

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7520809号
(P7520809)

(45)発行日 令和6年7月23日(2024.7.23)

(24)登録日 令和6年7月12日(2024.7.12)

(51)国際特許分類		F I	
H O 4 N	19/13 (2014.01)	H O 4 N	19/13
H O 4 N	19/157 (2014.01)	H O 4 N	19/157
H O 4 N	19/91 (2014.01)	H O 4 N	19/91
H O 4 N	19/176 (2014.01)	H O 4 N	19/176
H O 4 N	19/12 (2014.01)	H O 4 N	19/12
請求項の数 16 (全36頁) 最終頁に続く			
(21)出願番号	特願2021-506651(P2021-506651)	(73)特許権者	518338149
(86)(22)出願日	令和1年9月13日(2019.9.13)		インターデジタル ヴイシー ホールディ
(65)公表番号	特表2022-500891(P2022-500891 A)		ングス, インコーポレイテッド
(43)公表日	令和4年1月4日(2022.1.4)		アメリカ合衆国, デラウェア州 1 9 8
(86)国際出願番号	PCT/US2019/051089		0 9, ウィルミントン, ベルビュー パ
(87)国際公開番号	WO2020/060867	(74)代理人	ークウェイ 2 0 0, スイート 3 0 0
(87)国際公開日	令和2年3月26日(2020.3.26)		100079108
審査請求日	令和4年9月9日(2022.9.9)	(74)代理人	弁理士 稲葉 良幸
(31)優先権主張番号	18290103.3	(74)代理人	100109346
(32)優先日	平成30年9月21日(2018.9.21)		弁理士 大貫 敏史
(33)優先権主張国・地域又は機関	欧州特許庁(EP)	(74)代理人	100117189
(31)優先権主張番号	19305641.3		弁理士 江口 昭彦
(32)優先日	令和1年5月20日(2019.5.20)	(74)代理人	100134120
最終頁に続く		(74)代理人	弁理士 内藤 和彦
		(74)代理人	100108213
		最終頁に続く	

(54)【発明の名称】 スカラー量子化従属性のためのスカラー量子化器決定スキーム

(57)【特許請求の範囲】
【請求項1】

方法であって、
ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットをエントロピー復号することであって、前記第1の変換係数が、復号順序で前記ピクチャの前記ブロック内の第2の変換係数に先行する、ことと、
前記第2の変換係数に関連する第2のパラメータセットをエントロピー復号することであって、前記第1および第2のパラメータセットが、バイナリ算術コード化エンジンの通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセット内の少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の復号に依存し、前記第1および第2のパラメータセットが前記ブロックの第1のスキャンパスでエントロピー復号され、前記第1のスキャンパスが、前記ブロックのエントロピー復号変換係数についての1つ以上の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、少なくともSIGフラグ、第1のフラグおよび第2のフラグを含み、前記SIGフラグが、対応する変換係数がゼロであるか否かを示し、前記第1のフラグが、前記対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記第2のフラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示し、前記第2の変換係数を逆量子化する逆量子化器が、前記第1の変換係数に関連付けられる前記第1のパラメータセットの前記SIGフラグ、前記第1のフラグ、及び前記第2のフラグの和又はXOR関数に基づいて、2つ以上の逆量子化

器から選択される、ことと、

前記復号された変換係数に応答して前記ブロックを再構築することと、
を備える、方法。

【請求項 2】

前記第 2 の変換係数を逆量子化する逆量子化器が、前記第 1 の変換係数に基づいて、2 つ以上の量子化器から選択される、請求項 1 に記載の方法。

【請求項 3】

前記逆量子化器が、前記第 1 の変換係数の前記第 1 のパラメータセットに基づいて選択される、請求項 2 に記載の方法。

【請求項 4】

前記逆量子化器が、前記第 1 のフラグ、前記第 2 のフラグ、または前記第 1 のパラメータセットの別のフラグに基づいて選択され、前記別のフラグが、前記対応する変換係数の絶対値が x よりも大きいかどうかを示す、請求項 2 に記載の方法。

【請求項 5】

前記 S I G、第 1、第 2 のフラグ、または別のフラグのコンテキストモデリングが、前記量子化器または前記量子化器の選択において使用される状態に基づく、請求項 4 に記載の方法。

【請求項 6】

1 つ以上のプロセッサを備える装置であって、前記 1 つ以上のプロセッサが、
ピクチャのブロック内の第 1 の変換係数に関連する第 1 のパラメータセットをエントロピー復号することであって、前記第 1 の変換係数が、復号順序で前記ピクチャの前記ブロック内の第 2 の変換係数に先行する、ことと、

前記第 2 の変換係数に関連する第 2 のパラメータセットをエントロピー復号することであって、前記第 1 および第 2 のパラメータセットが、バイナリ算術コード化エンジンの通常モードでエントロピー符号化され、前記第 2 の変換係数の前記第 2 のパラメータセット内の少なくともパラメータのコンテキストモデリングが、前記第 1 の変換係数の復号に依存し、前記第 1 および第 2 のパラメータセットが前記ブロックの第 1 のスキャンパスでエントロピー復号され、前記第 1 のスキャンパスが、前記ブロックのエントロピー復号変換係数についての 1 つ以上の他のスキャンパスの前に実行され、前記第 1 および第 2 の変換係数についての前記第 1 および第 2 のパラメータセットの各セットが、少なくとも S I G フラグ、第 1 のフラグおよび第 2 のフラグを含み、前記 S I G フラグが、対応する変換係数がゼロであるか否かを示し、前記第 1 のフラグが、前記対応する変換係数の絶対値が 1 よりも大きいかどうかを示し、前記第 2 のフラグが、前記対応する変換係数の絶対値が 2 よりも大きいかどうかを示し、前記第 2 の変換係数を逆量子化する逆量子化器が、前記第 1 の変換係数に関連付けられる前記第 1 のパラメータセットの前記 S I G フラグ、前記第 1 のフラグ、及び前記第 2 のフラグの和又は X O R 関数に基づいて、2 つ以上の逆量子化器から選択される、ことと、

前記復号された変換係数に応答して前記ブロックを再構築することと、
を行うように構成されている、装置。

【請求項 7】

前記第 2 の変換係数を逆量子化する逆量子化器が、前記第 1 の変換係数に基づいて、2 つ以上の量子化器から選択される、請求項 6 に記載の装置。

【請求項 8】

前記逆量子化器が、前記第 1 の変換係数についての前記第 1 のパラメータセットに基づいて選択される、請求項 7 に記載の装置。

【請求項 9】

前記逆量子化器が、前記第 1 のフラグ、前記第 2 のフラグ、または前記第 1 のパラメータセットの別のフラグに基づいて選択され、前記別のフラグが、前記対応する変換係数の絶対値が x よりも大きいかどうかを示す、請求項 7 に記載の装置。

【請求項 10】

10

20

30

40

50

前記SIG、第1、第2のフラグまたは別のフラグのコンテキストモデリングが、前記量子化器または前記量子化器の選択において使用されるステートに基づく、請求項7に記載の装置。

【請求項11】

方法であって、

ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスすることであって、前記第1の変換係数が、符号化順序で前記ピクチャの前記ブロック内の第2の変換係数に先行する、ことと、

前記第2の変換係数に関連する第2のパラメータセットにアクセスすることであって、前記第1および第2のパラメータセットが、バイナリ算術コード化エンジンの通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の符号化に依存する、ことと、

前記ブロックの第1のスキャンパスにおける前記第1および第3のパラメータセットをエントロピー符号化することであって、前記第1のスキャンパスが、前記ブロックのエントロピー符号化変換係数の1つ以上の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、少なくともSIGフラグ、第1のフラグおよび第2のフラグを含み、前記SIGフラグが、対応する変換係数がゼロであるか否かを示し、前記第1のフラグが、前記対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記第2のフラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示し、前記第2の変換係数を量子化する量子化器が、前記第1のパラメータセットの前記SIGフラグ、前記第1のフラグ、及び前記第2のフラグの和又はXOR関数に基づいて、2つ以上の量子化器から選択される、ことと、

を備える、方法。

【請求項12】

前記第2の変換係数を量子化する前記量子化器が、前記第1の変換係数に基づいて選択される、請求項11に記載の方法。

【請求項13】

前記SIG、第1、第2のフラグのコンテキストモデリングが、前記量子化器または前記量子化器の選択において使用されるステートに基づく、請求項11に記載の方法。

【請求項14】

1つ以上のプロセッサを備える装置であって、前記1つ以上のプロセッサが、

ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスすることであって、前記第1の変換係数が、符号化順序で前記ピクチャの前記ブロック内の第2の変換係数に先行する、ことと、

前記第2の変換係数に関連する第2のパラメータセットにアクセスすることであって、前記第1および第2のパラメータセットが、バイナリ算術コード化エンジンの通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の符号化に依存する、ことと、

前記ブロックの第1のスキャンパスにおける前記第1および第3のパラメータセットをエントロピー符号化することであって、前記第1のスキャンパスが、前記ブロックのエントロピー符号化変換係数の1つ以上の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、少なくともSIGフラグ、第1のフラグおよび第2のフラグを含み、前記SIGフラグが、対応する変換係数がゼロであるか否かを示し、前記第1のフラグが、前記対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記第2のフラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示し、前記第2の変換係数を量子化する量子化器が、前記第1のパラメータセットの前記SIGフラグ、前記第1のフラグ、及び前記第2のフラグの和又はXOR関数に基づいて、2つ以上の量子化器から選択される、ことと

10

20

30

40

50

を行うように構成されている、装置。

【請求項 15】

前記第2の変換係数を量子化する前記量子化器が、前記第1の変換係数に基づいて選択される、請求項14に記載の装置。

【請求項 16】

前記SIG、第1、第2のフラグのコンテキストモデリングが、前記量子化器または前記量子化器の選択において使用される状態に基づく、請求項14に記載の装置。

【発明の詳細な説明】

【技術分野】

【0001】

本実施形態は、一般に、ビデオ符号化または復号のための方法および装置に関する。

【背景技術】

【0002】

高い圧縮効率を達成するために、画像およびビデオの符号化スキームは、通常、予測および変換を用いて、ビデオコンテンツの空間的および時間的冗長性を活用する。一般に、イントラまたはインター予測は、イントラまたはインターピクチャ相関を活用するために使用され、次いで、元のブロックと予測ブロックとの間の差、多くの場合、予測誤差または予測残差と称される差が、変換され、量子化され、エントロピー符号化される。ビデオを再構成するために、圧縮データは、エントロピー符号化、量子化、変換、および予測に対応する逆プロセスによって復号される。

【発明の概要】

【発明が解決しようとする課題】

【0003】

実施形態によれば、ビデオ復号の方法であって、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスすることであって、前記第1の変換係数が、復号順序で前記ピクチャの前記ブロック内の第2の変換係数に先行することと、前記第2の変換係数に関連する第2のパラメータセットにアクセスすることであって、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の復号に依存することと、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー復号することであって、前記第1のスキャンパスが、前記ブロックのエントロピー復号変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、gt1フラグおよびgt2フラグのうちの少なくとも1つを含み、前記gt1フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記gt2フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示すことと、前記復号された変換係数に応答して前記ブロックを再構築することと、を備える、方法が提供される。

【0004】

実施形態によれば、ビデオ符号化の方法であって、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスすることであって、前記第1の変換係数が、符号化順序で前記ピクチャの前記ブロック内の第2の変換係数に先行することと、前記第2の変換係数に関連する第2のパラメータセットにアクセスすることであって、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の符号化に依存することと、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー符号化することであって、前記第1のスキャンパスが、前記ブロックのエントロピー符号化変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、gt1フラグおよびgt2フラグのうちの少なく

とも1つを含み、前記 g_t1 フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 g_t2 フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示すことと、を備える、方法が提供される。

【0005】

別の実施形態によれば、1つ以上のプロセッサを備えるビデオ復号のための装置であって、前記1つ以上のプロセッサが、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスし、前記第1の変換係数が、復号順序で前記ピクチャの前記ブロック内の第2の変換係数に先行し、前記第2の変換係数に関連する第2のパラメータセットにアクセスし、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の復号に依存し、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー復号し、前記第1のスキャンパスが、前記ブロックのエントロピー復号変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、 g_t1 フラグおよび g_t2 フラグのうちの少なくとも1つを含み、前記 g_t1 フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 g_t2 フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示し、前記復号された変換係数に応答して前記ブロックを再構築するように構成されている、装置が提供される。装置は、さらに、前記1つ以上のプロセッサに結合された1つ以上のメモリを備えることができる。

【0006】

別の実施形態によれば、1つ以上のプロセッサを備えるビデオ符号化のための装置であって、前記1つ以上のプロセッサが、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスし、前記第1の変換係数が、符号化順序で前記ピクチャの前記ブロック内の第2の変換係数に先行し、前記第2の変換係数に関連する第2のパラメータセットにアクセスし、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の符号化に依存し、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー符号化し、前記第1のスキャンパスが、前記ブロックのエントロピー符号化変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、 g_t1 フラグおよび g_t2 フラグのうちの少なくとも1つを含み、前記 g_t1 フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 g_t2 フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示すように構成されている、装置が提供される。

【0007】

別の実施形態によれば、ビデオ復号の装置であって、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスする手段であって、前記第1の変換係数が、復号順序で前記ピクチャの前記ブロック内の第2の変換係数に先行する手段と、前記第2の変換係数に関連する第2のパラメータセットにアクセスする手段であって、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の復号に依存する手段と、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー復号する手段であって、前記第1のスキャンパスが、前記ブロックのエントロピー復号変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、 g_t1 フラグおよび g_t2 フラグのうちの少なくとも1つを含み、前記 g_t1 フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 g_t2 フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示す手段と、前記復号された変換係数に応答して前記ブロックを再構築する手段と、を備

える、装置が提供される。

【0008】

別の実施形態によれば、ビデオ符号化の装置であって、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスする手段であって、前記第1の変換係数が、符号化順序で前記ピクチャの前記ブロック内の第2の変換係数に先行する手段と、前記第2の変換係数に関連する第2のパラメータセットにアクセスする手段であって、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の符号化に依存する手段と、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー符号化する手段であって、前記第1のスキャンパスが、前記ブロックのエントロピー符号化変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、 $gt1$ フラグおよび $gt2$ フラグのうちの少なくとも1つを含み、前記 $gt1$ フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 $gt2$ フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示す手段と、を備える、装置が提供される。

10

【0009】

別の実施形態によれば、符号化されたビデオを含む信号であって、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスすることであって、前記第1の変換係数が、符号化順序で前記ピクチャの前記ブロック内の第2の変換係数に先行することと、前記第2の変換係数に関連する第2のパラメータセットにアクセスすることであって、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の符号化に依存することと、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー符号化することであって、前記第1のスキャンパスが、前記ブロックのエントロピー符号化変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、 $gt1$ フラグおよび $gt2$ フラグのうちの少なくとも1つを含み、前記 $gt1$ フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 $gt2$ フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示すことと、を実行することにより、信号が形成される。

20

30

【図面の簡単な説明】

【0010】

【図1】図1は、本実施形態の態様が実装されることができる、システムのブロック図を示している。

【図2】図2は、ビデオ符号化器の実施形態のブロック図を示している。

【図3】図3は、ビデオ復号器の実施形態のブロック図を示している。

【図4】図4は、J V E T - J 0 0 1 4において提案された従属量子化で使用される2つのスカラー量子化器を示す図の例である。

【図5】図5は、J V E T - J 0 0 1 4において提案された従属量子化のための状態遷移および量子化器選択を示す図の例である。

40

【図6】図6は、J V E T - J 0 0 1 4において提案された、C Gにおける係数ビンの順序を示す図の例である。

【図7】図7は、J V E T - K 0 3 1 9において提案された、S I Gベースの状態遷移を示す図の例である。

【図8】図8は、J V E T - K 0 3 1 9において提案された、C Gにおける係数ビンの順序を示す図の例である。

【図9】図9は、一実施形態にかかる、C Gにおける係数ビンの順序を示す図の例である。

【図10】図10は、一実施形態にかかる、S U M (S I G , $gt1$, $gt2$) ベースの状態遷移を示す図の例である。

50

【図 1 1】図 1 1 は、一実施形態にかかる、X O R (S I G、g t 1、g t 2) ベースの状態遷移を示す図の例である。

【図 1 2】図 1 2 は、一実施形態にかかる、g t 1 ベースの状態遷移を示す図の例である。

【図 1 3】図 1 3 は、一実施形態にかかる、現在ブロックを符号化するプロセスを示している。

【図 1 4】図 1 4 は、一実施形態にかかる、現在ブロックを復号するプロセスを示している。

【発明を実施するための形態】

【0 0 1 1】

図 1 は、様々な態様および実施形態が実装されることができるシステムの例のブロック図を示している。システム 1 0 0 は、以下で説明される様々な構成要素を含む装置として具体化されることができ、本出願で説明される態様の 1 つ以上を実行するように構成される。そのような装置の例は、これらに限定されるものではないが、パーソナルコンピュータ、ラップトップコンピュータ、スマートフォン、タブレットコンピュータ、デジタルマルチメディアセットトップボックス、デジタルテレビ受像機、パーソナルビデオ録画システム、コネクテッド家電、およびサーバなどの様々な電子装置を含む。システム 1 0 0 の要素は、単独でも組み合わせでも、単一の集積回路、複数の I C、および／または別個の構成要素に具体化されることができる。例えば、少なくとも 1 つの実施形態において、システム 1 0 0 の処理および符号化器／復号器要素は、複数の I C および／または別個の構成要素にわたって分散している。様々な実施形態において、システム 1 0 0 は、他のシステムに、または他の電子装置に、例えば、通信バスを介して、または専用の入力および／または出力ポートを介して、通信可能に結合される。様々な実施形態において、システム 1 0 0 は、本出願に記載の態様のうちの 1 つ以上を実装するように構成される。

【0 0 1 2】

システム 1 0 0 は、例えば本出願に記載の様々な態様を実装するために、そこにロードされる命令を実行するように構成された少なくとも 1 つのプロセッサ 1 1 0 を含む。プロセッサ 1 1 0 は、埋め込みメモリ、入力出力インターフェース、および当該技術分野において知られているような他の様々な回路を含むことができる。システム 1 0 0 は、少なくとも 1 つのメモリ 1 2 0 (例えば、揮発性メモリ装置、および／または不揮発性メモリ装置) を含む。システム 1 0 0 は、これらに限定されるものではないが、E E P R O M、R O M、P R O M、R A M、D R A M、S R A M、フラッシュ、磁気ディスクドライブ、および／または光ディスクドライブを含む、不揮発性メモリおよび／または揮発性メモリを含むことができる記憶装置 1 4 0 を含む。記憶装置 1 4 0 は、非限定的な例として、内部記憶装置、着脱可能記憶装置、および／またはネットワークアクセス可能記憶装置を含むことができる。

【0 0 1 3】

システム 1 0 0 は、例えば、符号化されたビデオまたは復号されたビデオを提供するようにデータを処理するように構成された符号化器／復号器モジュール 1 3 0 を含み、符号化器／復号器モジュール 1 3 0 は、独自のプロセッサおよびメモリを含むことができる。符号化器／復号器モジュール 1 3 0 は、符号化および／または復号機能を実行する装置に含まれることができるモジュール (複数可) を表す。公知であるように、装置は、符号化モジュールおよび復号モジュールの一方または双方を含むことができる。さらに、符号化器／復号器モジュール 1 3 0 は、システム 1 0 0 の別個の要素として実装されてもよく、または、当業者にとって公知のハードウェアとソフトウェアとの組み合わせとしてプロセッサ 1 1 0 内に組み込まれてもよい。

【0 0 1 4】

本出願に記載の態様を実行するようにプロセッサ 1 1 0 または符号化器／復号器 1 3 0 にロードされることになるプログラムコードは、記憶装置 1 4 0 に記憶され、続いて、プロセッサ 1 1 0 による実行のためにメモリ 1 2 0 上にロードされることができる。様々な実施形態によれば、プロセッサ 1 1 0、メモリ 1 2 0、記憶装置 1 4 0、および符号化器

10

20

30

40

50

／復号器モジュール１３０のうちの１つ以上は、本出願に記載のプロセスの実行中、様々な項目のうちの１つ以上を記憶することができる。そのような記憶される項目は、これらに限定されるものではないが、入力ビデオ、復号されたビデオまたは復号されたビデオの一部、ビットストリーム、行列、変数、ならびに方程式、式、演算、および演算ロジックの処理からの中間結果または最終結果を含むことができる。

【００１５】

いくつかの実施形態において、プロセッサ１１０および／または符号化器／復号器モジュール１３０の内部のメモリを使用して、命令を記憶し、符号化中または復号中に必要とされる処理のためにワーキングメモリを提供する。しかしながら、他の実施形態において、処理装置（例えば、処理装置は、プロセッサ１１０または符号化器／復号器モジュール１３０のいずれかとする）の外部のメモリは、これらの機能のうちの１つ以上のために使用される。外部メモリは、メモリ１２０および／または記憶装置１４０とすることができ、例えば、ダイナミック揮発性メモリおよび／または不揮発性フラッシュメモリとすることができる。いくつかの実施形態において、テレビのオペレーティングシステムを記憶するために外部不揮発性フラッシュメモリが使用される。少なくとも１つの実施形態において、ＭＰＥＧ－２、ＨＥＶＣ、またはＶＶＣなど、ビデオ符号化および復号動作のために、ＲＡＭなどの高速外部ダイナミック揮発性メモリがワーキングメモリとして使用される。

【００１６】

システム１００の要素への入力は、ブロック１０５に示されるような様々な入力装置を介して提供されることができる。そのような入力装置には、これらに限定されるものではないが、（ｉ）例えば、ブロードキャストによって無線を介して送信されたＲＦ信号を受信するＲＦ部、（ｉｉ）コンポジット入力端子、（ｉｉｉ）ＵＳＢ入力端子、および／または（ｉｖ）ＨＤＭＩ入力端子を含む。

【００１７】

様々な実施形態において、ブロック１０５の入力装置は、当該技術分野において知られているような関連するそれぞれの入力処理要素を有する。例えば、ＲＦ部は、（ｉ）所望の周波数を選択する（信号を選択する、またはある周波数帯域に信号を帯域制限する、とも称される）こと、（ｉｉ）選択された信号をダウンコンバートすること、（ｉｉｉ）（例えば）ある特定の実施形態ではチャンネルと称される場合がある信号周波数帯域を選択するように、より狭い周波数帯域に再び帯域制限すること、（ｉｖ）ダウンコンバートされて帯域制限された信号を復調すること、（ｖ）誤り訂正を実行すること、および（ｖｉ）逆多重化して所望のデータパケットストリームを選択することに対して好適な要素に関連付けられることができる。様々な実施形態のＲＦ部は、これらの機能を実行する１つ以上の要素、例えば、周波数セレクタ、信号セレクタ、帯域リミッタ、チャンネルセレクタ、フィルタ、ダウンコンバータ、復調器、誤り訂正器、およびデマルチプレクサを含む。ＲＦ部は、例えば、受信された信号をより低い周波数に（例えば、中間周波数またはベースバンドに近い周波数）、またはベースバンドにダウンコンバートすることを含む、様々なこれらの機能を実行するチューナを含むことができる。１つのセットトップボックスの実施形態において、ＲＦ部およびその関連付けられた入力処理要素は、有線（例えば、ケーブル）媒体経由で送信されたＲＦ信号を受信し、フィルタリングし、ダウンコンバートし、所望の周波数帯域に再びフィルタリングすることによって、周波数選択を実行する。様々な実施形態は、上記（および他の）要素の順序を並べ替え、これらの要素のうちのいくつかを取り除き、および／または同様もしくは異なる機能を実行する他の要素を追加する。要素を追加することは、既存の要素間に要素を挿入すること、例えば、増幅器およびアナログ－デジタル変換器を挿入することを含むことができる。様々な実施形態において、ＲＦ部は、アンテナを含む。

【００１８】

さらに、ＵＳＢおよび／またはＨＤＭＩ端子は、ＵＳＢおよび／またはＨＤＭＩ接続にわたって他の電子装置にシステム１００を接続するためのそれぞれのインターフェースプ

10

20

30

40

50

ロセッサを含むことができる。入力処理の様々な態様、例えばリード・ソロモン誤り訂正が、例えば、必要に応じて、別個の入力処理 IC 内またはプロセッサ 110 内で実装されることができることを理解されたい。同様に、USB または HDMI インターフェース処理の態様は、必要に応じて、別個のインターフェース IC 内またはプロセッサ 110 内で実装されることができる。復調され、誤り訂正され、かつ多重分離されたストリームは、例えば、プロセッサ 110 と、出力装置上での表示のために、必要に応じてデータストリームを処理するようにメモリおよび記憶要素と組み合わせて動作する符号化器／復号器 130 と、を含む様々な処理要素に提供される。

【0019】

システム 100 の様々な要素は、一体型ハウジング内に提供されることができ、一体型ハウジング内では、様々な要素が相互接続され、好適な接続配置 115、例えば、I2C バス、配線、およびプリント回路基板を含む、当該技術分野において知られているような内部バスを使用して、それらの間でデータを送信することができる。

【0020】

システム 100 は、通信チャンネル 190 を介して他の装置との通信を可能にする通信インターフェース 150 を含む。通信インターフェース 150 は、これに限定されるものではないが、通信チャンネル 190 経由でデータを送受信するように構成されたトランシーバを含むことができる。通信インターフェース 150 は、これらに限定されるものではないが、モデムまたはネットワークカードを含むことができ、通信チャンネル 190 は、例えば、有線および／または無線媒体内に実装されることができる。

【0021】

様々な実施形態において、データは、IEEE 802.11 などの Wi-Fi ネットワークを使用して、システム 100 にストリーミングされる。これらの実施形態の Wi-Fi 信号は、Wi-Fi 通信に適合された通信チャンネル 190 および通信インターフェース 150 経由で受信される。これらの実施形態の通信チャンネル 190 は、通常、アプリケーションをストリーミングすることおよび他のオーバーザトップ通信を可能にするためにインターネットを含む外部ネットワークへのアクセスを提供するアクセスポイントまたはルータに接続される。他の実施形態は、入力ブロック 105 の HDMI 接続経由でデータを配信するセットトップボックスを使用して、ストリーミングされたデータをシステム 100 に提供する。さらに他の実施形態は、入力ブロック 105 の RF 接続を使用して、ストリーミングされたデータをシステム 100 に提供する。

【0022】

システム 100 は、ディスプレイ 165、スピーカ 175、および他の周辺装置 185 を含む、様々な出力装置に出力信号を提供することができる。他の周辺装置 185 は、実施形態の様々な例において、スタンドアロン DVR、ディスクプレーヤ、ステレオシステム、照明システム、およびシステム 100 の出力に基づいて機能を提供する他の装置のうちの 1 つ以上を含む。様々な実施形態において、システム 100 と、ディスプレイ 165、スピーカ 175、または他の周辺装置 185 との間で、AV リンク、CEC、またはユーザの介入の有無に関わらず、デバイス・ツー・デバイス制御を可能にする他の通信プロトコルなどのシグナリングを使用して、制御信号が伝送される。出力装置は、それぞれのインターフェース 160、170、および 180 による専用接続を介してシステム 100 に通信可能に結合されることができる。あるいは、出力装置は、通信インターフェース 150 を介して、通信チャンネル 190 を使用してシステム 100 に接続されることができる。ディスプレイ 165 およびスピーカ 175 は、例えば、テレビなどの電子装置内のシステム 100 の他の構成要素と、単一のユニットにおいて一体化されることができる。様々な実施形態において、ディスプレイインターフェース 160 は、ディスプレイドライバ、例えば、タイミングコントローラ (TCON) チップを含む。

【0023】

ディスプレイ 165 およびスピーカ 175 は、例えば、入力 105 の RF 部が別個のセットトップボックスの一部である場合、他の構成要素のうちの 1 つ以上から代替的に分離

10

20

30

40

50

されることができる。ディスプレイ 165 およびスピーカ 175 が外部構成要素である様々な実施形態において、例えば、HDMIポート、USBポート、またはコンポジット (COMP) 出力を含む、専用出力接続を介して出力信号が提供されることができる。

【0024】

図2は、高効率ビデオ符号化 (HEVC) 符号化器などのビデオ符号化器 200 の例を示している。図2はまた、JVET (Joint Video Exploration Team) によって開発中のVVC (バーサタイルビデオ符号化) 符号化器などの、HEVC規格を改良した符号化器またはHEVCに類似した技術を採用した符号化器を示すことができる。

【0025】

本出願では、「再構築された (reconstructed)」および「復号された (decoded)」という用語は、互換的に使用されることができ、「符号化された (encoded)」または「コード化された (coded)」という用語は、互換的に使用されることができ、「画像 (image)」、「ピクチャ (picture) および「フレーム (frame)」という用語は、互換的に使用されることができる。必須ではないが、通常は、「再構築された」という用語は、符号化器側において使用される一方で「復号された」は、復号器側において使用される。

【0026】

符号化される前に、ビデオシーケンスは、事前符号化処理 (201)、例えば、入力色ピクチャへの変換 (例えば、RGB 4:4:4 から YCbCr 4:2:0 への変換) を適用すること、または、(例えば、色成分のうちの1つのヒストグラム等化を使用して) 圧縮に対してより復元力のある信号分布を得るために、入力ピクチャ成分の再マッピングを実行することを経ることができる。メタデータは、事前処理に関連付けられることができ、ビットストリームに添付されることができる。

【0027】

1つ以上のピクチャでビデオシーケンスを符号化するために、ピクチャは、各スライスが1つ以上のスライスセグメントを含むことができる1つ以上のスライスに分割される (202)。HEVCでは、スライスセグメントは、符号化ユニットと、予測ユニットと、変換ユニットと、に編成される。HEVC仕様は、「ブロック」と「ユニット」とを区別し、ここで「ブロック」は、サンプルアレイの特定の領域 (例えば、輝度、Y) をアドレス指定し、「ユニット」は、全ての符号化された色成分 (Y、Cb、Cr、またはモノクロ)、構文要素、およびブロックに関連付けられている予測データ (例えば、動きベクトル) の配列されたブロックを含む。

【0028】

HEVCでのコード化では、ピクチャは、構成可能サイズ (通常、64×64、128×128、または256×256ピクセル) を有する正方形のコード化ツリーブロック (CTB) に分割され、コード化ツリーブロックの連続したセットは、スライスにグループ化される。コード化ツリーユニット (CTU) は、符号化された色成分のCTBを包含する。CTBは、コード化ブロック (CB) への四分木分割のルートであり、コード化ブロックは、1つ以上の予測ブロック (PB) に分割されることができ、変換ブロック (TB) への四分木分割のルートを形成する。4×4よりも大きい変換ブロック (TB) は、係数グループ (CG) と呼ばれる量子化係数の4×4サブブロックに分割される。コード化ブロック、予測ブロック、および変換ブロックに対応して、コード化ユニット (CU) は、予測ユニット (PU) と、変換ユニット (TU) のツリー構造セットと、を含み、PUは、全ての色成分についての予測情報を含み、TUは、各色成分の残差コード化構文構造を含む。輝度成分のCB、PB、およびTBのサイズは、対応するCU、PU、およびTUに適用される。本出願では、「ブロック」という用語は、例えば、CTU、CU、PU、TU、CG、CB、PB、およびTBのいずれかを指すために使用されることができる。さらに、「ブロック」は、H.264/AVCまたは他のビデオコード化規格において規定されているマクロブロックおよびパーティションを指すためにも使用されることがで

10

20

30

40

50

き、より一般的には、様々なサイズのデータのアレイを指すために使用されることができる。

【0029】

符号化器200では、以下に説明されているように、ピクチャが、符号化器要素によって符号化される。符号化されるピクチャは、例えば、CUの単位で処理される。各コード化ユニットは、イントラモードまたはインターモードのいずれかを使用して符号化される。コード化ユニットがイントラモードで符号化されるとき、イントラ予測を実行する(260)。インターモードにおいて、動き推定(275)および動き補償(270)が行われる。符号化器は、イントラモードまたはインターモードのいずれをコード化ユニットの符号化に使用するかを決定し(205)、予測モードフラグによってイントラ／インター決定を示す。予測残差は、元の画像ブロックから予測ブロックを減算することによって計算される(210)。

10

【0030】

次いで、予測残差が変換され(225)、量子化される(230)。量子化された変換係数に加えて、動きベクトルおよび他の構文要素は、ビットストリームを出力するためにエントロピー符号化される(245)。非限定的な例として、コンテキストベースの適応バイナリ算術コード化(CABAC)を使用して、構文要素をビットストリームに符号化することができる。

【0031】

CABACで符号化するために、非バイナリ構文要素の値は、2値化プロセスを介して、ビンストリングと呼ばれるバイナリシーケンスにマッピングされる。ビンの場合、コンテキストモデルが選択される。「コンテキストモデル」は、1つ以上のビンの確率モデルであり、最近コード化されたシンボルの統計に応じて、利用可能なモデルの選択から選択される。各ビンのコンテキストモデルは、コンテキストモデルインデックス(「コンテキストインデックス」とも呼ばれる)によって識別され、異なるコンテキストインデックスが異なるコンテキストモデルに対応する。コンテキストモデルは、各ビンが「1」または「0」である確率を記憶し、適応型または静的型とすることができる。静的モデルは、ビン「0」および「1」に対して等しい確率でコード化エンジンをトリガする。適応コード化エンジンでは、ビンの実際のコード化値に基づいてコンテキストモデルが更新される。適応モデルおよび静的モデルに対応する動作モードは、それぞれ、通常モードおよびバイパスモードと呼ばれる。コンテキストに基づいて、バイナリ算術コード化エンジンは、対応する確率モデルにしたがってビンを符号化または復号する。

20

30

【0032】

スキャンパターンは、2次元ブロックを1次元配列に変換し、サンプルまたは係数の処理順序を定義する。スキャンパスは、特定の構文要素をコード化するために、(選択されたスキャンパターンにしたがった)ブロック内の変換係数にわたる反復である。

【0033】

HEVCでは、TBにわたるスキャンパスは、次に、スキャンパターン(対角、水平、垂直)にしたがって各CGを順番に処理することで構成され、各CG内の16個の係数は、同様に考慮されたスキャン順序にしたがってスキャンされる。スキャンは、TBの最後の有意係数から開始し、DC係数まで全ての係数を処理する。CGは、順次スキャンされる。最大5つのスキャンパスがCGに適用される。各スキャンパスは、以下のようにCG内の係数の構文要素をコード化する：

40

- ・有意係数フラグ(SIG、`significant__coeff__flag`)：係数の有意性(ゼロ／非ゼロ)。

- ・1よりも大きい係数絶対レベルフラグ(`gt1__coeff__abs__level__greater1__flag`)：係数レベルの絶対値が1よりも大きいかどうかを示す。

- ・2つのフラグよりも大きい係数絶対レベル(`gt2__coeff__abs__level__greater2__flag`)：係数レベルの絶対値が2よりも大きいかどうかを示す。

50

・係数符号フラグ (`coeff__sign__flag`) : 有意係数の符号 (0 : 正、1 : 負)。

・係数の残りの絶対レベル (`coeff__abs__level__remaining`) : 係数レベルの絶対値の残りの値 (値が前のパスにおいてコード化された値よりも大きい場合)。特定のCGには、最大8つの`coeff__abs__level__greater1__flag`がコード化されることができ、`coeff__abs__level__greater2__flag`は、大きさが1よりも大きいCGの第1の係数に対してのみコード化される。

【0034】

各スキャンパスでは、前のスキャンパスで決定された必要な場合にのみ構文がコード化される。例えば、係数が有意でない場合、残りのスキャンパスは、その係数に必要ない。最初の3つのスキャンパスのピンは、通常モードでコード化され、コンテキストモデルインデックスは、TB内の特定の係数の位置と、ローカルテンプレートによってカバーされる近傍の以前にコード化された係数の値とに依存する。スキャンパス4および5のピンは、CG内の全てのバイパスピンがグループ化されるように、バイパスモードでコード化される。

【0035】

符号化器はまた、変換をスキップし、 4×4 TUベースで非変換残差信号に直接量子化を適用することができる。符号化器はまた、変換および量子化の双方をバイパスすることもでき、すなわち、残差は、変換または量子化プロセスを適用せずに直接コード化される。直接PCMコード化では、予測は適用されず、コード化ユニットサンプルは、ビットストリームに直接コード化される。

【0036】

符号化器は、符号化されたブロックを復号して、さらに予測するための参照を提供する。量子化された変換係数は、逆量子化され (240)、逆変換され (250)、予測残差を復号する。復号された予測残差と予測ブロックとを組み合わせ (255)、画像ブロックが再構築される。インループフィルタ (265) は、再構築されたピクチャに適用され、例えば、デブロッキング/SAO (サンプル適合オフセット) フィルタリングを実行し、符号化アーティファクトを低減する。フィルタリングされた画像は、参照ピクチャバッファ (280) に記憶される。

【0037】

図3は、HEVC復号器などのビデオ復号器300の例のブロック図を示している。復号器300では、以下に説明されているように、ビットストリームが、復号器要素によって復号される。ビデオ復号器300は、一般に、図2で説明されたような符号化パスの逆の復号パスを実行し、これは、ビデオデータの符号化の一部として、ビデオ復号を実行する。図3はまた、VVC復号器などのHEVC規格を改良した復号器またはHEVCに類似した技術を採用した復号器を示すことができる。

【0038】

特に、復号器の入力は、ビデオ符号化器200によって生成されることができるビデオビットストリームを含む。ビットストリームは、まずエントロピー復号され (330)、変換係数、動きベクトル、ピクチャパーティショニング情報、および他のコード化された情報を取得する。エントロピー符号化にCABACが使用される場合、コンテキストモデルは、符号化器コンテキストモデルと同じ方法で初期化され、構文要素は、コンテキストモデルに基づいてビットストリームから復号される。

【0039】

ピクチャパーティショニング情報は、ピクチャが分割される方法、例えばCTUのサイズ、およびCTUがCUに、および適用可能な場合はPUにできる限り分割される方法を示す。したがって、復号器は、復号されたピクチャパーティショニング情報にしたがって、例えば、ピクチャをCTUに分割 (335) し、各CTUをCUに分割することができる。変換係数は、逆量子化され (340)、逆変換され (350)、予測残差を復号する。

【0040】

復号された予測残差と予測ブロックとを組み合わせ（355）、画像ブロックが再構築される。予測ブロックは、イントラ予測（360）または動き補償予測（すなわち、インター予測）（375）から取得されることができる（370）。インループフィルタ（365）が再構築された画像に適用される。フィルタリングされた画像は、参照ピクチャバッファ（380）に記憶される。

【0041】

復号されたピクチャは、事後復号処理（385）、例えば、逆色変換（例えば、YCbCr 4:2:0からRGB 4:4:4への変換）または事前符号化処理（201）で行われる再マッピングプロセスの逆を実行する逆再マッピングをさらに経ることができる。事後復号処理は、事前符号化処理において導出されかつビットストリームで信号通知された、メタデータを使用することができる。

10

【0042】

従属スカラー量子化は、「Description of SDR, HDR and 360° video coding technology proposal by Fraunhofer HHI」、Document JVET-J0014、第10回ミーティング：米国サンディエゴ、2018年4月10日〜20日（以下、「JVET-J0014」）と題された記事で提案され、異なる再構築レベルを有する2つのスカラー量子化器が量子化のために切り替えられる。従来の独立スカラー量子化（HEVCおよびVTM-1において使用される）と比較して、変換係数の許容可能な再構築値のセットは、再構築順序で現在の変換係数レベルに先行する変換係数レベルの値に依存する。

20

【0043】

従属スカラー量子化のアプローチは、（a）異なる再構築レベルによって2つのスカラー量子化器を定義し、（b）2つのスカラー量子化器を切り替えるプロセスを定義することによって実現される。

【0044】

Q0およびQ1によって示される、使用される2つのスカラー量子化器は、図4に示されている。利用可能な再構築レベルの位置は、量子化ステップサイズ Δ によって一意に指定される。変換係数の実際の再構築が整数演算を使用するという事実を無視すると、2つのスカラー量子化器Q0およびQ1は、以下のように特徴付けられる：

30

Q0：第1の量子化器Q0の再構築レベルは、量子化ステップサイズ Δ の偶数の整数倍によって与えられる。この量子化器が使用されると、再構築された変換係数 t' は、以下にしたがって計算される。

$$t' = 2 \cdot k \cdot \Delta、$$

ここで、 k は、関連する変換係数レベルを示す。「変換係数レベル」（ k ）という用語は、量子化された変換係数値を指し、例えば、それは、以下のresidual_coding構文構造で説明されるようにTransCoeffLevelに対応することに留意されたい。「再構築された変換係数」（ t' ）という用語は、逆量子化された変換係数値を指す。

Q1：第2の量子化器Q1の再構築レベルは、量子化ステップサイズ Δ の奇数の整数倍によって与えられ、さらに、再構築レベルはゼロに等しい。再構築された変換係数 t' への変換係数レベル k のマッピングは、以下によって指定される。

40

$$t' = (2 \cdot k - \text{sgn}(k)) \cdot \Delta、$$

ここで、 $\text{sgn}(\cdot)$ は、符号関数を示し、 $\text{sgn}(x) = (k == 0 ? 0 : (k < 0 ? -1 : 1))$ である。

【0045】

使用されるスカラー量子化器（Q0またはQ1）は、ビットストリームで明示的にシグナリングされない。代わりに、現在の変換係数に使用される量子化器は、コード化／再構築の順序で現在の変換係数に先行する変換係数レベルのパリティによって決定される。

【0046】

50

図5に示されるように、2つのスカラー量子化器（Q0およびQ1）間の切り替えは、4つの状態を有するステートマシンを介して実行される。ステートは、0、1、2、3の4つの異なる値を取ることができる。ステートは、コード化／再構築の順序で現在の変換係数に先行する変換係数レベルのパリティによって一意に決定される。変換ブロックの逆量子化の開始時に、ステートは0に設定される。変換係数は、スキャン順序で（すなわち、エントロピー符号化／復号されるのと同じ順序で）再構築される。現在の変換係数が再構築された後、ステートは、表1に示すように更新され、ここで、kは、変換係数レベルの値を示す。次のステートは、現在のステートと現在の変換係数レベルkのパリティ（ $k \& 1$ ）にのみ依存することに留意されたい。kが現在の変換係数レベルの値を表す場合、ステートの更新は、以下のように記述されることができる。

ステート = stateTransTable [ステート] [$k \& 1$]、

ここで、stateTransTableは、図5および表1に示される表を表し、演算子&は、2の補数演算におけるビット単位の「and」演算子を指定する。

【0047】

ステートは、使用されるスカラー量子化器を一意に指定する。現在の変換係数についてのステートが0または1に等しい場合、スカラー量子化器Q0が使用される。それ以外の場合（ステートは2または3に等しい）、スカラー量子化器Q1が使用される。

【表1】

表1

現在のステート	次のステート・・・	
	$(k \& 1) == 0$	$(k \& 1) == 1$
0	0	2
1	2	0
2	1	3
3	3	1

【0048】

より一般的には、変換係数の量子化器は、3つ以上のスカラー量子化器から選択されることができ、ステートマシンは、5つ以上のステートを有することができる。または、量子化器の切り替えは、他の可能なメカニズムを介して処理されることができる。

【0049】

従属スカラー量子化と組み合わせた係数コード化スキームもまた、JVET-J0014において提案され、これにより、量子化係数のコンテキストモデリングは、使用した量子化器に依存する。具体的には、有意フラグ（SIG）および1よりも大きいフラグ（gt1）のそれぞれは、2つのセットのコンテキストモデルを有し、特定のSIGまたはgt1に対して選択されるセットは、関連する係数に使用される量子化器に依存する。したがって、JVET-J0014において提案された係数コード化は、量子化器、したがって次の係数についてのコンテキストセットを決定するために使用されるパリティを知るために、次のスキャン位置に移動する前に、量子化係数の絶対レベル（absLevel）を完全に再構築する必要がある。すなわち、係数（n+1）をエントロピー復号するためのコンテキストモデルを取得するために、係数n（SIG、gt1、・・・、gt4、符号フラグ、絶対残存レベル）のエントロピー復号を完了する必要がある。その結果、係数（n+1）のいくつかの通常のコッド化ピンは、係数nのいくつかのバイパスコード化ピンの復号を待つ必要があり、したがって、異なる係数のバイパスコード化ピンは、図6に示されるように通常のコッド化ピンとインターリーブされる。

【0050】

図6に示されるように、JVET-J0014における係数コード化の設計は、以下に説明するように、HEVCまたはVTM-1の設計と比較してスループットを低下させる可能性がある：

1. 通常のコッド化ピンが増加する。通常のコッド化ピンは、コンテキストの選択と間

隔の細分割の計算により、バイパスされたコード化ビンよりも処理が遅くなる。J V E T-J 0 0 1 4 の係数コード化では、係数グループ (C G) の通常のコード化ビンの数は、V T M-1 (S I G の場合は 1 6、g t 1 の場合は 8、g t 2 の場合は 1) の 2 5 と比較して最大 8 0 である。

2. バイパスコード化ビンはグループ化されない。バイパスビンをより長いチェーンにグループ化すると、サイクルごとに処理されるビンの数が増えるため、単一のバイパスビンを処理するために必要なサイクル数が減少する。しかしながら、J V E T-J 0 0 1 4 の係数コード化では、C G のバイパスコード化ビンは、グループ化されず、代わりに、それらは、係数ごとに通常のコード化ビンでインターリーブされる。

【0 0 5 1】

本出願は、H E V C および V T M-1 の係数コード化設計として、ほぼ同じレベルのスループットを達成するとともに、従属スカラー量子化によって提供されるゲインのほとんどを維持するためのスカラー量子化器の決定スキームを対象としている。

【0 0 5 2】

寄書 J V E T-K 0 3 1 9 (「C E 7-Related: T C Q with High Throughput Coefficient Coding」、Document J V E T-K 0 3 1 9, J V E T 第 1 1 回ミーティング: スロベニア、リュブリャナ、2 0 1 8 年 7 月 1 0 日-1 8 日、以下「J V E T-K 0 3 1 9」を参照) において、J V E T-J 0 0 1 4 において提案されたパリティベースのステート遷移は、図 7 および表 2 に示されるように S I G ベースのステート遷移に置き換えられるとともに、J V E T-J 0 0 1 4 における他の依存スカラー量子化設計は、変更されないままである。これを行うことにより、現在の変換係数を量子化するために使用されるスカラー量子化器は、スキャン順序で現在の変換係数に先行する量子化係数の S I G によって決定される。

【表 2】

表 2

現在のステート	次のステート . . .	
	s i g == 0	s i g == 1
0	0	2
1	2	0
2	1	3
3	3	1

【0 0 5 3】

J V E T-K 0 3 1 9 において提案された係数コード化は、H E V C および V T M-1 係数コード化に基づいている。相違は、S I G および g t 1 のそれぞれが 2 つのセットのコンテキストモデルを有し、エントロピー符号化器が、関連する係数によって使用される量子化器にしたがって、特定の S I G または g t 1 のコンテキストセットを選択するということである。したがって、従属スカラー量子化のスカラー量子化器をパリティベースから S I G ベースに変更すると、H E V C および V T M-1 係数コード化と同様の高スループット設計が可能になる。C G における係数ビンの提案された順序が図 8 に示されている。すなわち、J V E T-K 0 3 1 9 において提案された係数コード化では、通常のコード化ビンは、C G ごとに最大 2 5 であり、これは、H E V C および V T M-1 係数コード化と同じままであり、C G 内の全てのバイパスコード化ビンは、ともにグループ化される。

【0 0 5 4】

J V E T-J 0 0 1 4 における従属量子化のアプローチは、C E 7 のテスト 7. 2. 1 ソフトウェアでテストされ、シミュレーション結果は、V T M-1. 0 アンカーと比較して、4. 9 9 % A I (全てのイントラ)、3. 4 0 % R A (ランダムアクセス)、および 2. 7 0 % L D B (低遅延 B) の B D レート低下を示している。しかしながら、J V E T-K 0 3 1 9 のシミュレーション結果は、V T M-1. 0 アンカーと比較して、3. 9 8

% A I、2.58% R A、および1.80% L D BのB Dレート低下を示している。すなわち、変換係数をJ V E T-K 0 3 1 9 (S I Gのみに基づくスイッチング)として量子化するために使用されるスカラー量子化器は、J V E T-J 0 0 1 4 (量子化係数の完全なa b s L e v e lに基づくスイッチング)において提案されたものと比較して、コード化効率を低下させる可能性がある。

【0055】

本出願は、高スループットとコード化効率との間の適切なトレードオフを達成するために、従属スカラー量子化に使用されるスカラー量子化器のいくつかの代替決定スキームを提案する。絶対レベルまたはS I G値のパリティを使用する代わりに、ステート遷移と、通常のコード化ビンに基づくコンテキストモデルの選択が提案される。以下では、従属スカラー量子化に使用されるスカラー量子化器を決定するいくつかの実施形態について説明する。

【0056】

J V E T-J 0 0 1 4において提案された従属スカラー量子化の場合、次の係数についての量子化器を決定するために使用されるパリティを知るために、次のスキャン位置に移動する前に、量子化係数の絶対レベル(a b s L e v e l)を完全に再構築する必要がある。したがって、C G内のバイパスコード化ビンは、グループ化されず、それらは、係数ごとに通常のコード化ビンとインターリーブされる。さらに、以下の構文表に示されるように、H E V CおよびV T M-1とは対照的に、変換係数レベルごとの通常のコード化ビンの最大数が増加する(J V E T-J 0 0 1 4において提案されたアプローチでは、変換係数レベルごとに最大5つの通常のコード化ビンが発生する可能性がある)。V T M-1に対する変更は、斜体で示している。エントロピー符号化器は、変換係数レベルの情報に依存しかつ量子化器を決定するために使用される「ステート」にしたがって、特定のS I Gまたはg t 1のコンテキストセットを選択する。ビンのコード化順序は、次の構文で示されており、ここで、関数g e t S i g C t x I d (x C, y C, ステート)は、現在の係数スキャン位置(x C, y C)とステートとに基づいて、構文s i g _ c o e f f _ f l a gのコンテキストを導出するために使用され、d e c o d e S i g C o e f f F l a g (s i g C t x I d)は、関連するコンテキストs i g C t x I dとともに構文s i g _ c o e f f _ f l a gを復号するためのものであり、g e t G r e a t e r 1 C t x I d (x C, y C, ステート)は、現在の係数スキャン位置(x C, y C)とステートとに基づいて構文a b s _ l e v e l _ g t 1 _ f l a gのコンテキストを導出するために使用され、d e c o d e A b s L e v e l G t 1 F l a g (g r e a t e r 1 C t x I d)は、関連するコンテキストg r e a t e r 1 C t x I dとともに構文a b s _ l e v e l _ g t 1 _ f l a gを復号するためのものである。

10

20

30

40

50

【表 3】

residual_coding(x0, y0, log2TrafoSizeX, log2TrafoSizeY, cIdx) {	記述子
if (transform_skip_enabled_flag && !cu_transquant_bypass_flag && (log2TrafoSizeX + log2TrafoSizeY <= (Log2MaxTransformSkipSize < 1)))	
transform_skip_flag[x0][y0][cIdx]	ae(v)
if (CuPredMode[x0][y0] == MODE_INTER && explicit_rdpcm_enabled_flag && (transform_skip_flag[x0][y0][cIdx] cu_transquant_bypass_flag)) {	
explicit_rdpcm_flag[x0][y0][cIdx]	ae(v)
if (explicit_rdpcm_flag[x0][y0][cIdx])	
explicit_rdpcm_dir_flag[x0][y0][cIdx]	ae(v)
}	
last_sig_coeff_x_prefix	ae(v)
last_sig_coeff_y_prefix	ae(v)
if (last_sig_coeff_x_prefix > 3)	
last_sig_coeff_x_suffix	ae(v)
if (last_sig_coeff_y_prefix > 3)	
last_sig_coeff_y_suffix	ae(v)
log2SbSize = (min(log2TrafoSizeX, log2TrafoSizeY) < 2 ? 1 : 2)	
numSbCoeff = 1 << (log2SbSize < 1)	
lastScanPos = numSbCoeff	
lastSubBlock = (1 << (log2TrafoSizeX + log2TrafoSizeY - 2 * log2SbSize)) - 1	
do {	
if (lastScanPos == 0) {	
lastScanPos = numSbCoeff	
lastSubBlock--	
}	
lastScanPos--	
xS = ScanOrder[log2TrafoSizeX - log2SbSize][log2TrafoSizeY - log2SbSize][scanIdx][lastSubBlock][0]	
yS = ScanOrder[log2TrafoSizeX - log2SbSize][log2TrafoSizeY - log2SbSize][scanIdx][lastSubBlock][1]	
xC = (xS << log2SbSize) + ScanOrder[log2SbSize][log2SbSize][scanIdx][lastScanPos][0]	
yC = (yS << log2SbSize) + ScanOrder[log2SbSize][log2SbSize][scanIdx][lastScanPos][1]	
} while ((xC != LastSignificantCoeffX) (yC != LastSignificantCoeffY))	
for (i = lastSubBlock; i >= 0; i--) {	
xS = ScanOrder[log2TrafoSizeX - log2SbSize][log2TrafoSizeY - log2SbSize][scanIdx][lastSubBlock][0]	
yS = ScanOrder[log2TrafoSizeX - log2SbSize][log2TrafoSizeY - log2SbSize][scanIdx][lastSubBlock][1]	
inferSbDcSigCoeffFlag = 0	
ステート = 0	
if ((i < lastSubBlock) && (i > 0)) {	
coded_sub_block_flag[xS][yS]	ae(v)
inferSbDcSigCoeffFlag = 1	
}	

10

20

30

40

50

for (n= (i==lastSubBlock) ? lastScanPos-1 : numSbCoeff-1; n>=0; n--) {	
xC= (xS<<log2SbSize)+ScanOrder [log2SbSize] [log2SbSize] [scanIdx] [n] [0]	
yC= (yS<<log2SbSize)+ScanOrder [log2SbSize] [log2SbSize] [scanIdx] [n] [1]	
if (coded_sub_block_flag [xS] [yS] && (n>0 !inferSbDcSigCoeffFlag)) {	
sigCtxId=getSigCtxId (xC, yC, ステート)	
sig_coeff_flag [xC] [yC]=decodeSigCoeffFlag (sigCtxId)	ae (v)
}	
absLevel [xC] [yC]=0	
if (sig_coeff_flag [xC] [yC]) {	
inferSbDcSigCoeffFlag=0	
absLevel [xC] [yC]=1	
greaterCtxId=getGreaterCtxId (xC, yC, ステート)	
abs_level_gt1_flag [xC] [yC] = decodeAbsLevelGt1Flag (greaterCtxId)	ae (v)
if (abs_level_gt1_flag [xC] [yC]) {	
absLevel [xC] [yC]=2	
abs_level_gt2_flag [xC] [yC]	ae (v)
if (abs_level_gt2_flag [xC] [yC]) {	
absLevel [xC] [yC]=3	
abs_level_gt3_flag [xC] [yC]	ae (v)
if (abs_level_gt3_flag [xC] [yC]) {	
absLevel [xC] [yC]=4	
abs_level_gt4_flag [xC] [yC]	ae (v)
if (abs_level_gt4_flag [xC] [yC]) {	
abs_level_remaining [xC] [yC]	ae (v)
absLevel [xC] [yC]=5+abs_level_remaining [xC] [yC]	
}	
}	
}	
}	
ステート=stateTransTable [ステート] [absLevel [xC] [yC] & 1]	
}	
for (n=numSbCoeff-1; n>=0; n--) {	
xC= (xS<<log2SbSize)+ScanOrder [log2SbSize] [log2SbSize] [scanIdx] [n] [0]	
yC= (yS<<log2SbSize)+ScanOrder [log2SbSize] [log2SbSize] [scanIdx] [n] [1]	
if (sig_coeff_flag [xC] [yC]) {	
coeff_sign_flag [xC] [yC]	ae (v)
TransCoeffLevel [x0] [y0] [cIdx] [xC] [yC]	
=	
absLevel [xC] [yC] * (1 - 2 * coeff_sign_flag [xC] [yC])	
}	
}	
}	
}	

10

20

30

40

【0057】

前に説明したように、これらの構文の変更のために、高スループットのハードウェア実装に関して潜在的な問題がある。我々の提案では、従属スカラー量子化をサポートしながら、HEVCとほぼ同じレベルのスループットを達成するための代替アプローチが提案される。以下では、HEVCを例として使用して、提案された変更を説明する。

【0058】

一実施形態では、変換係数レベルごとの通常のコード化ビンの最大数は、5ではなく3に保たれる（SIG、gt1およびgt2は、通常コード化されている）。各CGの場合、通常のコード化ビンおよびバイパスコード化ビンは、コード化順に分離され、最初に、

50

CGの全ての通常のコード化ビンが送信され、その後、バイパスコード化ビンが送信される。CGにおける係数ビンの提案された順序が図9に示されている。CGのビンは、CGのスキャン位置にわたって複数のパスでコード化される。

・パス1：コード化順に、重要度（SIG、sig_coeff_flag）、1よりも大きいフラグ（gt1、abs_level_gt1_flag）、および2よりも大きいフラグ（gt2、abs_level_gt2_flag）のコード化。1よりも大きいフラグは、sig_coeff_flagが1に等しい場合にのみ存在する。2よりも大きいフラグ（abs_level_gt2_flag）のコード化は、abs_level_gt1_flagが1に等しいスキャン位置に対してのみ実行される。gt1およびgt2の値は、ビットストリームに存在しない場合は0であると推測される。SIG、gt1、およびgt2フラグは、通常モードでコード化され、SIGのコンテキストモデリングの選択は、関連する係数に対してどの状態が選択されているかに依存する。

10

・パス2：abs_level_gt2_flagが1に等しい全てのスキャン位置の構文要素abs_level_remainingのコード化。非バイナリ構文要素は2値化され、結果のビンは、算術符号化エンジンのバイパスモードでコード化される。

・パス3：sig_coeff_flagが1に等しい全てのスキャン位置の符号（coeff_sign_flag）のコード化。符号は、バイパスモードでコード化される。
【0059】

上記の実施形態は、HEVCと比較して提案された変更を示している。変更は、他のソリューションに基づくこともできる。例えば、JVET-J0014がベースとして使用される場合、SIGおよびgt1の双方のコンテキストモデリングは、量子化器の選択に依存し、xよりも大きいフラグ（gtx、x=3および4）は、パス1またはパス2内にコード化されることができる。gt5、gt6、gt7フラグなど、他の通常のコード化ビンが存在する場合、それらは、パス1またはパス2内にコード化されることができる。さらに、パス3の符号（coeff_sign_flag）もまた、通常モードでコード化されることができる。

20

【0060】

2019年3月のミーティングにおいて、JVETは、変換スキップ残差ブロックに新たな残差コード化プロセスを採用した。変換スキップ（TS）が有効になっている場合、予測残差の変換はスキップされる。係数グループ（CG）の残差レベルは、スキャン位置を3回通過するときに以下のようにコード化される：

30

パス1：以下のフラグがシグナリングされる

o sig_coeff_flag

o coeff_sign_flag

o 1よりも大きいフラグ（abs_level_gtx_flag[0]）

o パリティ（par_level_flag）フラグ

パス2：以下のフラグがシグナリングされる

o 3よりも大きいフラグ（abs_level_gtx_flag[1]）

o 5よりも大きいフラグ（abs_level_gtx_flag[2]）

o 7よりも大きいフラグ（abs_level_gtx_flag[3]）

o 9よりも大きいフラグ（abs_level_gtx_flag[4]）

40

パス3：Golomb-Riceコード化を使用して、残りの絶対レベル（abs_remainder）のコード化をバイパスする。

【0061】

上記の提案された実施形態はまた、この新たに採用されたTS残差コード化にも適用されることができ、例えば、以下に示すように、3よりも大きいフラグの位置は、第1のパスに移動される。

パス1：以下のフラグがシグナリングされる

o sig_coeff_flag

o coeff_sign_flag

50

o 1よりも大きいフラグ (abs_level_gt_x_flag [0])
 o パリティ (par_level_flag) フラグ
 o 3よりも大きいフラグ (abs_level_gt_x_flag [1])
 パス2：以下のフラグがシグナリングされる
 o 5よりも大きいフラグ (abs_level_gt_x_flag [2])
 o 7よりも大きいフラグ (abs_level_gt_x_flag [3])
 o 9よりも大きいフラグ (abs_level_gt_x_flag [4])
 パス3：Golomb-Riceコード化を使用して、残りの絶対レベル (abs_remainder) のコード化をバイパスする。
 【0062】

10

実施形態1－関数SUM(SIG, gt1, gt2)に基づくスカラー量子化器決定スキーム

【0063】
 高スループットのハードウェア実装の問題を解決するために、量子化係数の絶対レベル (absLevel) の完全な再構築は、状態を決定するために実行されず、2つのスカラー量子化器間の切り替えは、完全な変換係数の絶対レベルのパリティに依存しない。前述のように、J V E T - K 0 3 1 9の場合と同様にSIGによってのみ決定されるスカラー量子化器は、コード化効率を低下させる可能性がある。一実施形態では、現在の変換係数のSIG、gt1およびgt2値をともに考慮に入れる関数SUM(SIG, gt1, gt2)に基づいてスカラー量子化器を決定することを提案する。

20

【表4】

表3

マーキング値m	gt2	gt1	SIG	可能な変換絶対レベル
0	0	0	0	0
1	0	0	1	1
2	0	1	1	2
3	1	1	1	≥ 3

【0064】

30

変換係数のSIG、gt1、およびgt2値の可能な組み合わせが表3に示されている。これらの4つの異なる組み合わせから4つの可能なマーキングレベル値にマッピングされた1対1の対応が存在し、ここで、mは、これらの4つの可能なケースのマーキング値を示す。SIG、gt1、およびgt2の値からマーキング値mを導出する関数は、以下のように記述されることができる：

$$m = \text{SUM}(\text{SIG}, \text{gt1}, \text{gt2}) = \text{SIG} + \text{gt1} + \text{gt2}.$$

【0065】

図10に示されるように、2つのスカラー量子化器間の切り替えは、符号化／再構築順序で現在の変換係数に先行する変換係数レベルのマーキング値mのパリティによって一意に決定される。変換ブロックの逆量子化の開始時に、状態は0に設定される。現在の変換係数の通常のコード化ビンが再構築された後、状態は、図10および表4に示されるように更新される。次の状態は、現在の状態と、現在の変換係数レベルのマーキング値mのパリティ (m&1) にのみ依存することに留意されたい。状態の更新は、以下のように記述されることができる：

40

状態 = stateTransTable [状態] [m&1]、

ここで、stateTransTableは、図10および表4に示される表を表し、演算子&は、2の補数演算におけるビット単位の「and」演算子を指定する。

50

【表 5】

表 4

現在のステート	次のステート . . .	
	(m & 1) == 0	(m & 1) == 1
0	0	2
1	2	0
2	1	3
3	3	1

10

【表 6】

residual_coding (x0, y0, log2TrafoSizeX, log2TrafoSizeY, cIdx) {	記述子
...	
ステート = 0	
for (i = lastSubBlock; i >= 0; i--) {	
...	
//====第1のパス: sig、gt1、gt2の通常ビン====	
for (n = (i == lastSubBlock) ? lastScanPos - 1 : numSbCoeff - 1; n >= 0; n--) {	
m[xC][yC] = 0	
xC = (xS << log2SbSize) + ScanOrder[log2SbSize][log2SbSize][scanIdx][n][0]	
yC = (yS << log2SbSize) + ScanOrder[log2SbSize][log2SbSize][scanIdx][n][1]	
if (coded_sub_block_flag[xS][yS] && (n > 0 !inferSbDcSigCoeffFlag)) {	
sigCtxId = getSigCtxId(xC, yC, ステート)	
sig_coeff_flag[xC][yC] = decodeSigCoeffFlag(sigCtxId)	ae(v)
m[xC][yC] += sig_coeff_flag[xC][yC]	
}	
absLevel[xC][yC] = 0	
if (sig_coeff_flag[xC][yC]) {	
absLevel[xC][yC] = 1	
abs_level_gt1_flag[xC][yC]	ae(v)
m[xC][yC] += abs_level_gt1_flag[xC][yC]	
if (abs_level_gt1_flag[xC][yC]) {	
absLevel[xC][yC] = 2	
abs_level_gt2_flag[xC][yC]	ae(v)
m[xC][yC] += abs_level_gt2_flag[xC][yC]	
}	
}	
ステート = stateTransTable[ステート][m[xC][yC] & 1]	
}	
//====第1のパスの終わり: sig、gt1、gt2の通常ビン====	
//====第2のパス: 残りのバイパスビン====	
for (n = numSbCoeff - 1; n >= 0; n--) {	
xC = (xS << log2SbSize) + ScanOrder[log2SbSize][log2SbSize][scanIdx][n][0]	
yC = (yS << log2SbSize) + ScanOrder[log2SbSize][log2SbSize][scanIdx][n][1]	
if (abs_level_gt2_flag[xC][yC]) {	
abs_level_remaining[xC][yC]	ae(v)
absLevel[xC][yC] = 3 + abs_level_remaining[xC][yC]	
}	
}	

20

30

40

50

//=====第2のパスの終わり：残りのパイパスピン=====	
//=====第3のパス：符号=====	
for (n=numSbCoeff-1; n>=0; n--) {	
xC = (xS<<log2SbSize) + ScanOrder [log2SbSize] [log2SbSize] [scanIdx] [n] [0]	
yC = (yS<<log2SbSize) + ScanOrder [log2SbSize] [log2SbSize] [scanIdx] [n] [1]	
if (sig_coeff_flag [xC] [yC]) {	
coeff_sign_flag [xC] [yC]	ae (v)
TransCoeffLevel [x0] [y0] [cIdx] [xC] [yC]	
=	
absLevel [xC] [yC] * (1-2*coeff_sign_flag [xC] [yC])	
}	
}	
//=====第3のパスの終わり：符号=====	
}	
}	

10

【0066】

20

変換および量子化の後、ほとんどの変換係数の大きさは、通常、非常に低い値になる。表3および表4に示されるように、変換係数の絶対レベルが3よりも小さい場合、提案された方法は、JVET-J0014とほぼ同じ結果を達成することができる。一方、提案された方法は、量子化係数の絶対レベル (absLevel) の完全な再構築を必要とせず、高スループットのハードウェア実装の問題を解決した。

【0067】

コード化の順序とピンの存在、および送信されたデータからの変換係数レベルの再構築の詳細は、上記の構文表に示されている。説明を簡単にするために、スキャン位置の異なるパスが構文表においてコメント化されている。HEVCおよびVTM-1に関連する変更は、斜体で示している。エントロピー符号化器は、変換係数レベルの情報に依存しかつ量子化器を決定するために使用される「ステート」にしたがって、特定のSIGのコンテキストセットを選択する。

30

【0068】

実施形態2-関数XOR (SIG, gt1, gt2)に基づくスカラー量子化器決定スキーム

【0069】

別の実施形態では、現在の変換係数のSIG、gt1およびgt2値が考慮され、関数XOR (SIG, gt1, gt2)に基づいてスカラー量子化器が選択される。SIG、gt1およびgt2の値から排他的論理和値xを導出する関数は、以下のように記述されることができる：

40

$$x = \text{XOR}(\text{SIG}, \text{gt1}, \text{gt2}) = \text{SIG} \wedge \text{gt1} \wedge \text{gt2}.$$

変換係数のSIG、gt1、およびgt2の値の可能な組み合わせに対応する排他的論理和値xが表5に提示される。

50

【表 7】

表 5

g t 2	g t 1	S I G	排他的論理和値 x
0	0	0	0
0	0	1	1
0	1	1	0
1	1	1	1

【0070】

第1の実施形態と比較して、2つのスカラー量子化器間の切り替えは、S I G、g t 1 および g t 2 フラグの排他的論理和値 x によって一意に決定される。状態の更新は、以下のように記述されることができる：

状態 = stateTransTable [状態] [x]、

ここで、stateTransTable は、図 11 および表 6 に示されている表を表す。状態マシンの他の部分は、以前のアプローチと同様のままである。

【表 8】

表 6

現在の状態	次の状態・・・	
	x == 0	x == 1
0	0	2
1	2	0
2	1	3
3	3	1

【0071】

表 5 および表 6 に示すように、変換係数の絶対レベルが 3 よりも小さい場合、提案された方法は、J V E T - J 0 0 1 4 とほぼ同じ結果を達成することができる。一方、提案された方法は、量子化係数の絶対レベル (a b s L e v e l) の完全な再構築を必要とせず、高スループットのハードウェア実装の問題を解決した。

【0072】

実施形態 3 - 通常のコード化ビンの 1 つに基づくスカラー量子化器決定スキーム

【0073】

上記の実施形態によれば、現在の変換係数 (S I G、g t 1 および g t 2) の全ての通常のコード化ビンは、スカラー量子化器を決定するために考慮される。別の実施形態では、2つのスカラー量子化器間の切り替えは、通常のコード化ビンの 1 つ、例えば、g t 1 フラグに基づくことができる。この実施形態では、図 12 に示されるように、前の状態遷移は、g t 1 ベースの状態遷移によって置き換えられることができる一方で、他の設計は、変更されないままである。これにより、現在の変換係数を量子化するために使用されるスカラー量子化器は、スキャン順序で現在の変換係数に先行する量子化係数の g t 1 フラグによって決定される。状態の更新は、以下のように記述されることができる：

状態 = stateTransTable [状態] [g t 1]、

ここで、stateTransTable は、図 12 および表 7 に示されている表を表す。

【0074】

この実施形態では、C G のピンは、以下の C G 内のスキャン位置上の 3 つのスキャンパスでコード化される：s i g、g t 1 および g t 2 の第 1 のパス、残りの絶対レベルの第 2 のパス、および符号情報の第 3 のパス。変形例では、C G のピンは、以下の C G のスキ

ャン位置上の4つのスキャンパスでコード化される：s i gおよびg t 1の第1のパス、g t 2の第2のパス、残りの絶対レベルの第3のパス、および符号情報の第4のパス。この変形は、前の実施形態で提案された3つのスキャンパスと比較して、ビン間の依存性をさらに低減することができる。

【表 9】

表 7

現在のステート	次のステート . . .	
	g t 1 == 0	g t 1 == 1
0	0	2
1	2	0
2	1	3
3	3	1

10

【0075】

あるいは、現在の変換係数を量子化するために使用されるスカラー量子化器は、スキャン順序で現在の変換係数に先行する量子化係数のg t 2フラグによって決定される。より一般的には、現在の変換係数を量子化するために使用されるスカラー量子化器は、スキャン順序で現在の変換係数に先行する量子化係数の1つの通常のコード化ビン（例えば、g t xフラグ）によって決定される。

20

【0076】

上記の例では、係数に3つの通常のコード化ビン（S I G，g t 1，g t 2）を使用するH E V Cに基づくいくつかの実施形態を示している。係数の通常のコード化ビンがH E V Cのものと異なる場合、提案された実施形態は、変換係数ごとに異なる数（3よりも多いまたは少ない）の通常のコード化ビンを取ることによって実行されることができる。

【0077】

上記において、和関数および排他的論理和関数が考慮され、提案された実施形態はまた、変換係数レベルごとに通常のコード化ビンからの異なるステート更新導出（1 / 0）関数を用いて実行されることができる。

30

【0078】

上記では、説明は、主に逆量子化に関するものである。それに応じて量子化が調整されることに留意されたい。使用されるスカラー量子化器（Q 0またはQ 1）は、ビットストリームにおいて明示的にシグナリングされない。例えば、符号化器側の量子化モジュールは、ステートに基づいて現在の変換係数に使用する量子化器を選択する。現在の変換係数のステートが0または1に等しい場合、スカラー量子化器Q 0が使用される。それ以外の場合（ステートは2または3に等しい）、スカラー量子化器Q 1が使用される。ステートは、図10～図12に記載の方法を使用して、変換係数レベルの情報によって一意に決定され、復号器は、ビットストリームを適切に復号するために同じ量子化器を選択する。

【0079】

40

図13は、実施形態にかかる、現在の符号化ユニットを符号化する方法（1300）を示している。ステップ1305において、初期ステートは、ゼロに設定される。コード化ユニットを符号化するために、コード化ユニット内の係数がスキャンされる。コード化ユニットのスキャンパスは、コード化ユニットの各C Gをスキャンパターン（対角、水平、垂直）にしたがって順番に処理し、各C G内の係数も、同様に考慮されたスキャン順序にしたがってスキャンされる。スキャンは、コード化ユニットの最後の有意係数を有するC Gにおいて開始（1315）し、D C係数を有する最初のC Gまで全ての係数を処理する。

【0080】

C Gが最後の有意係数またはD C係数を含まない場合（1320）、C Gがゼロ以外の係数を含むかどうかを示すフラグ（c o d e d _ s u b _ b l o c k _ f l a g）が符号

50

化される (1325)。最後の非ゼロのレベルまたはDC係数を含むCGの場合、`coded_sub_block_flag`は、1に等しいと推測され、ビットストリームには表示されない。

【0081】

`coded_sub_block_flag`が真の場合 (1330)、3つのスキャンパスがCGに適用される。第1のパス (1335-1360) では、係数について、SIGフラグ (`sig_coeff_flag`) が符号化される (1335)。SIGフラグを符号化するために、コンテキストモードインデックスは、例えば、`sigCtxId=getSigCtxId` (ステート) などのステートを使用して決定される。SIGフラグが真の場合 (1340)、`gt1`フラグ (`abs_level_gt1_flag`) が符号化される (1345)。`gt1`フラグが真の場合 (1350)、`gt2`フラグ (`abs_level_gt2_flag`) が符号化される (1355)。SIG、`gt1`および`gt2`フラグのうちの1つ以上に基づいて、例えば、図10～図12に記載されている方法を使用して、ステートが更新される (1360)。

10

【0082】

第2のスキャンパス (1365、1370) では、符号化器は、`gt2`フラグが真であるかどうかをチェックする (1365)。真の場合、残りの絶対レベル (`abs_level_remaining`) が符号化される (1370)。第3のスキャンパス (1375、1380) では、符号化器は、SIGフラグが真であるかどうかをチェックする (1375)。真の場合、符号フラグ (`coeff_sign_flag`) が符号化される (1380)。ステップ1385において、符号化器は、処理されるべきCGがさらにあるかどうかをチェックする。そうである場合、処理される次のCGに移動する (1390)。

20

【0083】

図14は、実施形態にかかる、現在の符号化ユニットを復号する方法 (1400) を示している。ステップ1405において、初期ステートは、ゼロに設定される。符号化器側と同様に、コード化ユニットを復号するために、コード化ユニット内の係数位置がスキャンされる。スキャンは、コード化ユニットの最後の有意係数を有するCGで開始し (1415)、DC係数を有する最初のCGまで全ての係数を処理する。

【0084】

CGが最後の有意係数またはDC係数を含まない場合 (1420)、CGが任意の非ゼロの係数を含むかどうかを示すフラグ (`coded_sub_block_flag`) が復号される (1425)。最後の非ゼロレベルまたはDC係数を含むCGの場合、`coded_sub_block_flag`は、1に等しいと推測される。

30

【0085】

`coded_sub_block_flag`が真の場合 (1430)、3つのスキャンパスがCGに適用される。第1のパス (1435-1460) では、係数について、SIGフラグ (`sig_coeff_flag`) が復号される (1435)。SIGフラグを復号するために、コンテキストモードインデックスは、例えば、`sigCtxId=getSigCtxId` (ステート) などのステートを使用して決定される。SIGフラグが真の場合 (1440)、`gt1`フラグ (`abs_level_gt1_flag`) が復号される (1445)。`gt1`フラグが真の場合 (1450)、`gt2`フラグ (`abs_level_gt2_flag`) が復号される (1455)。SIG、`gt1`および`gt2`フラグのうちの1つ以上に基づいて、例えば、図10～図12に記載されている方法を使用して、ステートが更新される (1460)。

40

【0086】

第2のスキャンパス (1470、1475) では、復号器は、`gt2`フラグが真であるかどうかをチェックする (1470)。真の場合、残りの絶対レベル (`abs_level_remaining`) が復号される (1475)。第3のスキャンパス (1480、1485) では、復号器は、SIGフラグが真であるかどうかをチェックする (1480)。真の場合、符号フラグ (`coeff_sign_flag`) が復号される (1485)。

50

）。ステップ 1487 において、復号器は、利用可能なSIG、gt1、gt2、符号フラグ、および残りの絶対値に基づいて、変換係数を計算する。

【0087】

復号器は、処理されるべきCGがまだあるかどうかをチェックする（1490）。そうである場合、処理される次のCGに移動する（1495）。全ての係数がエントロピー復号される場合、変換係数は、依存スカラー量子化を使用して逆量子化される（1497）。変換係数のために使用されたスカラー量子化器（Q0またはQ1）は、ステートによって決定され、これは、復号された変換係数レベルの情報とともに、図10～図12に記載された方法を使用して導出される。

【0088】

様々な方法が、本明細書に記載されており、それらの方法のそれぞれは、説明された方法を達成するための1つ以上のステップまたは動作を備える。本方法の正しい動作のために特定の順序のステップまたは動作が必要でない限り、特定のステップおよび／または動作の順序および／または使用は、変更されてもよく、または組み合わせられてもよい。さらに、「第1の」、「第2の」などの用語は、例えば「第1の復号」および「第2の復号」など、要素、構成要素、ステップ、動作などを変更するために様々な実施形態において使用されることができる。そのような用語の使用は、特に必要な場合を除き、変更された動作の順序を意味するものではない。したがって、この例では、第1の復号は、第2の復号の前に実行される必要はなく、例えば、第2の復号との重複期間の前、最中、または重複する期間に発生することができる。

【0089】

本出願に説明されている様々な方法および他の態様を使用して、図2および図3に示されるように、モジュール、例えば、ビデオ符号化器200および復号器300のエントロピー符号化および復号モジュール（245、330）を変更することができる。さらに、本態様は、VVCまたはHEVCに限定されるものではなく、例えば、他の標準規格および推奨、ならびに任意のそのような標準規格および推奨に適用されることができる。特に指示されていない限り、または技術的に除外されていない限り、本出願で説明される態様は、個別にまたは組み合わせて使用されることができる。

【0090】

本出願では様々な数値が使用される。特定の値は、例示的な目的のためであり、記載された態様は、これらの特定の値に限定されるものではない。

【0091】

実施形態によれば、ビデオ復号の方法であって、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスすることであって、前記第1の変換係数が、復号順序で前記ピクチャの前記ブロック内の第2の変換係数に先行することと、前記第2の変換係数に関連する第2のパラメータセットにアクセスすることであって、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の復号に依存することと、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー復号することであって、前記第1のスキャンパスが、前記ブロックのエントロピー復号変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、gt1フラグおよびgt2フラグのうちの少なくとも1つを含み、前記gt1フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記gt2フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示すことと、前記復号された変換係数に応答して前記ブロックを再構築することと、を備える、方法が提供される。

【0092】

実施形態によれば、ビデオ符号化の方法であって、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスすることであって、前記第1の変換係

10

20

30

40

50

数が、符号化順序で前記ピクチャの前記ブロック内の第2の変換係数に先行することと、前記第2の変換係数に関連する第2のパラメータセットにアクセスすることであって、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の符号化に依存することと、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー符号化することであって、前記第1のスキャンパスが、前記ブロックのエントロピー符号化変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、 $gt1$ フラグおよび $gt2$ フラグのうちの少なくとも1つを含み、前記 $gt1$ フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 $gt2$ フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示すことと、を備える、方法が提供される。

【0093】

別の実施形態によれば、1つ以上のプロセッサを備えるビデオ復号のための装置であって、前記1つ以上のプロセッサが、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスし、前記第1の変換係数が、復号順序で前記ピクチャの前記ブロック内の第2の変換係数に先行し、前記第2の変換係数に関連する第2のパラメータセットにアクセスし、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の復号に依存し、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー復号し、前記第1のスキャンパスが、前記ブロックのエントロピー復号変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、 $gt1$ フラグおよび $gt2$ フラグのうちの少なくとも1つを含み、前記 $gt1$ フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 $gt2$ フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示し、前記復号された変換係数に応答して前記ブロックを再構築するように構成されている、装置が提供される。装置は、さらに、前記1つ以上のプロセッサに結合された1つ以上のメモリを備えることができる。

【0094】

別の実施形態によれば、1つ以上のプロセッサを備えるビデオ符号化のための装置であって、前記1つ以上のプロセッサが、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスし、前記第1の変換係数が、符号化順序で前記ピクチャの前記ブロック内の第2の変換係数に先行し、前記第2の変換係数に関連する第2のパラメータセットにアクセスし、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の符号化に依存し、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー符号化し、前記第1のスキャンパスが、前記ブロックのエントロピー符号化変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、 $gt1$ フラグおよび $gt2$ フラグのうちの少なくとも1つを含み、前記 $gt1$ フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 $gt2$ フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示すように構成されている、装置が提供される。

【0095】

別の実施形態によれば、ビデオ復号の装置であって、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスする手段であって、前記第1の変換係数が、復号順序で前記ピクチャの前記ブロック内の第2の変換係数に先行する手段と、前記第2の変換係数に関連する第2のパラメータセットにアクセスする手段であって、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第

2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の復号に依存する手段と、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー復号する手段であって、前記第1のスキャンパスが、前記ブロックのエントロピー復号変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、 g_{t1} フラグおよび g_{t2} フラグのうちの少なくとも1つを含み、前記 g_{t1} フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 g_{t2} フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示す手段と、前記復号された変換係数に応答して前記ブロックを再構築する手段と、を備える、装置が提供される。

10

【0096】

別の実施形態によれば、ビデオ符号化の装置であって、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスする手段であって、前記第1の変換係数が、符号化順序で前記ピクチャの前記ブロック内の第2の変換係数に先行する手段と、前記第2の変換係数に関連する第2のパラメータセットにアクセスする手段であって、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の符号化に依存する手段と、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー符号化する手段であって、前記第1のスキャンパスが、前記ブロックのエントロピー符号化変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、 g_{t1} フラグおよび g_{t2} フラグのうちの少なくとも1つを含み、前記 g_{t1} フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 g_{t2} フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示す手段と、を備える、装置が提供される。

20

【0097】

別の実施形態によれば、符号化されたビデオを含む信号であって、ピクチャのブロック内の第1の変換係数に関連する第1のパラメータセットにアクセスすることであって、前記第1の変換係数が、符号化順序で前記ピクチャの前記ブロック内の第2の変換係数に先行することと、前記第2の変換係数に関連する第2のパラメータセットにアクセスすることであって、前記第1および第2のパラメータセットが、通常モードでエントロピー符号化され、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータのコンテキストモデリングが、前記第1の変換係数の符号化に依存することと、前記ブロックの第1のスキャンパスにおける前記第1および第2のパラメータセットをエントロピー符号化することであって、前記第1のスキャンパスが、前記ブロックのエントロピー符号化変換係数の他のスキャンパスの前に実行され、前記第1および第2の変換係数についての前記第1および第2のパラメータセットの各セットが、 g_{t1} フラグおよび g_{t2} フラグのうちの少なくとも1つを含み、前記 g_{t1} フラグが、対応する変換係数の絶対値が1よりも大きいかどうかを示し、前記 g_{t2} フラグが、前記対応する変換係数の絶対値が2よりも大きいかどうかを示すことと、を実行することにより、信号が形成される。

30

40

【0098】

実施形態によれば、前記第2の変換係数の前記第2のパラメータセットの少なくともパラメータの前記コンテキストモデリングは、前記第1の変換係数の前記第1のパラメータセットの復号に依存し、(1)前記第1の変換係数を表すために使用されかつ(2)バイパスモードでエントロピー符号化されたパラメータとは無関係である。

【0099】

実施形態によれば、SIGフラグはまた、第1のスキャンパスにおいて符号化または復号され、前記SIGフラグは、前記対応する変換係数がゼロであるかどうかを示す。

【0100】

実施形態によれば、前記 g_{t1} フラグは、第1のスキャンパスにおいて符号化または復

50

号され、前記 $g_t 2$ フラグは、第 2 のスキャンパスにおいて符号化または復号される。

【0101】

実施形態によれば、前記第 2 の変換係数を逆量子化するための逆量子化器は、前記第 1 の変換係数に基づいて、2 つ以上の量子化器の間で選択される。

【0102】

実施形態によれば、前記逆量子化器は、前記第 1 の変換係数の前記第 1 のパラメータセットに基づいて選択される。

【0103】

実施形態によれば、前記第 2 の変換係数の前記第 2 のパラメータセット内の少なくともパラメータのコンテキストモデリングは、前記第 1 および第 2 の変換係数の復号に依存する。

【0104】

実施形態によれば、前記逆量子化器は、前記 SIG 、 $g_t 1$ 、および $g_t 2$ フラグの和に基づいて選択される。

【0105】

実施形態によれば、前記逆量子化器は、前記 SIG 、 $g_t 1$ 、および $g_t 2$ フラグの XOR 関数に基づいて選択される。

【0106】

実施形態によれば、前記逆量子化器は、前記 $g_t 1$ フラグ、前記 $g_t 2$ フラグ、または前記対応する変換係数の絶対値が x よりも大きいかどうかを示す $g_t x$ フラグに基づいて選択される。

【0107】

実施形態によれば、(1) 前記ブロック内の変換係数を表すために使用されるパラメータおよび (2) バイパスモードでコード化されるパラメータは、パラメータが (1) 前記ブロック内の変換係数を表すために使用されかつ (2) 前記通常モードでコード化された後の 1 つ以上のスキャンパスでエントロピー符号化または復号される。

【0108】

実施形態によれば、前記 SIG 、 $g_t 1$ 、 $g_t 2$ フラグまたは $g_t x$ フラグのコンテキストモデリングは、量子化器または量子化器の選択において使用される状態に基づく。

【0109】

実施形態は、1 つ以上のプロセッサによって実行されると、1 つ以上のプロセッサに、上述した実施形態のいずれかにかかる符号化方法または復号方法を実行させる命令を備えるコンピュータプログラムを提供する。本実施形態のうちの 1 つ以上はまた、上述した方法のいずれかにしたがってビデオデータを符号化または復号する命令を記憶したコンピュータ可読記憶媒体を提供する。1 つ以上の実施形態はまた、上述した方法にしたがって生成されたビットストリームを記憶したコンピュータ可読記憶媒体を提供する。1 つ以上の実施形態はまた、上述した方法にしたがって生成されたビットストリームを送信または受信する方法および装置を提供する。

【0110】

様々な実装が復号を伴う。本出願で使用される「復号」は、例えば、表示に適した最終出力を生成するために、受信した符号化されたシーケンスで実行されるプロセスの全てまたは一部を包含することができる。様々な実施形態では、そのようなプロセスは、復号器によって通常実行されるプロセスのうちの 1 つ以上、例えば、エントロピー復号、逆量子化、逆変換、および差分復号を含む。「復号プロセス」という語句が、具体的に動作のサブセットを指すことを意図しているか、または一般により広い復号プロセスを指すことを意図しているかは、特定の説明のコンテキストに基づいて明確になり、当業者によって十分に理解されると考えられる。

【0111】

様々な実装は符号化を伴う。「復号」に関する上記の考察と同様に、本出願で使用される「符号化」は、例えば、符号化されたビットストリームを生成するために入力ビデオシ

10

20

30

40

50

ーケンスで実行されるプロセスの全てまたは一部を包含することができる。

【0112】

例えば、`sig__coeff__flag`、`abs__level__gt1__flag`などの本明細書で使用される構文要素は、説明的な用語であることに留意されたい。したがって、それらは、他の構文要素名の使用を排除するものではない。

【0113】

本明細書で説明された実装および態様は、例えば、方法もしくはプロセス、装置、ソフトウェアプログラム、データストリーム、または信号で実装されることができる。単一形式の実装のコンテキストにおいて考察される（例えば、方法としてのみ考察される）のみの場合であっても、考察される特徴の実装はまた、他の形態（例えば、装置またはプログラム）で実装されてもよい。装置は、例えば、適切なハードウェア、ソフトウェア、およびファームウェアで実装されることができる。この方法は、例えば、装置、例えば、コンピュータ、マイクロプロセッサ、集積回路、またはプログラマブルロジック装置を含む、一般に処理装置を指す、例えば、プロセッサなどに実装されることができる。プロセッサはまた、通信装置、例えば、コンピュータ、携帯電話、ポータブル／パーソナルデジタルアシスタンス（「PDA」）、およびエンドユーザ間の情報の伝達を容易にする他の装置も含む。

【0114】

「一実施形態」もしくは「実施形態」、または「一実装」もしくは「実装」、ならびにそれらの他の変形への言及は、実施形態に関連して説明された特定の特徴、構造、特性などが、少なくとも1つの実施形態に含まれることを意味する。したがって、本文書全体にわたって様々な箇所においてみられる、「一実施形態では」もしくは「実施形態では」または「一実装では」もしくは「実装では」という句、ならびに任意の他の変形の出現は、必ずしも全てが同じ実施形態を指しているわけではない。

【0115】

さらに、本出願は、情報の様々な部分を「決定すること」に言及する場合がある。情報を決定することは、例えば、情報を推定すること、情報を計算すること、情報を予測すること、またはメモリから情報を検索することのうちの1つ以上を含むことができる。

【0116】

さらに、本出願は、情報の様々な部分に「アクセスすること」に言及する場合がある。情報にアクセスすることは、例えば、情報を受信すること、情報を検索すること（例えば、メモリから）、情報を記憶すること、情報を移動させること、情報をコピーすること、情報を計算すること、情報を決定すること、情報を予測すること、または情報を推定することのうちの1つ以上を含むことができる。

【0117】

さらに、本出願は、情報の様々な部分を「受信すること」に言及する場合がある。受信することは、「アクセスすること」と同様に、広義の用語であることが意図されている。情報を受信することは、例えば、情報にアクセスすること、または（例えば、メモリから）情報を検索することのうちの1つ以上を含むことができる。さらに、「受信すること」は、典型的には、何らかの方法で、例えば、情報を記憶すること、情報を処理すること、情報を送信すること、情報を移動させること、情報をコピーすること、情報を消去すること、情報を計算すること、情報を決定すること、情報を予測すること、または情報を推定することなどの動作中に含まれる。

【0118】

例えば、「A／B」、「Aおよび／またはB」、ならびに「AおよびBのうちの少なくとも1つ」の場合、次の「／」、「および／または」、ならびに「のうちの少なくとも1つ」のいずれかの使用は、最初に挙げた選択肢（A）のみの選択、または2番目に挙げた選択肢（B）のみの選択、または双方の選択肢（AおよびB）の選択を網羅することを意図していることが分かるはずである。さらなる例として、「A、B、および／またはC」ならびに「A、BおよびCのうちの少なくとも1つ」の場合、そのような言い回しは、最

10

20

30

40

50

初に挙げた選択肢（Ａ）のみの選択、または２番目に挙げた選択肢（Ｂ）のみの選択、または３番目に挙げた（Ｃ）のみの選択、または最初および２番目に挙げた選択肢（ＡおよびＢ）のみの選択、または最初および３番目に挙げた選択肢（ＡおよびＣ）のみの選択、または２番目および３番目に挙げた選択肢（ＢおよびＣ）のみの選択、または３つ全ての選択肢（Ａ、ＢおよびＣ）の選択、を網羅することを意図している。これは、当業者にとって明らかなように、挙げられる項目の数だけ拡張されることができる。

【０１１９】

当業者にとって明らかであるように、実装は、例えば、記憶または送信されることができる情報を搬送するようにフォーマットされる多種多様な信号を生成することができる。情報は、例えば、方法を実行する命令、または説明される実装のうちの１つにより生成されたデータを含むことができる。例えば、信号は、説明された実施形態のビットストリームを搬送するようにフォーマットされてもよい。そのような信号は、例えば、電磁波（例えば、スペクトルの無線周波数部分を使用して）として、またはベースバンド信号としてフォーマットされてもよい。フォーマットすることは、例えば、データストリームを符号化することと、搬送波を符号化データストリームで変調することと、を含むことができる。信号が搬送する情報は、例えば、アナログ情報またはデジタル情報とすることができる。信号は、既知の如く、多種多様な異なる有線リンクまたは無線リンクを介して送信されることができる。信号は、プロセッサ可読媒体に記憶されることができる。

10

20

30

40

50

【図 5】

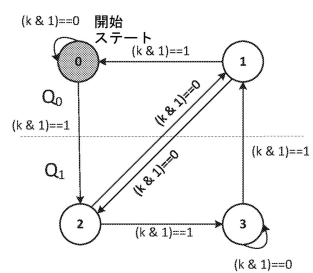


图 5

【図 6】

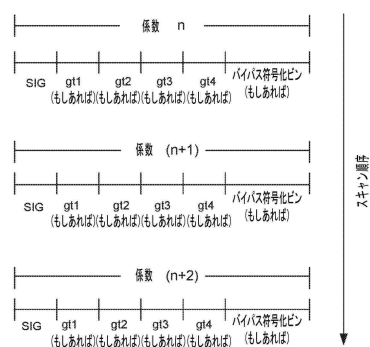


图 6

【圖 7】

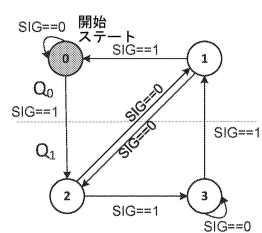
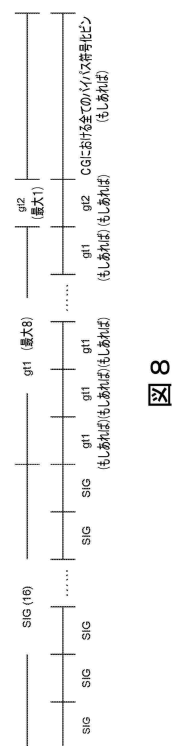
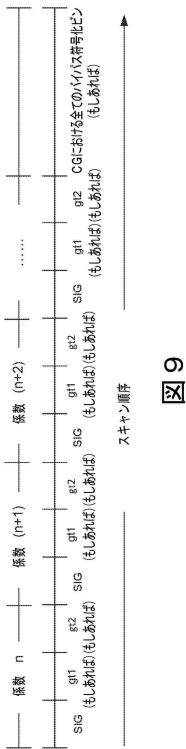


图 7

【図 8】



【図 9】



【図 10】

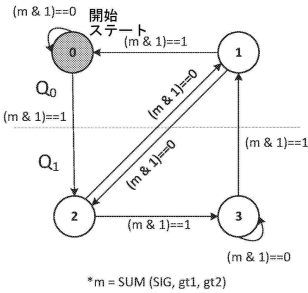


図 10

10

【図 11】

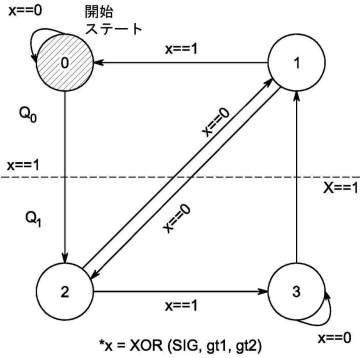


図 11

【図 12】

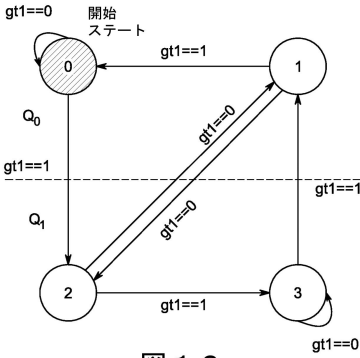


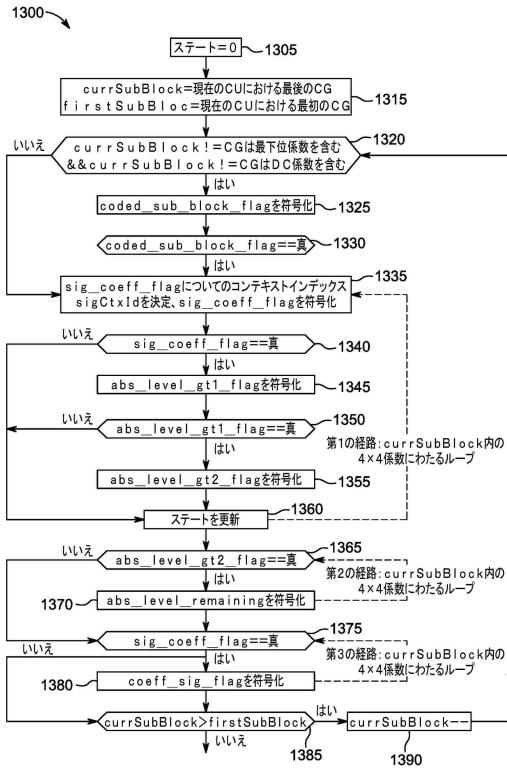
図 12

30

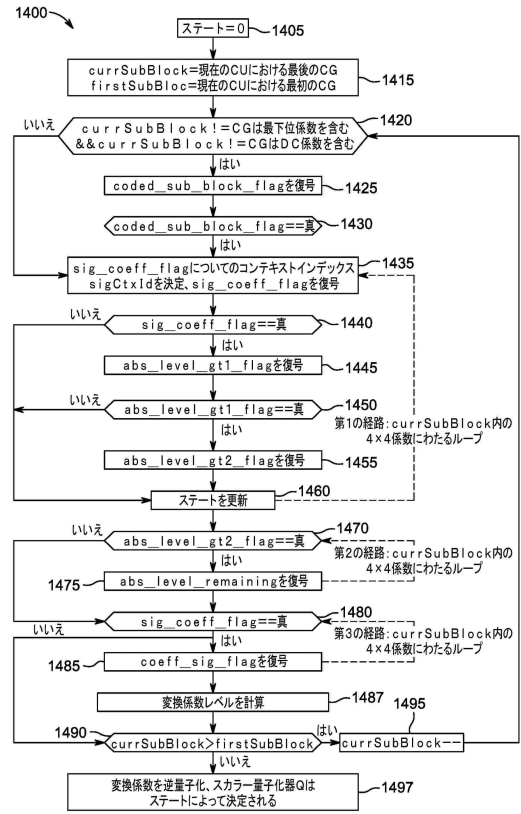
40

50

【図 13】



【図 14】



10

20

30

40

50

フロントページの続き

(51)国際特許分類

H 0 4 N 19/18 (2014.01)

F I

H 0 4 N 19/18

(33)優先権主張国・地域又は機関

欧州特許庁(EP)

弁理士 阿部 豊隆

(72)発明者 チェン, ヤ

フランス国, エフー3 5 5 7 6 セソンーセビニエ, セーエス 1 7 6 1 6, ザック デ シャン ブ
ラン, アベニュー デ シャン ブラン 9 7 5

(72)発明者 ル リアネック, ファブリス

フランス国, エフー3 5 5 7 6 セソンーセビニエ, セーエス 1 7 6 1 6, ザック デ シャン ブ
ラン, アベニュー デ シャン ブラン 9 7 5

(72)発明者 ボワリエ, タンギ

フランス国, エフー3 5 5 7 6 セソンーセビニエ, セーエス 1 7 6 1 6, ザック デ シャン ブ
ラン, アベニュー デ シャン ブラン 9 7 5

(72)発明者 ド ラグランジュ, フィリップ

フランス国, エフー3 5 5 7 6 セソンーセビニエ, セーエス 1 7 6 1 6, ザック デ シャン ブ
ラン, アベニュー デ シャン ブラン 9 7 5

審査官 田部井 和彦

(56)参考文献

国際公開第2 0 2 0 / 0 0 7 7 8 5 (WO, A 1)

特表2 0 2 1 - 5 3 0 1 5 5 (JP, A)

特開2 0 1 8 - 1 1 0 4 0 5 (JP, A)

Heiko Schwarz et al., CE7: Transform Coefficient Coding and Dependent Quantization (Tes
ts 7.1.2, 7.2.1) [online], JVET-K0071 (JVET-K0071.doc) , [2023年10月20日検索], イン
ターネット <URL: https://jvet-experts.org/doc_end_user/documents/11_Ljubljana/wg11/JVET-K0071-v2.zip> , 2018年07月11日Heiko Schwarz et al., Non-CE7: Alternative Entropy Coding for Dependent Quantization [on
line], JVET-K0072 (JVET-K0072.doc) , [2023年10月14日検索], インターネット <URL:
https://jvet-experts.org/doc_end_user/documents/11_Ljubljana/wg11/JVET-K0072-v1.zip
> , 2018年07月03日Heiko Schwarz et al., Hybrid Video Coding with Trellis-Coded Quantization [online], Publi
shed in: 2019 Data Compression Conference (DCC), [2023年10月20日検索], 2019年05月
13日, pp.182-191, インターネット <URL: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8712696>> , DOI: 10.1109/DCC.2019.00026

(58)調査した分野 (Int.Cl., DB名)

H 0 4 N 19 / 13

H 0 4 N 19 / 15 7

H 0 4 N 19 / 9 1

H 0 4 N 19 / 17 6

H 0 4 N 19 / 12

H 0 4 N 19 / 18

I E E E X p l o r e