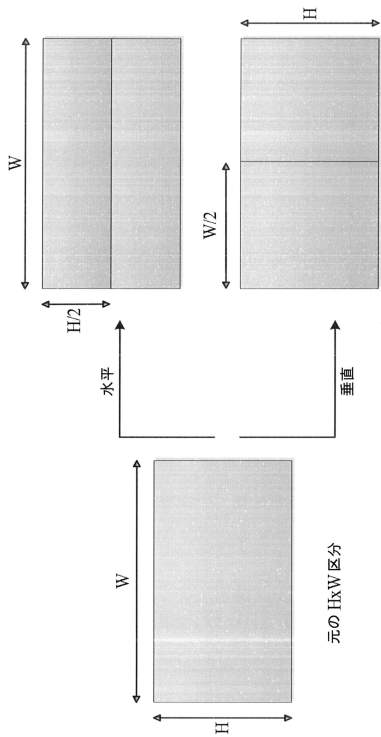
【図 4】



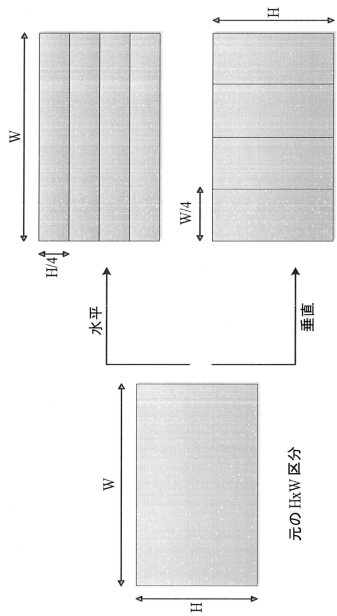
【図 5】



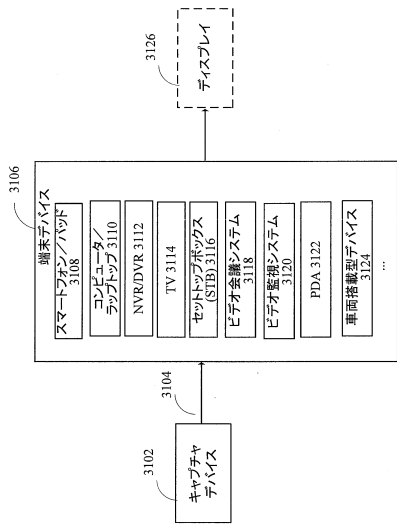
【図 6】



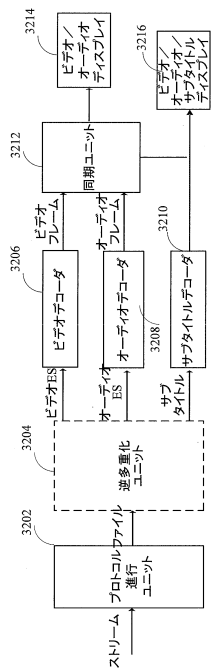
【図 7】



【図 8】



【図 9】



10

20

30

40

50

フロントページの続き

(33)優先権主張国・地域又は機関

米国(US)

(72)発明者 セミフ・エセンリク

ドイツ・８０９９２・ミュンヘン・リースシュトラッセ・２５

(72)発明者 ビャオ・ワン

ドイツ・８０９９２・ミュンヘン・リースシュトラッセ・２５

(72)発明者 アナンド・メヘル・コトラ

ドイツ・８０９９２・ミュンヘン・リースシュトラッセ・２５

(72)発明者 ジエンレ・チェン

アメリカ合衆国・カリフォルニア・９５０５０・サンタ・クララ・セントラル・エクスプレスウェイ・２３３０

審査官 岩井 健二

(56)参考文献 米国特許出願公開第２０１８／０３３２２８９（ＵＳ，Ａ１）

Jani Lainema, CE6: Shape adaptive transform selection (Test 3.1), Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JVET-M0303, 13th Meeting: Marrakech, 2019年01月, pp.1-5

Benjamin Bross, Jianle Chen, and Shan Liu, Versatile Video Coding (Draft 4), Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JVET-M1001 (version 5), 13th Meeting: Marrakech, MA, 2019年02月, pp.43-46,88-92,224-231

Han Gao, et al., Non-CE6: Unification of Implicit Transform Core Selection, 14th Meeting: Geneva, CH, JVET-N0172-v3, 14th Meeting: Geneva, CH, 2019年03月, pp.1-10

Sung-Chang Lim, et al., Non-CE6: Simplification on implicit transform selection in ISP mode, Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JVET-N0375_r1, 14th Meeting: Geneva, CH, 2019年03月, pp.1-7

(58)調査した分野 (Int.Cl., DB名)

H04N 19/00 - 19/98