

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第6843844号  
(P6843844)

(45) 発行日 令和3年3月17日(2021.3.17)

(24) 登録日 令和3年2月26日(2021.2.26)

(51) Int.Cl.

F 1

H04N 19/85 (2014.01)  
H04N 19/70 (2014.01)H04N 19/85  
H04N 19/70

請求項の数 30 (全 98 頁)

(21) 出願番号 特願2018-514786 (P2018-514786)  
 (86) (22) 出願日 平成28年9月20日 (2016.9.20)  
 (65) 公表番号 特表2018-533278 (P2018-533278A)  
 (43) 公表日 平成30年11月8日 (2018.11.8)  
 (86) 國際出願番号 PCT/US2016/052633  
 (87) 國際公開番号 WO2017/053277  
 (87) 國際公開日 平成29年3月30日 (2017.3.30)  
 審査請求日 令和1年8月26日 (2019.8.26)  
 (31) 優先権主張番号 62/221,586  
 (32) 優先日 平成27年9月21日 (2015.9.21)  
 (33) 優先権主張国・地域又は機関  
米国(US)  
 (31) 優先権主張番号 62/236,804  
 (32) 優先日 平成27年10月2日 (2015.10.2)  
 (33) 優先権主張国・地域又は機関  
米国(US)

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

最終頁に続く

(54) 【発明の名称】ビデオコーディングにおけるコンポーネントのレンジ調整の固定小数点実装

## (57) 【特許請求の範囲】

## 【請求項 1】

ビデオデータを処理する方法であって、前記方法が、  
 逆ダイナミックレンジ調整プロセスを実行するための1つまたは複数のパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を受信することと、ここにおいて、前記1つまたは複数のパラメータが、レンジパラメータ、スケールパラメータ、またはオフセットパラメータのうちの1つまたは複数を備え、前記1つまたは複数のシンタックス要素を受信することが、1つまたは複数のシンタックス構造で前記1つまたは複数のシンタックス要素を受信することを備え、前記情報が、1つまたは複数の色コンポーネントについての最小値および最大値を含み、

10

前記1つまたは複数の色コンポーネントについての前記最小値および前記最大値に基づいて、前記1つまたは複数のパラメータを決定することと、

ビデオデータを復号することと、

前記1つまたは複数のパラメータに従って、固定小数点算出を使用して、前記復号ビデオデータに対して前記逆ダイナミックレンジ調整プロセスを実行することとを備える、方法。

## 【請求項 2】

前記情報が、前記レンジパラメータを決定するために使用される小数ビットの第1の数、前記スケールパラメータを決定するために使用される小数ビットの第2の数、および前記オフセットパラメータを決定するために使用される小数ビットの第3の数のうちの1つ

20

または複数をさらに示す、請求項1に記載の方法。

【請求項3】

小数ビットの前記第1の数と、小数ビットの前記第2の数と、小数ビットの前記第3の数とを使用して前記1つまたは複数のパラメータを決定することをさらに備える、請求項2に記載の方法。

【請求項4】

前記1つまたは複数のパラメータを決定することは、

小数ビットの前記第1の数、小数ビットの前記第2の数、または小数ビットの前記第3の数のうちの少なくとも2つが互いに異なる場合、前記1つまたは複数のパラメータを決定するために使用される任意の中間計算プロセス中に任意の小数ビットを累算することによって、前記1つまたは複数のパラメータを決定することと、

所定の小数正確さに基づいて、前記1つまたは複数のパラメータを決定するための最終結果をクリッピングすることとをさらに備える、請求項3に記載の方法。

【請求項5】

前記1つまたは複数のパラメータを決定することが、

前記1つまたは複数のパラメータを決定するために使用されるすべての前記中間計算プロセス中に所望の小数正確さを越える任意の小数ビットをトランケートすることとをさらに備える、請求項4に記載の方法。

【請求項6】

前記情報が、前記復号ビデオデータの前記1つまたは複数の色コンポーネントのためのサンプル値の所定のレンジに対するインデックスを含み、前記方法が、

前記インデックスに基づいて、前記復号ビデオデータの前記1つまたは複数の色コンポーネントについての前記最小値および前記最大値を決定することと、

前記決定された最小値と前記決定された最大値とに基づいて前記1つまたは複数のパラメータを決定することとをさらに備える、請求項1に記載の方法。

【請求項7】

前記1つまたは複数のパラメータが符号付きであるのか符号なしであるのかを示すシンタックス要素を受信することと、

前記1つまたは複数のシンタックス構造の前記情報に対してパースプロセスを実行することと、ここにおいて、前記パースプロセスが、前記シンタックス要素の値にかかわらず同じである、をさらに備える、請求項1に記載の方法。

【請求項8】

前記復号ビデオデータに対して前記逆ダイナミックレンジ調整プロセスを実行した後に前記復号ビデオデータを表示することとをさらに備える、請求項1に記載の方法。

【請求項9】

ビデオデータを処理するように構成された装置であって、前記装置が、

復号ビデオデータを記憶するように構成されたメモリと、

1つまたは複数のプロセッサと

を備え、前記1つまたは複数のプロセッサが、

逆ダイナミックレンジ調整プロセスを実行するための1つまたは複数のパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を受信することと、ここにおいて、前記1つまたは複数のパラメータが、レンジパラメータ、スケールパラメータ、またはオフセットパラメータのうちの1つまたは複数を備え、

前記1つまたは複数のシンタックス要素を受信することが、1つまたは複数のシンタックス構造で前記1つまたは複数のシンタックス要素を受信することを備え、前記情報が、1つまたは複数の色コンポーネントについての最小値および最大値を含み、

前記1つまたは複数の色コンポーネントについての前記最小値および前記最大値に基づ

10

20

30

40

50

いて、前記 1 つまたは複数のパラメータを決定することと、

ビデオデータを復号することと、

前記 1 つまたは複数のパラメータに従って、固定小数点算出を使用して、前記復号ビデオデータに対して前記逆ダイナミックレンジ調整プロセスを実行することとを行うように構成された、装置。

【請求項 1 0】

前記情報が、前記レンジパラメータを決定するために使用される小数ビットの第 1 の数、前記スケールパラメータを決定するために使用される小数ビットの第 2 の数、および前記オフセットパラメータを決定するために使用される小数ビットの第 3 の数のうちの 1 つまたは複数を さらに 示す、請求項 9 に記載の装置。

10

【請求項 1 1】

前記 1 つまたは複数のプロセッサが、  
小数ビットの前記第 1 の数と、小数ビットの前記第 2 の数と、小数ビットの前記第 3 の数とを使用して前記 1 つまたは複数のパラメータを決定することと  
を行うようにさらに構成された、請求項 1 0 に記載の装置。

【請求項 1 2】

前記 1 つまたは複数のパラメータを決定するために、前記 1 つまたは複数のプロセッサは、

小数ビットの前記第 1 の数、小数ビットの前記第 2 の数、または小数ビットの前記第 3 の数のうちの少なくとも 2 つが互いに異なる場合、前記 1 つまたは複数のパラメータを決定するために使用される任意の中間計算プロセス中に任意の小数ビットを累算することによって、前記 1 つまたは複数のパラメータを決定することと、

20

所定の小数正確さに基づいて、前記 1 つまたは複数のパラメータを決定するための最終結果をクリッピングすることと  
を行うようにさらに構成された、請求項 1 1 に記載の装置。

【請求項 1 3】

前記 1 つまたは複数のパラメータを決定するために、前記 1 つまたは複数のプロセッサが、

前記 1 つまたは複数のパラメータを決定するために使用されるすべての前記中間計算プロセス中に所望の小数正確さを越える任意の小数ビットをトランケートすることと  
を行うようにさらに構成された、請求項 1 2 に記載の装置。

30

【請求項 1 4】

前記情報が、前記復号ビデオデータの前記 1 つまたは複数の色コンポーネントのためのサンプル値の所定のレンジに対するインデックスを含み、ここにおいて、前記 1 つまたは複数のプロセッサが、

前記インデックスに基づいて、前記復号ビデオデータの前記 1 つまたは複数の色コンポーネントについての前記最小値および前記最大値を決定することと、

前記決定された最小値と前記決定された最大値とに基づいて前記 1 つまたは複数のパラメータを決定することと

を行うようにさらに構成された、請求項 9 に記載の装置。

40

【請求項 1 5】

前記 1 つまたは複数のプロセッサが、

前記 1 つまたは複数のパラメータが符号付きであるのか符号なしであるのかを示すシンタックス要素を受信することと、

前記 1 つまたは複数のシンタックス構造の前記情報に対してパースプロセスを実行することと、ここにおいて、前記パースプロセスが、前記シンタックス要素の値にかかわらず同じである、を行うようにさらに構成された、請求項 9 に記載の装置。

【請求項 1 6】

前記装置は、

前記 1 つまたは複数のプロセッサが、前記復号ビデオデータに対して前記逆ダイナミック

50

クレンジ調整プロセスを実行した後に、前記復号ビデオデータを表示するように構成されたディスプレイをさらに備える、請求項9に記載の装置。

【請求項17】

前記装置が、カメラ、コンピュータ、モバイルデバイス、ブロードキャスト受信機デバイス、またはセットトップボックスのうちの1つまたは複数を備える、請求項9に記載の装置。

【請求項18】

ビデオデータを処理する方法であって、前記方法が、

1つまたは複数の色コンポーネントについての最小値および最大値を使用してダイナミックレンジ調整プロセスのための1つまたは複数のパラメータを決定することと、ここにおいて、前記1つまたは複数のパラメータが、レンジパラメータ、スケールパラメータ、またはオフセットパラメータのうちの1つまたは複数を含み、

前記決定された1つまたは複数のパラメータおよび固定小数点算出を使用してビデオデータに対して前記ダイナミックレンジ調整プロセスを実行することと、

固定小数点算出を使用して、前記ダイナミックレンジ調整プロセスに対して、逆ダイナミックレンジ調整プロセスを実行するための前記1つまたは複数のパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を発生することと、ここにおいて、前記情報が、前記1つまたは複数の色コンポーネントについての前記最小値および前記最大値を示し、前記1つまたは複数のシンタックス要素を発生することが、1つまたは複数のシンタックス構造で前記1つまたは複数のシンタックス要素を発生することを備え、

前記ダイナミックレンジ調整プロセスを実行した後に前記ビデオデータを符号化することと、

を備える、方法。

【請求項19】

前記情報が、前記レンジパラメータを決定するために使用される小数ビットの第1の数、前記スケールパラメータを決定するために使用される小数ビットの第2の数、および前記オフセットパラメータを決定するために使用される小数ビットの第3の数のうちの1つまたは複数をさらに示す、請求項18に記載の方法。

【請求項20】

前記情報が、前記復号ビデオデータのサンプル値の所定のレンジに対するインデックスをさらに含む、請求項18に記載の方法。

【請求項21】

カメラを用いて前記ビデオデータをキャプチャすることをさらに備える、請求項18に記載の方法。

【請求項22】

ビデオデータを処理するように構成された装置であって、前記装置が、ビデオデータを記憶するように構成されたメモリと、1つまたは複数のプロセッサとを備え、前記1つまたは複数のプロセッサが、

1つまたは複数の色コンポーネントについての最小値および最大値を使用してダイナミックレンジ調整プロセスのための1つまたは複数のパラメータを決定することと、ここにおいて、前記1つまたは複数のパラメータが、レンジパラメータ、スケールパラメータ、またはオフセットパラメータのうちの1つまたは複数を含み、

前記決定された1つまたは複数のパラメータおよび固定小数点算出を使用してビデオデータに対して前記ダイナミックレンジ調整プロセスを実行することと、

固定小数点算出を使用して、前記ダイナミックレンジ調整プロセスに対して、逆ダイナミックレンジ調整プロセスを実行するための前記1つまたは複数のパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を発生

10

20

30

40

50

することと、ここにおいて、前記情報が、前記1つまたは複数の色コンポーネントについての前記最小値および前記最大値を示し、前記1つまたは複数のシンタックス要素を発生することが、1つまたは複数のシンタックス構造で前記1つまたは複数のシンタックス要素を発生することを備え、

前記逆ダイナミックレンジ調整プロセスを実行した後に前記ビデオデータを符号化することと、

を行うように構成された、装置。

【請求項23】

前記情報が、前記レンジパラメータを決定するために使用される小数ビットの第1の数、前記スケールパラメータを決定するために使用される小数ビットの第2の数、および前記オフセットパラメータを決定するために使用される小数ビットの第3の数のうちの1つまたは複数をさらに示す、請求項22に記載の装置。 10

【請求項24】

前記情報が、前記復号ビデオデータのサンプル値の所定のレンジに対するインデックスをさらに含む、請求項22に記載の装置。

【請求項25】

前記ビデオデータをキャプチャするように構成されたカメラをさらに備える、請求項22に記載の装置。

【請求項26】

前記装置が、カメラ、コンピュータ、モバイルデバイス、ブロードキャスト受信機デバイス、またはセットトップボックスのうちの1つまたは複数を備える、請求項22に記載の装置。 20

【請求項27】

前記1つまたは複数のシンタックス構造が、補足エンハンスメント情報(S EI)メッセージまたはパラメータセットのうちの少なくとも1つを備える、請求項1に記載の方法。

【請求項28】

前記1つまたは複数のシンタックス構造が、前記パラメータセットを備え、前記パラメータセットが、ビデオパラメータセット(VPS)、シーケンスパラメータセット(SPS)、またはピクチャパラメータセット(PPS)のうちの少なくとも1つを備える、請求項27に記載の方法。 30

【請求項29】

前記1つまたは複数のシンタックス構造が、補足エンハンスメント情報(S EI)メッセージまたはパラメータセットのうちの少なくとも1つを備える、請求項9に記載の装置。

【請求項30】

前記1つまたは複数のシンタックス構造が、前記パラメータセットを備え、前記パラメータセットが、ビデオパラメータセット(VPS)、シーケンスパラメータセット(SPS)、またはピクチャパラメータセット(PPS)のうちの少なくとも1つを備える、請求項29に記載の装置。 40

【発明の詳細な説明】

【技術分野】

【0001】

[0001] 本出願は、その各々の内容全体が参照により本明細書に組み込まれる、2015年9月21日に出願された米国仮出願第62/221,586号、2015年10月2日に出願された米国仮出願第62/236,804号、および2015年10月13日に出願された米国仮出願第62/241,063号の利益を主張する。

【0002】

[0002] 本開示はビデオ処理に関する。

【背景技術】

10

20

30

40

50

## 【0003】

[0003] デジタルビデオ能力は、デジタルテレビジョン、デジタルダイレクトブロードキャストシステム、ワイヤレスブロードキャストシステム、携帯情報端末（PDA）、ラップトップまたはデスクトップコンピュータ、タブレットコンピュータ、電子ブックリーダー、デジタルカメラ、デジタル記録デバイス、デジタルメディアプレーヤ、ビデオゲームデバイス、ビデオゲームコンソール、セルラーまたは衛星無線電話、いわゆる「スマートフォン」、ビデオ遠隔会議デバイス、ビデオストリーミングデバイスなどを含む、広範囲にわたるデバイスに組み込まれ得る。デジタルビデオデバイスは、MPEG-2、MPEG-4、ITU-T H.263、ITU-T H.264/MPEG-4，Part 10，アドバンストビデオコーディング（AVC：Advanced Video Coding）、ITU-T H.265，高効率ビデオコーディング（HEVC：High Efficiency Video Coding）によって定義された規格、およびそのような規格の拡張に記載されているビデオコーディング技法など、ビデオコーディング技法を実装する。ビデオデバイスは、そのようなビデオコーディング技法を実装することによって、デジタルビデオ情報をより効率的に送信、受信、符号化、復号、および／または記憶し得る。10

## 【0004】

[0004] ビデオコーディング技法は、ビデオシーケンスに固有の冗長性を低減または除去するための空間（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）予測を含む。ブロックベースのビデオコーディングでは、ビデオスライス（たとえば、ビデオフレームまたはビデオフレームの一部分）が、ツリーブロック、コーディングユニット（CU：coding unit）および／またはコーディングノードと呼ばれることもある、ビデオブロック（video block）に区分され得る。ピクチャのイントラコード化（I）スライス中のビデオブロックは、同じピクチャ中の隣接ブロック中の参照サンプルに対する空間予測を使用して符号化される。ピクチャのインターフォーム（PまたはB）スライス中のビデオブロックは、同じピクチャ中の隣接ブロック中の参照サンプルに対する空間予測、または他の参照ピクチャ中の参照サンプルに対する時間予測を使用し得る。ピクチャはフレームと呼ばれることがあり、参照ピクチャは参照フレームと呼ばれることがある。20

## 【0005】

[0005] 空間予測または時間予測は、コーディングされるべきブロックのための予測ブロックを生じる。残差データは、コーディングされるべき元のブロックと予測ブロックとの間のピクセル差分を表す。インターフォーム（IF）コード化ブロックは、予測ブロックを形成する参照サンプルのブロックを指す動きベクトルと、コード化ブロックと予測ブロックとの間の差分を示す残差データとに従って符号化される。イントラコード化ブロックは、イントラコーディングモードと残差データとに従って符号化される。さらなる圧縮のために、残差データは、ピクセル領域から変換領域に変換され、残差変換係数が生じ得、その残差変換係数は、次いで量子化され得る。最初に2次元アレイで構成される量子化された変換係数は、変換係数の1次元ベクトルを生成するために走査され得、なお一層の圧縮を達成するために、エントロピーコーディングが適用され得る。30

## 【0006】

[0006] キャプチャ（capture）され、コーディングされ、表示され得る色値の総数が、色域（color gamut）によって定義され得る。色域は、デバイスがキャプチャする（たとえば、カメラ）か、または再現する（たとえば、ディスプレイ）ことができる、色のレンジ（range）を指す。しばしば、色域はデバイスごとに異なる。ビデオコーディングのために、ビデオコーディングプロセスにおける各デバイスが同じ色域中でピクセル値を処理するように構成され得るように、ビデオデータ（video data）のためのあらかじめ定義された色域が使用され得る。いくつかの色域は、ビデオコーディングのために旧来使用されてきた色域よりも大きいレンジの色を用いて定義される。より大きいレンジの色を用いたそのような色域は、広色域（WCG：wide color gamut）と呼ばれることがある。40

## 【0007】

[0007] ビデオデータの別のアスペクトはダイナミックレンジ（dynamic range）であ50

る。ダイナミックレンジは、一般に、ビデオ信号の最大輝度 (maximum brightness) と最小輝度 (minimum brightness) (たとえば、ルミナンス (luminance)) との間の比として定義される。過去に使用された一般的なビデオデータのダイナミックレンジは、標準ダイナミックレンジ (S D R : standard dynamic range) を有すると見なされる。ビデオデータのための他の例示的な仕様は、最大輝度と最小輝度との間のより大きい比を有する色データを定義する。そのようなビデオデータは、高ダイナミックレンジ (H D R : high dynamic range) を有するものとして説明される。

【発明の概要】

【0008】

[0008] 本開示は、固定小数点実装 (fixed-point implementation) を使用して、ビデオデータのコンポーネントのダイナミックレンジ調整 (dynamic range adjustment) を実装するための例示的な技法およびデバイスについて説明する。説明される技法は、H.264 / AVC、H.265 / HEVC に限定されないビデオコーディング規格、および高ダイナミックレンジ (H D R) コンテンツを符号化および復号するように構成された他の規格に適用可能である。 10

【0009】

[0009] 本開示の一例では、ビデオデータを処理する方法は、逆ダイナミックレンジ調整プロセス (inverse dynamic range adjustment process) を実行するためのパラメータ (parameter) をどのように決定すべきかを指定する情報を含んでいる 1 つまたは複数のシンタックス要素 (syntax element) を受信することと、復号ビデオデータ (decoded video data) を受信することと、受信された情報に従って、固定小数点算出 (fixed-point computing) を使用して、復号ビデオデータに対して逆ダイナミックレンジ調整プロセスを実行することとを備える。 20

【0010】

[0010] 本開示の別の例では、ビデオデータを処理するように構成された装置は、復号ビデオデータを記憶するように構成されたメモリと、1 つまたは複数のプロセッサとを備え、1 つまたは複数のプロセッサは、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる 1 つまたは複数のシンタックス要素を受信することと、復号ビデオデータを受信することと、受信された情報に従って、固定小数点算出を使用して、復号ビデオデータに対して逆ダイナミックレンジ調整プロセスを実行することとを行うように構成される。 30

【0011】

[0011] 本開示の別の例では、ビデオデータを処理するように構成された装置は、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる 1 つまたは複数のシンタックス要素を受信するための手段と、復号ビデオデータを受信するための手段と、受信された情報に従って、固定小数点算出を使用して、復号ビデオデータに対して逆ダイナミックレンジ調整プロセスを実行するための手段とを備える。

【0012】

[0012] 別の例では、本開示は、実行されたとき、ビデオデータを処理するように構成されたデバイスの 1 つまたは複数のプロセッサに、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる 1 つまたは複数のシンタックス要素を受信することと、復号ビデオデータを受信することと、受信された情報に従って、固定小数点算出を使用して、復号ビデオデータに対して逆ダイナミックレンジ調整プロセスを実行することとを行わせる命令 (instruction) を記憶するコンピュータ可読記憶媒体 (computer-readable storage medium) について説明する。 40

【0013】

[0013] 本開示の別の例では、ビデオデータを処理する方法は、固定小数点算出 (fixed-point computing) を使用して、ビデオデータに対してダイナミックレンジ調整プロセス (dynamic range adjustment process) を実行することと、固定小数点算出を使用して 50

、ダイナミックレンジ調整プロセスに対して、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を発生することとを備える。

【0014】

[0014] 本開示の別の例では、ビデオデータを処理するように構成された装置は、ビデオデータを記憶するように構成されたメモリと、1つまたは複数のプロセッサとを備え、1つまたは複数のプロセッサは、固定小数点算出を使用して、ビデオデータに対してダイナミックレンジ調整プロセスを実行することと、固定小数点算出を使用して、ダイナミックレンジ調整プロセスに対して、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を発生することとを行うように構成される。 10

【0015】

[0015] 本開示の別の例では、ビデオデータを処理するように構成された装置は、固定小数点算出を使用して、ビデオデータに対してダイナミックレンジ調整プロセスを実行するための手段と、固定小数点算出を使用して、ダイナミックレンジ調整プロセスに対して、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を発生するための手段とを備える。

【0016】

[0016] 別の例では、本開示は、実行されたとき、ビデオデータを処理するように構成されたデバイスの1つまたは複数のプロセッサに、固定小数点算出を使用して、ビデオデータに対してダイナミックレンジ調整プロセスを実行することと、固定小数点算出を使用して、ダイナミックレンジ調整プロセスに対して、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を発生することとを行わせる命令を記憶するコンピュータ可読記憶媒体について説明する。 20

【0017】

[0017] 1つまたは複数の例の詳細が添付の図面および以下の説明に記載される。他の特徴、目的、および利点は、説明、図面、および特許請求の範囲から明らかになろう。

【図面の簡単な説明】

【0018】

【図1】[0018] 本開示の技法を実装するように構成された例示的なビデオ符号化および復号システムを示すブロック図。

【図2】[0019] 色再マッピング情報 (C R I : color remapping information) プロセスの一般的な構造を示す概念図。

【図3】[0020] H D R データの概念を示す概念図。

【図4】[0021] 例示的な色域を示す概念図。

【図5】[0022] H D R / W C G 表現コンバージョン (representation conversion) の一例を示す流れ図。

【図6】[0023] H D R / W C G 逆コンバージョン (inverse conversion) の一例を示す流れ図。 40

【図7】[0024] 知覚的に均一なコードレベルから線形ルミナンスへの (S D R と H D R とを含む) ビデオデータコンバージョンのために利用される電気光学伝達関数 (E O T F : Electro-optical transfer function) の例を示す概念図。

【図8】[0025] 本開示の技法に従って動作する例示的な H D R / W C G コンバージョン装置を示すブロック図。

【図9】[0026] 本開示の技法による例示的な H D R / W C G 逆コンバージョン装置を示すブロック図。

【図10】[0027] 本開示の技法を実装し得るビデオエンコーダの一例を示すブロック図。

。

【図11】[0028] 本開示の技法を実装し得るビデオデコーダの一例を示すブロック図。

【図12】[0029] 本開示の1つの例示的なビデオ処理技法を示すフローチャート。

【図13】[0030] 本開示の別の例示的なビデオ処理技法を示すフローチャート。

【発明を実施するための形態】

【0019】

[0031] 本開示は、高ダイナミックレンジ(HDR)および広色域(WCG)表現をもつビデオデータの処理および/またはコーディングに関する。より詳細には、本開示の技法は、(たとえば、浮動小数点処理動作(floating point processing operation)とは対照的に)固定小数点処理動作(fixed point processing operation)を使用して、ビデオデータコンポーネント(video data component)のレンジ調整を実行するための技法を含む。本明細書で説明される技法およびデバイスは、HDRおよびWCGビデオデータを含む、ビデオデータをコーディングするために利用されるハイブリッドベースのビデオコーディングシステム(たとえば、H.265/HEVC、H.264/AVCなど)の圧縮効率を改善し得る。

10

【0020】

[0032] ハイブリッドベースのビデオコーディング規格を含むビデオコーディング規格は、ITU-T H.261、ISO/IEC MPEG-1 Visual、ITU-T H.262またはISO/IEC MPEG-2 Visual、ITU-T H.263、ISO/IEC MPEG-4 Visual、およびそのスケーラブルビデオコーディング(SVC:Scalable Video Coding)拡張とマルチビュービデオコーディング(MVC:Multiview Video Coding)拡張とを含む、(ISO/IEC MPEG-4 AVCとしても知られる)ITU-T H.264を含む。新しいビデオコーディング規格、すなわち、高効率ビデオコーディング(H.265とも呼ばれるHEVC)の設計が、ITU-Tビデオコーディングエキスパートグループ(VCEG:Video Coding Experts Group)とISO/IECモーションピクチャエキスパートグループ(MPEG:Motion Picture Experts Group)とのジョイントコラボレーションチームオンビデオコーディング(JCT-VC:Joint Collaboration Team on Video Coding)によって確定された。HEVCワーキングドラフト10(WD10)と呼ばれるHEVCドラフト仕様、Brossら、「High efficiency video coding(HEVC) text specification draft 10(for FDIS & Last Call)」、ITU-T SG16 WP3とISO/IEC JTC1/SC29/WG11とのジョイントコラボレーティブチームオンビデオコーディング(JCT-VC)、第12回会議、ジュネーブ、スイス、2013年1月14~23日、JCTVC-L1003v34は、

20

[http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/12\\_Geneva/wg11/JCTVC-L1003-v34.zip](http://phenix.int-evry.fr/jct/doc_end_user/documents/12_Geneva/wg11/JCTVC-L1003-v34.zip)から入手可能である。確定されたHEVC規格はHEVCバージョン1と呼ばれる。

30

【0021】

[0033] 不具合報告(defect report)、Wangら、「High efficiency video coding(HEVC) Defect Report」、ITU-T SG16 WP3とISO/IEC JTC1/SC29/WG11とのジョイントコラボレーティブチームオンビデオコーディング(JCT-VC)、第14回会議、ウィーン、オーストリア、2013年7月25日~8月2日、JCTVC-N1003v1は、

40

[http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/14\\_Vienna/wg11/JCTVC-N1003-v1.zip](http://phenix.int-evry.fr/jct/doc_end_user/documents/14_Vienna/wg11/JCTVC-N1003-v1.zip)から利用可能である。確定されたHEVC規格ドキュメントは、ITU-T H.265, Series H: Audiovisual and Multimedia Systems, Infrastructure of audiovisual services - Coding of moving video, High efficiency video coding、国際電気通信連合(ITU)の電気通信標準化部門、2013年4月として公開されており、確定されたHEVC規格の別のバージョンは、2014年10月に発行された。H.265/HEVC仕様テキストのコピーは、<http://www.itu.int/rec/T-REC-H.265-201504-1/en>からダウンロードされ得る。

50

## 【0022】

[0034] 図1は、本開示の技法を利用し得る例示的なビデオ符号化および復号システム10を示すブロック図である。図1に示されているように、システム10は、宛先デバイス14によって後で復号されるべき符号化ビデオデータを与えるソースデバイス12を含む。特に、ソースデバイス12は、コンピュータ可読媒体16を介して宛先デバイス14にビデオデータを与える。ソースデバイス12および宛先デバイス14は、デスクトップコンピュータ、ノートブック(すなわち、ラップトップ)コンピュータ、タブレットコンピュータ、セットトップボックス(set-top box)、いわゆる「スマート」フォンなどの電話ハンドセット、いわゆる「スマート」パッド、テレビジョン、カメラ、ディスプレイデバイス、デジタルメディアプレーヤ、ビデオゲームコンソール、ビデオストリーミングデバイス、ブロードキャスト受信機デバイスなどを含む、広範囲にわたるデバイスのいずれかを備え得る。いくつかの場合には、ソースデバイス12および宛先デバイス14は、ワイヤレス通信のために装備され得る。

## 【0023】

[0035] 宛先デバイス14は、コンピュータ可読媒体16を介して復号されるべき符号化ビデオデータを受信し得る。コンピュータ可読媒体16は、ソースデバイス12から宛先デバイス14に符号化ビデオデータを移動することが可能な任意のタイプの媒体またはデバイスを備え得る。一例では、コンピュータ可読媒体16は、ソースデバイス12が、符号化ビデオデータを宛先デバイス14にリアルタイムで直接送信することを可能にするための通信媒体を備え得る。符号化ビデオデータは、ワイヤードまたはワイヤレス通信プロトコルなどの通信規格に従って変調され、宛先デバイス14に送信され得る。通信媒体は、無線周波数(RF)スペクトルまたは1つまたは複数の物理伝送線路など、任意のワイヤレスまたはワイヤード通信媒体を備え得る。通信媒体は、ローカルエリアネットワーク、ワイドエリアネットワーク、またはインターネットなどのグローバルネットワークなど、パケットベースネットワークの一部を形成し得る。通信媒体は、ルータ、スイッチ、基地局、またはソースデバイス12から宛先デバイス14への通信を可能にするために有用であり得る任意の他の機器を含み得る。

## 【0024】

[0036] 他の例では、コンピュータ可読媒体16は、ハードディスク、フラッシュドライブ、コンパクトディスク、デジタルビデオディスク、B1u-ray(登録商標)ディスク、または他のコンピュータ可読媒体などの非一時的記憶媒体を含み得る。いくつかの例では、ネットワークサーバ(図示せず)は、たとえば、ネットワーク送信を介して、ソースデバイス12から符号化ビデオデータを受信し、その符号化ビデオデータを宛先デバイス14に与え得る。同様に、ディスクスタンピング設備など、媒体製造設備のコンピューティングデバイスは、ソースデバイス12から符号化ビデオデータを受信し、その符号化ビデオデータを含んでいるディスクを生成し得る。したがって、コンピュータ可読媒体16は、様々な例において、様々な形態の1つまたは複数のコンピュータ可読媒体を含むことが理解されよう。

## 【0025】

[0037] いくつかの例では、符号化データは、出力インターフェース22からストレージデバイスに出力され得る。同様に、符号化データは、入力インターフェースによってストレージデバイスからアクセスされ得る。ストレージデバイスは、ハードドライブ、B1u-rayディスク、DVD、CD-ROM、フラッシュメモリ、揮発性または不揮発性メモリ、あるいは符号化ビデオデータを記憶するための任意の他の好適なデジタル記憶媒体など、様々な分散されたまたはローカルにアクセスされるデータ記憶媒体のいずれかを含み得る。さらなる一例では、ストレージデバイスは、ソースデバイス12によって発生された符号化ビデオを記憶し得るファイルサーバまたは別の中間ストレージデバイスに対応し得る。宛先デバイス14は、ストリーミングまたはダウンロードを介してストレージデバイスから記憶されたビデオデータにアクセスし得る。ファイルサーバは、符号化ビデオデータを記憶することと、符号化ビデオデータを宛先デバイス14に送信することとが

10

20

30

40

50

可能な任意のタイプのサーバであり得る。例示的なファイルサーバとしては、(たとえば、ウェブサイトのための)ウェブサーバ、FTPサーバ、ネットワーク接続ストレージ(NAS)デバイス、またはローカルディスクドライブがある。宛先デバイス14は、インターネット接続を含む、任意の標準のデータ接続を通して符号化ビデオデータにアクセスし得る。これは、ファイルサーバに記憶された符号化ビデオデータにアクセスするのに好適であるワイヤレスチャネル(たとえば、Wi-Fi(登録商標)接続)、ワイヤード接続(たとえば、DSL、ケーブルモデムなど)、またはその両方の組合せを含み得る。ストレージデバイスからの符号化ビデオデータの送信は、ストリーミング送信、ダウンロード送信、またはそれらの組合せであり得る。

【0026】

10

[0038] 本開示の技法は、必ずしもワイヤレス適用例または設定に限定されるとは限らない。本技法は、オーバージェアテレビジョン放送、ケーブルテレビジョン送信、衛星テレビジョン送信、動的適応ストリーミングオーバーHTTP(DASH)などのインターネットストリーミングビデオ送信、データ記憶媒体上に符号化されたデジタルビデオ、データ記憶媒体に記憶されたデジタルビデオの復号、または他の適用例など、様々なマルチメディア適用例のいずれかをサポートするビデオコーディングに適用され得る。いくつかの例では、システム10は、ビデオストリーミング、ビデオ再生、ビデオブロードキャスティング、および/またはビデオテレフォニーなどの適用例をサポートするために、一方または双方向のビデオ送信をサポートするように構成され得る。

【0027】

20

[0039] 図1の例では、ソースデバイス12は、ビデオソース18と、ビデオプリプロセッサユニット19およびビデオエンコーダ20を含むビデオ符号化ユニット21と、出力インターフェース22とを含む。宛先デバイス14は、入力インターフェース28と、ビデオポストプロセッサユニット31およびビデオデコーダ30を含むビデオ復号ユニット29と、ディスプレイデバイス32とを含む。本開示によれば、ソースデバイス12のビデオプリプロセッサユニット19および/またはビデオエンコーダ20、ならびに宛先デバイス14のビデオポストプロセッサユニット31および/またはビデオデコーダ30は、固定小数点実装を用いてHDRおよびWCGビデオデータのより効率的な圧縮を可能にするためにいくつかの色空間中でビデオデータに適用されるシグナリングおよび関係する動作を含む、本開示の技法を実装するように構成され得る。いくつかの例では、ビデオプリプロセッサユニット19はビデオエンコーダ20とは別個であり得る。他の例では、ビデオプリプロセッサユニット19はビデオエンコーダ20の一部であり得る。同様に、いくつかの例では、ビデオポストプロセッサユニット31はビデオデコーダ30とは別個であり得る。他の例では、ビデオポストプロセッサユニット31はビデオデコーダ30の一部であり得る。他の例では、ソースデバイスおよび宛先デバイスは他のコンポーネントまたは構成を含み得る。たとえば、ソースデバイス12は、外部カメラなど、外部ビデオソース18からビデオデータを受信し得る。同様に、宛先デバイス14は、内蔵ディスプレイデバイスを含むのではなく、外部ディスプレイデバイスとインターフェースし得る。

【0028】

30

[0040] 図1の図示されたシステム10は一例にすぎない。HDRおよびWCGビデオデータを処理するための技法は、任意のデジタルビデオ符号化および/またはビデオ復号デバイスによって実行され得る。さらに、本開示の技法はまた、ビデオプリプロセッサおよび/またはビデオポストプロセッサ(たとえば、ビデオプリプロセッサユニット19およびビデオポストプロセッサユニット31)によって実行され得る。概して、ビデオプリプロセッサは、符号化の前に(たとえば、HEVC符号化の前に)ビデオデータを処理するように構成された任意のデバイスであり得る。概して、ビデオポストプロセッサは、復号の後に(たとえば、HEVC復号の後に)ビデオデータを処理するように構成された任意のデバイスであり得る。ソースデバイス12および宛先デバイス14は、ソースデバイス12が宛先デバイス14への送信のためのコード化ビデオデータを発生するような、コーディングデバイスの例にすぎない。いくつかの例では、デバイス12、14は、デバイ

40

50

ス 1 2、 1 4 の各々が、 ビデオ符号化および復号コンポーネント、 ならびに、 ビデオプリプロセッサおよびビデオポストプロセッサ（たとえば、 それぞれ、 ビデオプリプロセッサユニット 1 9 およびビデオポストプロセッサユニット 3 1）を含むように、 実質的に対称的に動作し得る。 したがって、 システム 1 0 は、 たとえば、 ビデオストリーミング、 ビデオ再生、 ビデオプロードキャスティング、 またはビデオテレフォニーのための、 ビデオデバイス 1 2 とビデオデバイス 1 4 との間の一方向または双方向のビデオ送信をサポートし得る。

### 【 0 0 2 9 】

[0041] ソースデバイス 1 2 のビデオソース 1 8 は、 ビデオカメラなどのビデオキャプチャデバイス、 前にキャプチャされたビデオを含んでいるビデオアーカイブ、 および／またはビデオコンテンツプロバイダからビデオを受信するためのビデオフィードインターフェースを含み得る。 さらなる代替として、 ビデオソース 1 8 は、 ソースビデオとしてのコンピュータグラフィックスベースのデータ、 またはライブビデオとアーカイブビデオとコンピュータ発生ビデオとの組合せを発生し得る。 いくつかの場合には、 ビデオソース 1 8 がビデオカメラである場合、 ソースデバイス 1 2 および宛先デバイス 1 4 は、 いわゆるカメラフォンまたはビデオフォンを形成し得る。 ただし、 上述のように、 本開示で説明される技法は、 概してビデオコーディングおよびビデオ処理に適用可能であり得、 ワイヤレスおよび／またはワイヤード適用例に適用され得る。 各場合において、 キャプチャされたビデオ、 プリキャプチャされたビデオ、 またはコンピュータ発生ビデオは、 ビデオ符号化ユニット 2 1 によって符号化され得る。 符号化されたビデオ情報は、 次いで、 出力インターフェース 2 2 によってコンピュータ可読媒体 1 6 上に出力され得る。

### 【 0 0 3 0 】

[0042] 宛先デバイス 1 4 の入力インターフェース 2 8 はコンピュータ可読媒体 1 6 から情報を受信する。 コンピュータ可読媒体 1 6 の情報は、 ビデオエンコーダ 2 0 によって定義され、 またビデオ復号ユニット 2 9 によって使用される、 ブロックおよび他のコード化ユニット、 たとえば、 ピクチャグループ（ G O P ）の特性および／または処理を記述するシンタックス要素を含む、 シンタックス情報を含み得る。 ディスプレイデバイス 3 2 は、 復号ビデオデータをユーザに対して表示し、 陰極線管（ C R T ）、 液晶ディスプレイ（ L C D ）、 プラズマディスプレイ、 有機発光ダイオード（ O L E D ）ディスプレイ、 または別のタイプのディスプレイデバイスなど、 様々なディスプレイデバイスのいずれかを備え得る。

### 【 0 0 3 1 】

[0043] 図示のように、 ビデオプリプロセッサユニット 1 9 は、 ビデオソース 1 8 からビデオデータを受信する。 ビデオプリプロセッサユニット 1 9 は、 ビデオデータを、 ビデオエンコーダ 2 0 を用いて符号化するのに好適である形態にコンバートするために、 ビデオデータを処理するように構成され得る。 たとえば、 ビデオプリプロセッサユニット 1 9 は、 （たとえば、 非線形伝達関数を使用する）ダイナミックレンジ圧縮、 より圧縮したまたはロバストな色空間への色コンバージョン、 および／あるいは浮動-整数表現コンバージョンを実行し得る。 ビデオエンコーダ 2 0 は、 ビデオプリプロセッサユニット 1 9 によって出力されたビデオデータに対してビデオ符号化を実行し得る。 ビデオデコーダ 3 0 は、 ビデオデータを復号するためにビデオエンコーダ 2 0 の逆を実行し得、 ビデオポストプロセッサユニット 3 1 は、 ビデオデータを表示に好適な形態にコンバートするために、 ビデオプリプロセッサユニット 1 9 の逆を実行し得る。 たとえば、 ビデオポストプロセッサユニット 3 1 は、 表示に好適なビデオデータを発生するために、 整数-浮動コンバージョン、 圧縮したまたはロバストな色空間からの色コンバージョン、 および／またはダイナミックレンジ圧縮の逆を実行し得る。

### 【 0 0 3 2 】

[0044] ビデオエンコーダ 2 0 およびビデオデコーダ 3 0 はそれぞれ、 1 つまたは複数のマイクロプロセッサ、 デジタル信号プロセッサ（ D S P ）、 特定用途向け集積回路（ A S I C ）、 フィールドプログラマブルゲートアレイ（ F P G A ）、 ディスクリート論理、

10

20

30

40

50

ソフトウェア、ハードウェア、ファームウェアなど、様々な好適なエンコーダ回路のいずれか、またはそれらの任意の組合せとして実装され得る。本技法が部分的にソフトウェアで実装されるとき、デバイスは、好適な非一時的コンピュータ可読媒体にソフトウェアのための命令を記憶し、本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その命令をハードウェアで実行し得る。ビデオエンコーダ20およびビデオデコーダ30の各々は1つまたは複数のエンコーダまたはデコーダ中に含まれ得、そのいずれも、それぞれのデバイス内で複合エンコーダ/デコーダ(コーデック)の一部として統合され得る。

### 【0033】

[0045] ビデオプリプロセッサユニット19およびビデオポストプロセッサユニット31はそれぞれ、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(DSP)、特定用途向け集積回路(ASIC)、フィールドプログラマブルゲートアレイ(FPGA)、ディスクリート論理、ソフトウェア、ハードウェア、ファームウェアなど、様々な好適なエンコーダ回路のいずれか、またはそれらの任意の組合せとして実装され得る。本技法が部分的にソフトウェアで実装されるとき、デバイスは、好適な非一時的コンピュータ可読媒体にソフトウェアのための命令を記憶し、本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その命令をハードウェアで実行し得る。上記で説明されたように、ビデオプリプロセッサユニット19およびビデオポストプロセッサユニット31は、それぞれ、ビデオエンコーダ20およびビデオデコーダ30とは別個のデバイスである。他の例では、ビデオプリプロセッサユニット19は単一のデバイス内でビデオエンコーダ20と統合し得、逆ビデオポストプロセッサユニット31は単一のデバイス内でビデオデコーダ30と統合され得る。

### 【0034】

[0046] いくつかの例では、ビデオエンコーダ20およびビデオデコーダ30は、ISO/IEC MPEG-4 Visual、およびそのスケーラブルビデオコーディング(SVC)拡張と、マルチビュービデオコーディング(MVC)拡張と、MVCベースの3次元ビデオ(3DV)拡張とを含む、(ISO/IEC MPEG-4 AVCとしても知られる)ITU-T H.264などのビデオ圧縮規格に従って動作する。いくつかの事例では、MVCベースの3DVに準拠する任意のビットストリームは、MVCプロファイル、たとえば、ステレオハイプロファイルに準拠するサブビットストリームを常に含んでいる。さらに、H.264/AVCへの3DVコーディング拡張、すなわちAVCベース3DVを発生するための取り組みが進行中である。ビデオコーディング規格の他の例としては、ITU-T H.261、ISO/IEC MPEG-1 Visual、ITU-T H.262またはISO/IEC MPEG-2 Visual、ITU-T H.263、ISO/IEC MPEG-4 Visual、およびITU-T H.264、ISO/IEC Visualがある。他の例では、ビデオエンコーダ20およびビデオデコーダ30は、HEVC規格に従って動作するように構成され得る。

### 【0035】

[0047] HEVCおよび他のビデオコーディング規格では、ビデオシーケンスは一般に一連のピクチャを含む。ピクチャは「フレーム」と呼ばれることがある。ピクチャは、S<sub>L</sub>、S<sub>Cb</sub>、およびS<sub>Cr</sub>と示される3つのサンプルアレイを含み得る。S<sub>L</sub>はルーマサンプルの2次元アレイ(すなわち、ブロック)である。S<sub>Cb</sub>はCbクロミナンスサンプルの2次元アレイである。S<sub>Cr</sub>はCrクロミナンスサンプルの2次元アレイである。クロミナンスサンプルは、本明細書では「クロマ」サンプルと呼ばれることもある。他の事例では、ピクチャはモノクロームであり得、ルーマサンプルのアレイのみを含み得る。

### 【0036】

[0048] ビデオエンコーダ20は、コーディングツリーユニット(CTU: coding tree unit)のセットを発生し得る。CTUの各々は、ルーマサンプルのコーディングツリーブロックと、クロマサンプルの2つの対応するコーディングツリーブロックと、それらのコーディングツリーブロックのサンプルをコーディングするために使用されるシンタック

10

20

30

40

50

ス構造とを備え得る。モノクロームピクチャまたは3つの別個の色平面を有するピクチャでは、CTUは、単一のコーディングツリーブロックと、そのコーディングツリーブロックのサンプルをコーディングするために使用されるシンタックス構造(syntax structure)とを備え得る。コーディングツリーブロックはサンプルのN×Nブロックであり得る。CTUは「ツリーブロック」または「最大コーディングユニット」(LCU: largest coding unit)と呼ばれることもある。HEVCのCTUは、H.264/AVCなど、他のビデオコーディング規格のマクロブロックに広い意味で類似し得る。しかしながら、CTUは、必ずしも特定のサイズに限定されることは限らず、1つまたは複数のコーディングユニット(CU)を含み得る。スライスは、ラスタ走査において連続的に順序付けられた整数個のCTUを含み得る。

10

### 【0037】

[0049] 本開示は、サンプルの1つまたは複数のブロックと、サンプルの1つまたは複数のブロックのサンプルをコーディングするために使用されるシンタックス構造とを指すために、「ビデオユニット(video unit)」または「ビデオブロック(video block)」という用語を使用し得る。例示的なタイプのビデオユニットは、HEVCにおけるCTU、CU、PU、変換ユニット(TU)、または他のビデオコーディング規格におけるマクロブロック、マクロブロックパーティションなどを含み得る。

### 【0038】

[0050] コード化CTUを発生するために、ビデオエンコーダ20は、コーディングツリーブロックをコーディングブロックに分割するためにCTUのコーディングツリーブロックに対して4分木区分を再帰的に実行し得、したがって「コーディングツリーユニット(coding tree unit)」という名称がある。コーディングブロックはサンプルのN×Nブロックである。CUは、ルーマサンプルアレイとCbサンプルアレイとCrサンプルアレイとを有するピクチャのルーマサンプルのコーディングブロックと、そのピクチャのクロマサンプルの2つの対応するコーディングブロックと、それらのコーディングブロックのサンプルをコーディングするために使用されるシンタックス構造とを備え得る。モノクロームピクチャまたは3つの別個の色平面を有するピクチャでは、CUは、単一のコーディングブロックと、そのコーディングブロックのサンプルをコーディングするために使用されるシンタックス構造とを備え得る。

20

### 【0039】

[0051] ビデオエンコーダ20は、CUのコーディングブロックを1つまたは複数の予測ブロックに区分し得る。予測ブロックは、同じ予測が適用されるサンプルの矩形(すなわち、正方形または非正方形)ブロックであり得る。CUの予測ユニット(PU)は、ピクチャのルーマサンプルの予測ブロックと、クロマサンプルの2つの対応する予測ブロックと、予測ブロックサンプルを予測するために使用されるシンタックス構造とを備え得る。モノクロームピクチャまたは3つの別個の色平面を有するピクチャでは、PUは、単一の予測ブロックと、その予測ブロックサンプルを予測するために使用されるシンタックス構造とを備え得る。ビデオエンコーダ20は、CUの各PUのルーマ予測ブロック、Cb予測ブロック、およびCr予測ブロックのための予測ルーマブロック、予測Cbブロック、および予測Crブロックを発生し得る。

30

### 【0040】

[0052] ビデオエンコーダ20は、PUのための予測ブロックを発生するためにイントラ予測またはインター予測を使用し得る。ビデオエンコーダ20がPUの予測ブロックを発生するためにイントラ予測を使用する場合、ビデオエンコーダ20は、PUに関連するピクチャの復号サンプルに基づいてPUの予測ブロックを発生し得る。

40

### 【0041】

[0053] ビデオエンコーダ20が、PUの予測ブロックを発生するためにインター予測を使用する場合、ビデオエンコーダ20は、PUに関連するピクチャ以外の1つまたは複数のピクチャの復号サンプルに基づいて、PUの予測ブロックを発生し得る。インター予測は、単方向インター予測(すなわち、単予測)または双方向インター予測(すなわち、

50

双予測)であり得る。単予測または双予測を実行するために、ビデオエンコーダ20は、現在スライスに対して、第1の参照ピクチャリスト(RefPicList0)と第2の参照ピクチャリスト(RefPicList1)とを発生し得る。

【0042】

[0054] 参照ピクチャリストの各々は1つまたは複数の参照ピクチャを含み得る。単予測を使用するとき、ビデオエンコーダ20は、参照ピクチャ内の参照ロケーションを決定するために、RefPicList0とRefPicList1のいずれかまたは両方の中の参照ピクチャを探索し得る。さらに、単予測を使用するとき、ビデオエンコーダ20は、参照ロケーションに対応するサンプルに少なくとも部分的に基づいて、PUのための予測サンプルブロックを発生し得る。さらに、単予測を使用するとき、ビデオエンコーダ20は、PUの予測ブロックと参照ロケーションとの間の空間変位を示す单一の動きベクトルを発生し得る。PUの予測ブロックと参照ロケーションとの間の空間変位を示すために、動きベクトルは、PUの予測ブロックと参照ロケーションとの間の水平変位を指定する水平コンポーネントを含み得、PUの予測ブロックと参照ロケーションとの間の垂直変位を指定する垂直コンポーネントを含み得る。

【0043】

[0055] PUを符号化するために双予測を使用するとき、ビデオエンコーダ20は、RefPicList0中の参照ピクチャ中の第1の参照ロケーションと、RefPicList1中の参照ピクチャ中の第2の参照ロケーションとを決定し得る。ビデオエンコーダ20は、次いで、第1および第2の参照ロケーションに対応するサンプルに少なくとも部分的に基づいて、PUのための予測ブロックを発生し得る。さらに、PUを符号化するために双予測を使用するとき、ビデオエンコーダ20は、PUのサンプルブロックと第1の参照ロケーションとの間の空間変位を示す第1の動きと、PUの予測ブロックと第2の参照ロケーションとの間の空間変位を示す第2の動きとを発生し得る。

【0044】

[0056] ビデオエンコーダ20がCUの1つまたは複数のPUのための予測ルーマブロック、予測Cbブロック、および予測Crブロックを発生した後、ビデオエンコーダ20は、CUのためのルーマ残差ブロックを発生し得る。CUのルーマ残差ブロック中の各サンプルは、CUの予測ルーマブロックのうちの1つ中のルーマサンプルとCUの元のルーマコーディングブロック中の対応するサンプルとの間の差分を示す。さらに、ビデオエンコーダ20はCUのためのCb残差ブロックを発生し得る。CUのCb残差ブロック中の各サンプルは、CUの予測Cbブロックのうちの1つ中のCbサンプルとCUの元のCbコーディングブロック中の対応するサンプルとの間の差分を示し得る。ビデオエンコーダ20はCUのためのCr残差ブロックをも発生し得る。CUのCr残差ブロック中の各サンプルは、CUの予測Crブロックのうちの1つ中のCrサンプルとCUの元のCrコーディングブロック中の対応するサンプルとの間の差分を示し得る。

【0045】

[0057] さらに、ビデオエンコーダ20は、CUのルーマ残差ブロックと、Cb残差ブロックと、Cr残差ブロックとを、1つまたは複数のルーマ変換ブロックと、Cb変換ブロックと、Cr変換ブロックとに分解するために4分木区分を使用し得る。変換ブロックは、同じ変換が適用されるサンプルの矩形ブロックであり得る。CUの変換ユニット(TU)は、ルーマサンプルの変換ブロックと、クロマサンプルの2つの対応する変換ブロックと、変換ブロックサンプルを変換するために使用されるシンタックス構造とを備え得る。モノクロームピクチャまたは3つの別個の色平面を有するピクチャでは、TUは、単一の変換ブロックと、変換ブロックサンプルを変換するために使用されるシンタックス構造とを備え得る。したがって、CUの各TUは、ルーマ変換ブロック、Cb変換ブロック、およびCr変換ブロックに関連し得る。TUに関連するルーマ変換ブロックはCUのルーマ残差ブロックのサブブロックであり得る。Cb変換ブロックはCUのCb残差ブロックのサブブロックであり得る。Cr変換ブロックはCUのCr残差ブロックのサブブロックであり得る。

10

20

30

40

50

## 【0046】

[0058] ビデオエンコーダ20は、TUのためのルーマ係数ブロックを発生するために、TUのルーマ変換ブロックに1つまたは複数の変換を適用し得る。係数ブロックは変換係数の2次元アレイであり得る。変換係数はスカラー量であり得る。ビデオエンコーダ20は、TUのためのCb係数ブロックを発生するために、TUのCb変換ブロックに1つまたは複数の変換を適用し得る。ビデオエンコーダ20は、TUのためのCr係数ブロックを発生するために、TUのCr変換ブロックに1つまたは複数の変換を適用し得る。

## 【0047】

[0059] 係数ブロック(たとえば、ルーマ係数ブロック、Cb係数ブロックまたはCr係数ブロック)を発生した後に、ビデオエンコーダ20は、係数ブロックを量子化し得る。量子化は、概して、変換係数を表すために使用されるデータの量をできるだけ低減するために変換係数が量子化され、さらなる圧縮を行うプロセスを指す。さらに、ビデオエンコーダ20は、変換係数を逆量子化し、ピクチャのCUのTUの変換ブロックを再構成するために変換係数に逆変換を適用し得る。ビデオエンコーダ20は、CUのコーディングブロックを再構成するためにCUのTUの再構成された変換ブロックとCUのPUの予測ブロックとを使用し得る。ピクチャの各CUのコーディングブロックを再構成することによって、ビデオエンコーダ20はピクチャを再構成し得る。ビデオエンコーダ20は、再構成されたピクチャを復号ピクチャバッファ(decoded picture buffer)に記憶し得る。ビデオエンコーダ20は、インター予測およびイントラ予測のためにDPB中の再構成されたピクチャを使用し得る。

10

20

## 【0048】

[0060] ビデオエンコーダ20が係数ブロックを量子化した後に、ビデオエンコーダ20は、量子化された変換係数を示すシンタックス要素をエントロピー符号化し得る。たとえば、ビデオエンコーダ20は、量子化された変換係数を示すシンタックス要素に対してコンテキスト適応型バイナリ算術コーディング(Context-Adaptive Binary Arithmetic Coding)を実行し得る。ビデオエンコーダ20は、エントロピー符号化シンタックス要素をビットストリーム中に出力し得る。

## 【0049】

[0061] ビデオエンコーダ20は、コード化ピクチャと関連データとの表現を形成するビットのシーケンスを含むビットストリームを出力し得る。ビットストリームは、ネットワークアブストラクションレイヤ(NAL:network abstraction layer)ユニットのシーケンスを備え得る。NALユニットの各々はNALユニットヘッダを含み、ローバイトシーケンスペイロード(RBSP:raw byte sequence payload)をカプセル化する。NALユニットヘッダは、NALユニットタイプコードを示すシンタックス要素を含み得る。NALユニットのNALユニットヘッダによって指定されるNALユニットタイプコードは、NALユニットのタイプを示す。RBSPは、NALユニット内にカプセル化された整数個のバイトを含んでいるシンタックス構造であり得る。いくつかの事例では、RBSPは0ビットを含む。

30

## 【0050】

[0062] 異なるタイプのNALユニットは、異なるタイプのRBSPをカプセル化し得る。たとえば、第1のタイプのNALユニットはピクチャパラメータセット(PPS:picture parameter set)のためのRBSPをカプセル化し得、第2のタイプのNALユニットはコード化スライスのためのRBSPをカプセル化し得、第3のタイプのNALユニットは補足エンハンスメント情報(SEI:Supplemental Enhancement Information)のためのRBSPをカプセル化し得、以下同様である。PPSは、0個またはそれ以上のコード化ピクチャ全体に適用されるシンタックス要素を含んでいることがあるシンタックス構造である。(パラメータセットおよびSEIメッセージのためのRBSPとは対照的に)ビデオコーディングデータのためのRBSPをカプセル化するNALユニットは、ビデオコーディングレイヤ(VCL:video coding layer)NALユニットと呼ばれることがある。コード化スライスをカプセル化するNALユニットは、本明細書ではコード化スラ

40

50

イス N A L ユニットと呼ばれることがある。コード化スライスのための R B S P は、スライスヘッダとスライスデータとを含み得る。

#### 【 0 0 5 1 】

[0063] ビデオデコーダ 3 0 はビットストリームを受信し得る。さらに、ビデオデコーダ 3 0 は、ビットストリームからシンタックス要素を復号するために、ビットストリームをパースし得る。ビデオデコーダ 3 0 は、ビットストリームから復号されたシンタックス要素に少なくとも部分的に基づいてビデオデータのピクチャを再構成し得る。ビデオデータを再構成するためのプロセスは、概して、ビデオエンコーダ 2 0 によって実行されるプロセスに相反し得る。たとえば、ビデオデコーダ 3 0 は、現在 C U の P U のための予測ブロックを決定するために P U の動きベクトルを使用し得る。ビデオデコーダ 3 0 は、P U のための予測ブロックを発生するために P U の 1 つまたは複数の動きベクトルを使用し得る。10

#### 【 0 0 5 2 】

[0064] さらに、ビデオデコーダ 3 0 は、現在 C U の T U に関する係数ブロックを逆量子化し得る。ビデオデコーダ 3 0 は、現在 C U の T U に関する変換ブロックを再構成するために、係数ブロックに対して逆変換を実行し得る。ビデオデコーダ 3 0 は、現在 C U の P U のための予測サンプルブロックのサンプルを現在 C U の T U の変換ブロックの対応するサンプルに加算することによって、現在 C U のコーディングブロックを再構成し得る。ピクチャの各 C U のためのコーディングブロックを再構成することによって、ビデオデコーダ 3 0 はピクチャを再構成し得る。ビデオデコーダ 3 0 は、出力のためにおよび/または他のピクチャを復号する際に使用するために、復号ピクチャを復号ピクチャバッファに記憶し得る。20

#### 【 0 0 5 3 】

[0065] 補足エンハンスメント情報 (S E I) メッセージ (supplemental enhancement information (SEI) message) は、一般に、デコーダ (たとえば、ビデオデコーダ 3 0) によってビットストリームを復号するために必須でない情報を搬送するために、しばしばビデオビットストリーム中に含まれる。S E I メッセージ中に含まれている情報は、復号された出力の表示または処理を改善する際に有用であり得、たとえばそのような情報は、コンテンツの視認性を改善するためにデコーダ側エンティティによって使用され得る。適用規格に準拠するすべてのデバイスに品質の改善がもたらされ得るように、いくつかの適用規格が、ビットストリーム中のそのような S E I メッセージの存在 (たとえば、E T S I - T S 1 0 1 5 4 7 - 2、Digital Video Broadcasting (DVB) Plano-stereoscopic 3DTV; Part 2: Frame compatible plano-stereoscopic 3DTV) で説明される、S E I メッセージがビデオのあらゆるフレームのために搬送される、フレーム互換平面立体視 3 D T V ビデオフォーマット (frame-compatible plano-stereoscopic 3DTV video format) のためのフレームパッキング S E I メッセージの搬送、たとえば、3 G P P (登録商標)30

T S 2 6 . 1 1 4 v 1 3 . 0 . 0 、3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; IP Multimedia Subsystem (IMS); Multimedia Telephony; Media handling and interaction (Release 13) で説明される、リカバリポイント S E I メッセージの処理、または、たとえば、E T S I - T S 1 0 1 1 5 4 、Digital Video Broadcasting (DVB); Specification for the use of Video and Audio Coding in Broadcasting Applications based on the MPEG-2 Transport Stream で説明される、D V B におけるパンスキヤン走査矩形 S E I メッセージ (pan-scan rectangle SEI message) の使用) を規定することがあることも可能である。40

#### 【 0 0 5 4 】

[0066] トーンマッピング情報 S E I メッセージは、ルーマサンプルまたは R G B コンポーネントサンプルの各々をマッピングするために使用される。t o n e \_ m a p \_ i d の異なる値が異なる目的を定義するために使用され、それに応じて、トーンマップ S E I メッセージのシンタックスも修正される。t o n e \_ m a p \_ i d のための 1 の値は、S E I メッセージが、R G B サンプルを最小値 (minimum value) および最大値 (maximum v50

alue) にクリッピングする (clip) ことを可能にする。tone\_map\_id のための 3 の値は、ピボット点の形態のルックアップテーブルのシグナリングを可能にする。しかしながら、適用されたとき、同じ値が、すべての RGB コンポーネントに適用されるか、またはルーマコンポーネントのみに適用される。

#### 【0055】

[0067] ニー関数 (knee function) SEI メッセージは、正規化された線形領域中の復号ピクチャの RGB コンポーネントのマッピングを示すために使用される。入力および出力最大ルミナンス値も示され、ルックアップテーブルが、入力ルミナンス値を出力ルミナンス値にマッピングする。同じルックアップテーブルが、すべての 3 つの色コンポーネント (color component) に適用される。

10

#### 【0056】

[0068] H E V C 規格において定義された色再マッピング情報 (C R I) SEI メッセージは、1 つの色空間中のピクチャを別の色空間にマッピングするために使用される情報を伝達するために使用される。一例では、C R I SEI メッセージのシンタックスは、3 つの部分、すなわち、第 1 のルックアップテーブル (プリ L U T) と、その後に続く色再マッピング係数を示す  $3 \times 3$  行列と、その後に続く第 2 のルックアップテーブル (ポスト L U T) とを含む。各色コンポーネント、たとえば、R、G、B または Y、C b、C r について、独立した L U T が、プリ L U T とポスト L U T の両方のために定義される。C R I SEI メッセージはまた、colour\_remap\_id と呼ばれるシンタックス要素を含み、その異なる値が、SEI メッセージの異なる目的を示すために使用され得る。図 2 は、C R I SEI メッセージによって指定された色再マッピング情報プロセスの一般的な構造を示す。

20

#### 【0057】

[0069] ダイナミックレンジ調整 (D R A) SEI メッセージ。たとえば、D. Bugdayci Sansli, A. K. Ramasubramonian, D. Rusanovs kyy, S. Lee, J. Sole, M. Karczewicz, Dynamic range adjustment SEI message, m 3 6 3 3 0, M P E G 会議、ワルシャワ、ポーランド、2015 年 6 月 22 ~ 26 日で説明されるように、ダイナミックレンジ調整 SEI メッセージは、ビデオコーディング規格の一部として採用されなかったが、SEI メッセージは、入力サンプルをマッピングするために、スケールおよびオフセット数の 1 つのセットのシグナリングを含む。SEI メッセージはまた、異なるコンポーネントのための異なるルックアップテーブルのシグナリングを可能にし、また、同じスケールおよびオフセットが 2 つ以上のコンポーネントのために使用されるべきであるとき、シグナリング最適化を可能にする。スケールおよびオフセット数は、固定長正確さでシグナリングされる。

30

#### 【0058】

[0070] 次世代ビデオ適用例は、H D R および W C G をもつキャプチャされた風景を表すビデオデータを用いて動作することが予期される。利用されるダイナミックレンジおよび色域のパラメータは、ビデオコンテンツの 2 つの独立した属性であり、デジタルテレビジョンおよびマルチメディアサービスの目的のためのそれらの仕様は、いくつかの国際規格によって定義される。たとえば、I T U - R R e c . B T . 7 0 9、「Parameter values for the HDTV standards for production and international programme exchange」、および I T U - R R e c . B T . 2 0 2 0、「Parameter values for ultra-high definition television systems for production and international programme exchange」は、それぞれ、標準色域を越えて拡張する標準ダイナミックレンジ (S D R) および色プライマリなど、H D T V (高精細度テレビジョン) および U H D T V (超高精細度テレビジョン) のためのパラメータを定義する。R e c . B T . 2 1 0 0、「Image parameter values for high dynamic range television for use in production and international programme exchange」は、伝達関数と、広色域表現をサポートするプライマリを含む、H D R テレビジョン使用のための表現とを定義する。他のシステムにおいてダイナミックレンジおよび色域属性を指定する他の規格開発機関 (S D O : standards developing

40

50

organization) 文献もあり、たとえば、DCI-P3色域は SMPTE-231-2 (米国映画テレビ技術者協会 (Society of Motion Picture and Television Engineers) )において定義され、HDRのいくつかのパラメータは SMPTE-2084において定義される。ビデオデータのためのダイナミックレンジおよび色域の簡単な説明が以下で与えられる。

#### 【0059】

[0071] ダイナミックレンジは、一般に、ビデオ信号の最大輝度 (たとえば、ルミナンス) と最小輝度との間の比として定義される。ダイナミックレンジは、「f ストップ」に関して測定され得、ここで、1つのf ストップは、信号のダイナミックレンジの2倍に対応する。MPEGの定義では、16個よりも多いf ストップをもつ輝度変動を起用するコンテンツは、HDRコンテンツと呼ばれる。いくつかの項では、10個のf ストップと16個のf ストップとの間のレベルは、中間ダイナミックレンジとして見なされるが、それは他の定義ではHDRと見なされる。本開示のいくつかの例では、HDRビデオコンテンツは、標準ダイナミックレンジ (たとえば、ITU-R Rec. BT.709によって指定されたビデオコンテンツ) をもつ旧来使用されるビデオコンテンツよりも高いダイナミックレンジを有する任意のビデオコンテンツであり得る。

10

#### 【0060】

[0072] 人間の視覚系 (HVS : human visual system) は、SDRコンテンツおよびHDRコンテンツよりもはるかに大きいダイナミックレンジを知覚することが可能である。しかしながら、HVSは、HVSのダイナミックレンジをいわゆる同時レンジに狭くするための適応機構を含む。同時レンジの幅は、現在の照明状況 (たとえば、現在の輝度) に依存し得る。厳密なレンジは各個人およびディスプレイに基づいて変動し得るが、HDTVのSDR、UHDTVの予想されるHDRおよびHVSダイナミックレンジによって与えられるダイナミックレンジの可視化が図3に示されている。

20

#### 【0061】

[0073] 現在のビデオ適用例およびサービスは、ITU Rec. 709によって規制され、約0.1~100 (しばしば「ニト (nit)」と呼ばれる) カンデラ (candela) (cd) 每m<sup>2</sup>の輝度 (たとえば、ルミナンス) のレンジを一般にサポートするSDRを与え、10個未満のf ストップをもたらす。いくつかの例示的な次世代ビデオサービスは、最高16個のf ストップのダイナミックレンジを与えることが予想される。そのようなコンテンツのための詳細な仕様は現在開発中であるが、いくつかの初期パラメータは、SMPTE-2084およびITU-R Rec. 2020において指定されている。

30

#### 【0062】

[0074] HDR以外のより現実的なビデオエクスペリエンスのための別の態様は色次元である。色次元は一般に色域によって定義される。図4は、SDR色域 (BT.709色プライマリに基づく三角形100)、およびUHDTVのためのそのより広い色域 (BT.2020色プライマリに基づく三角形102) を示す概念図である。図4はまた、自然色の範囲を表す、(舌形のエリア104によって画定された) いわゆるスペクトル軌跡を示す。図3によって示されているように、BT.709 (三角形100) 色プライマリからBT.2020 (三角形102) 色プライマリに移動することは、約70%より多くの色をもつUHDTVサービスを与えることを目的とする。D65は、BT.709および/またはBT.2020仕様のための例示的な白色を指定する。

40

#### 【0063】

[0075] DCI-P3色空間、BT.709色空間、およびBT.2020色空間のための色域仕様の例が表1に示されている。

#### 【0064】

## 【表1】

表1—色域パラメータ

色空間	RGB色空間パラメータ							
	白色点		原色					
	$xx_w$	$yy_w$	$xx_r$	$yy_r$	$xx_g$	$yy_g$	$xx_b$	$yy_b$
DCI-P3	0.314	0.351	0.680	0.320	0.265	0.690	0.150	0.060
ITU-R BT.709	0.3127	0.3290	0.64	0.33	0.30	0.60	0.15	0.06
ITU-R BT.2020	0.3127	0.3290	0.708	0.292	0.170	0.797	0.131	0.046

## 【0065】

[0076] 表1に見られるように、色域は、白色点のX値およびY値によって、ならびに原色（たとえば、赤（R）、緑（G）、および青（B））のx値およびy値によって定義され得る。x値およびy値は、CIE1931色空間によって定義されるように、色の色度（XおよびZ）および輝度（Y）から導出された正規化された値を表す。CIE1931色空間は、（たとえば、波長に関して）純粋な色と人間の眼がどのようにそのような色を知覚するかとの間のリンクを定義する。

## 【0066】

[0077] HDR/WCGビデオデータは、一般に、4:4:4クロマフォーマットおよび極めて広い色空間（たとえば、CIE XYZ）を用いて、コンポーネントごとに極めて高い精度で（浮動小数点さえ）獲得され、記憶される。この表現は高精度をターゲットにし、ほとんど数学的に可逆である。しかしながら、HDR/WCGビデオデータを記憶するためのそのようなフォーマットは、多くの冗長性を含み得、圧縮目的のために最適でないことがある。HVSベースの仮定を用いたより低い精度のフォーマットが、一般に、最先端のビデオ適用例のために利用される。

## 【0067】

[0078] 圧縮の目的のためのビデオデータフォーマットコンバージョンプロセスの一例は、図5に示されているように、3つの主要なプロセスを含む。図5の技法は、ソースデバイス12によって実行され得る。線形RGBデータ110は、HDR/WCGビデオデータであり得、浮動小数点表現で記憶され得る。線形RGBデータ110は、ダイナミックレンジ圧縮のために非線形伝達関数（TF）112を使用して圧縮され得る。伝達関数112は、任意の数の非線形伝達関数、たとえば、SMPTE-2084において定義されているPQ TFを使用して、線形RGBデータ110を圧縮し得る。いくつかの例では、色コンバージョンプロセス114は、ハイブリッドビデオエンコーダによる圧縮により好適であるより圧縮したまたはロバストな色空間（たとえば、YUVまたはYCrCb色空間）に、圧縮されたデータをコンバートする。このデータは、次いで、コンバートされたHDR'データ118を生成するために、浮動-整数表現量子化ユニット116を使用して量子化される。この例では、HDR'データ118は整数表現にある。HDR'データは、現在、ハイブリッドビデオエンコーダ（たとえば、HEVC技法を適用するビデオエンコーダ20）による圧縮により好適なフォーマットにある。図5に示されているプロセスの順序は、一例として与えられ、他の適用例において異なり得る。たとえば、色コンバージョンはTFプロセスに先行し得る。さらに、追加の処理、たとえば、空間サブサンプリングが色コンポーネントに適用され得る。

## 【0068】

[0079] デコーダ側における逆コンバージョンは図6に示されている。図6の技法は、

10

20

30

40

50

宛先デバイス 14 によって実行され得る。コンバートされた H D R ' データ 120 は、ハイブリッドビデオデコーダ（たとえば、H E V C 技法を適用するビデオデコーダ 30 ）を使用してビデオデータを復号することを通して、宛先デバイス 14 において取得され得る。H D R ' データ 120 は、次いで、逆量子化ユニット 122 によって逆量子化され得る。次いで、逆色コンバージョンプロセス 124 が、逆量子化された H D R ' データに適用され得る。逆色コンバージョンプロセス 124 は、色コンバージョンプロセス 114 の逆であり得る。たとえば、逆色コンバージョンプロセス 124 は、H D R ' データを Y C r C b フォーマットから R G B フォーマットにコンバートし得る。次に、逆伝達関数 126 が、線形 R G B データ 128 を再作成するために、伝達関数 112 によって圧縮されたダイナミックレンジを再加算するために、データに適用され得る。

10

#### 【 0069 】

[0080] 図 5 に示されている技法は、次に、より詳細に説明される。概して、伝達関数は、量子化による誤差がルミナンス値のレンジに（ほぼ）わたって知覚的に均一になるように、データのダイナミックレンジを圧縮するためにデータ（たとえば、H D R / W C G ビデオデータ）に適用される。そのような圧縮は、より少ないビットを用いてデータが表されることを可能にする。一例では、伝達関数は、1 次元（1 D）非線形関数であり得、たとえば、R e c . 709 における S D R のために指定されるように、エンドユーザ表示の電気光学伝達関数（E O T F）の逆を反映し得る。別の例では、伝達関数は、H V S 知覚対輝度変化、たとえば、H D R のための S M P T E - 2084 において指定された P Q 伝達関数を近似し得る。O E T F の逆プロセスは、コードレベルをルミナンスにマッピングする、E O T F（電気光学伝達関数）である。図 7 は、E O T F として使用される非線形伝達関数のいくつかの例を示す。伝達関数はまた、各 R、G および B コンポーネントに別個に適用され得る。

20

#### 【 0070 】

[0081] 本開示のコンテキストでは、「信号値」または「色値」という用語は、画像要素のための（R、G、B、またはYなどの）特定の色コンポーネントの値に対応するルミナンスレベルについて説明するために使用され得る。信号値は、一般に、線形光源レベル（ルミナンス値）を表す。「コードレベル」または「デジタルコード値」という用語は、画像信号値のデジタル表現を指し得る。一般に、そのようなデジタル表現は非線形信号値を表す。E O T F は、ディスプレイデバイス（たとえば、ディスプレイデバイス 32 ）に与えられた非線形信号値とディスプレイデバイスによって生成された線形色値との間の関係を表す。

30

#### 【 0071 】

[0082] R G B が、一般に画像キャプチャセンサーによって生成されるデータのタイプであるので、R G B データは、一般に、入力色空間として利用される。しかしながら、R G B 色空間は、そのコンポーネントの間で高い冗長性を有し、圧縮した表現のために最適でない。より圧縮したおよびよりロバストな表現を達成するために、R G B コンポーネントは、一般に、圧縮により好適であるより無相関な色空間、たとえば、Y C b C r にコンバートされる（たとえば、色変換が実行される）。Y C b C r 色空間は、異なる、あまり相関しないコンポーネントにおいて、ルミナンス（Y）および色情報（C r C b）の形態で輝度を分離する。このコンテキストでは、ロバストな表現は、条件付きビットレートにおいて圧縮されたとき、誤差耐性のより高いレベルを起用する色空間を指し得る。

40

#### 【 0072 】

[0083] 色変換に続いて、ターゲット色空間中の入力データは、依然として、高ビット深度（たとえば浮動小数点正確さ）で表され得る。高ビット深度データは、たとえば、量子化プロセスを使用して、ターゲットビット深度にコンバートされ得る。いくつかの研究は、P Q 伝達と組み合わせた 10 ~ 12 ビット正確さが、丁度可知差（J N D : Just-Not-iceable Difference）を下回るひずみをもつ 16 個の f ストップの H D R データを与えるのに十分であることを示す。概して、J N D は、差が（たとえば、H V S によって）それとわかるようになるために、何か（たとえば、ビデオデータ）が変更しなければならない

50

量である。10ビット正確さで表されたデータは、さらに、最先端のビデオコーディングソリューションのほとんどを用いてコーディングされ得る。この量子化は、不可逆コーディングの要素であり、コンバートされたデータにもたらされる不正確さの源である。

【0073】

[0084] 次世代HDR/WCGビデオ適用例は、HDRおよびCGの異なるパラメータにおいてキャプチャされたビデオデータを用いて動作することが予期される。異なる構成の例は、最高1000ニトの、または最高10,000ニトのピーク輝度をもつHDRビデオコンテンツのキャプチャであり得る。異なる色域の例としては、BT.709、BT.2020同様にSMPTE指定P3、または他のものがあり得る。

【0074】

[0085] また、将来利用されるべき、すべての他の現在使用されている色域を組み込む（またはほとんど組み込む）、単一の色空間、たとえば、ターゲット色コンテナが予期される。そのようなターゲット色コンテナの一例は、BT.2020である。単一のターゲット色コンテナのサポートは、低減された数の動作点（たとえば、色コンテナ、色空間、色コンバージョンアルゴリズムなどの数）および／または低減された数の必要とされるアルゴリズムがデコーダ（たとえば、ビデオデコーダ30）によってサポートされるべきである、HDR/WCGシステムの規格化、実装および展開を著しく簡略化することになる。

【0075】

[0086] そのようなシステムの一例では、ターゲット色コンテナ（たとえばBT.2020）とは異なるネイティブ色域（たとえばP3またはBT.709）を用いてキャプチャされたコンテンツは、処理より前に（たとえば、ビデオ符号化より前に）ターゲットコンテナにコンバートされ得る。以下は、そのようなコンバージョンのいくつかの例である。BT.709からBT.2020色コンテナへのRGBコンバージョン。

【0076】

【数1】

- $R_{2020} = 0.627404078626 * R_{709} + 0.329282097415 * G_{709} + 0.043313797587 * B_{709}$
- $G_{2020} = 0.069097233123 * R_{709} + 0.919541035593 * G_{709} + 0.011361189924 * B_{709}$
- $B_{2020} = 0.016391587664 * R_{709} + 0.088013255546 * G_{709} + 0.895595009604 * B_{709}$

(1)

【0077】

P3からBT.2020色コンテナへのRGBコンバージョン。

【0078】

【数2】

- $R_{2020} = 0.753832826496 * R_{P3} + 0.198597635641 * G_{P3} + 0.047569409186 * B_{P3}$
- $G_{2020} = 0.045744636411 * R_{P3} + 0.941777687331 * G_{P3} + 0.012478735611 * B_{P3}$
- $B_{2020} = -0.001210377285 * R_{P3} + 0.017601107390 * G_{P3} + 0.983608137835 * B_{P3}$

(2)

【0079】

[0087] このコンバージョン中に、P3またはBT.709色域においてキャプチャされた信号の各コンポーネント（たとえば、RGB、YUV、YCrCbなど）によって占有される値レンジは、BT.2020表現において低減され得る。データが浮動小数点正確さで表されるので喪失はないが、色コンバージョン（たとえば、以下の式3に示されているRGBからYCrCbへのコンバージョン）および量子化（以下の式4中の例）と組み合わせられたとき、値レンジの縮小は、入力データのための増加された量子化誤差をもたらす。

【0080】

10

20

30

40

## 【数3】

$$\circ \quad Y' = 0.2627 * R' + 0.6780 * G' + 0.0593 * B'; \quad Cb = \frac{B' - Y'}{1.8814}; \quad Cr = \frac{R' - Y'}{1.4746} \quad (3)$$

## 【0081】

## 【数4】

$$\begin{aligned} \circ \quad D_{Y'} &= \left( \text{Round} \left( (1 \ll (\text{BitDepth}_Y - 8)) * (219 * Y' + 16) \right) \right) \\ \circ \quad D_{Cb} &= \left( \text{Round} \left( (1 \ll (\text{BitDepth}_{Cr} - 8)) * (224 * Cb + 128) \right) \right) \\ \circ \quad D_{Cr} &= \left( \text{Round} \left( (1 \ll (\text{BitDepth}_{Cb} - 8)) * (224 * Cr + 128) \right) \right) \end{aligned} \quad (4)$$

10

## 【0082】

式(4)において、 $D_{Y'}$ は、量子化された $Y'$ コンポーネントであり、 $D_{Cb}$ は、量子化された $Cb$ であり、 $D_{Cr}$ は、量子化された $Cr$ コンポーネントである。 $\ll$ という用語はビット単位右シフトを表す。 $\text{BitDepth}_Y$ 、 $\text{BitDepth}_{Cr}$ 、および $\text{BitDepth}_{Cb}$ はそれぞれ、量子化されたコンポーネントの所望のビット深度である。

## 【0083】

20

[0088] さらに、現実世界のコーディングシステムでは、低減されたダイナミックレンジを用いて信号をコーディングすることは、コード化クロマコンポーネントのための正確さの著しい喪失をもたらし得、コーディングアーティファクト、たとえば、色ミスマッチおよび/または色ブリーディングとして視聴者によって観測されることになる。

## 【0084】

[0089] 上記で説明された問題に対処するために、後続の技法が考慮され得る。1つの例示的な技法は、ネイティブ色空間におけるH D Rコーディングを伴う。そのような技法では、H D Rビデオコーディングシステムは、様々なタイプの現在知られている色域をサポートし、ビデオコーディング規格の拡張が将来の色域をサポートすることを可能にすることになる。このサポートは、異なる色コンバージョン変換、たとえばR G BからY C b C rへの、およびそれらの逆変換をサポートすることのみに限定されないことになり、また、色域の各々に調整された変換関数を指定することになる。ツールのそのような多様性のサポートは、複雑であり、費用がかかることになる。

30

## 【0085】

[0090] 別の例示的な技法は、色域アウェア(aware)ビデオコーデックを含む。そのような技法では、仮想ビデオエンコーダは、低減されたダイナミックレンジから生じるひずみを低減するために、入力信号のネイティブ色域を推定し、コーディングパラメータ(たとえば、コード化クロマコンポーネントのための量子化パラメータ)を調整するように構成される。しかしながら、そのような技法は、すべての入力データが整数点正確さにおいて一般的なコーデックに与えられるので、上記の式(4)中で行われる量子化により起こり得る正確さの喪失を復元することが可能でないであろう。

40

## 【0086】

[0091] 本開示は、色域コンバージョンによってH D R信号表現にもたらされるダイナミックレンジ変化を補償するために、ダイナミックレンジ調整(D R A)を実行するための技法、方法、および装置について説明する。ダイナミックレンジ調整は、色ミスマッチ、色ブリーディングなどを含む、色域コンバージョンによって引き起こされるひずみを防ぐ、および/または少なくするのを助け得る。本開示の1つまたは複数の例では、D R Aは、(たとえば、ソースデバイス12による)エンコーダ側における量子化より前に、および(たとえば、宛先デバイス14による)デコーダ側における逆量子化の後に、ターゲット色空間、たとえば、Y C b C rの各色コンポーネントの値に対して行われる。

50

## 【0087】

[0092] 図8は、本開示の技法に従って動作する例示的なH D R / W C Gコンバージョン装置を示すブロック図である。図8において、実線はデータフローを指定し、破線は制御信号を指定する。本開示の技法は、ソースデバイス12のビデオプリプロセッサユニット19によって実行され得る。上記で説明されたように、ビデオプリプロセッサユニット19は、ビデオエンコーダ20とは別個のデバイスであり得る。他の例では、ビデオプリプロセッサユニット19は、ビデオエンコーダ20と同じデバイスに組み込まれ得る。

## 【0088】

[0093] 図8に示されているように、R G BネイティブC Gビデオデータ200が、ビデオプリプロセッサユニット19に入力される。ビデオプリプロセッサユニット19によるビデオ前処理のコンテキストでは、R G BネイティブC Gビデオデータ200は、入力色コンテナによって定義される。入力色コンテナが、ビデオデータ200を表すために使用される色プライマリのセット（たとえば、B T . 709、B T . 2020、P 3など）を指定する。本開示の一例では、ビデオプリプロセッサユニット19は、R G BネイティブC Bビデオデータ200の色コンテナと色空間の両方を、H D R 'データ216のためのターゲット色コンテナおよびターゲット色空間にコンバートするように構成され得る。入力色コンテナのように、ターゲット色コンテナが、H D R 'データ216を表すために使用されるセットまたは色プライマリを指定し得る。本開示の一例では、R G BネイティブC Bビデオデータ200は、H D R / W C Gビデオであり得、B T . 2020またはP 3色コンテナ（または任意のW C G）を有し、R G B色空間中にあり得る。別の例では、R G BネイティブC Bビデオデータ200は、S D Rビデオであり得、B T . 709色コンテナを有し得る。一例では、H D R 'データ216のためのターゲット色コンテナは、H D R / W C Gビデオ（たとえば、B T . 2020色コンテナ）のために構成されていることがあり、ビデオ符号化のためにより最適な色空間（たとえば、Y C r C b）を使用し得る。

## 【0089】

[0094] 本開示の一例では、C Gコンバータ202が、R G BネイティブC Gビデオデータ200の色コンテナを、入力色コンテナ（たとえば、第1の色コンテナ）からターゲット色コンテナ（たとえば、第2の色コンテナ）にコンバートするように構成され得る。一例として、C Gコンバータ202は、R G BネイティブC Gビデオデータ200を、B T . 709色表現からB T . 2020色表現にコンバートし得、その例が以下に示されている。

## 【0090】

[0095] R G B B T . 709サンプル（R<sub>709</sub>、G<sub>709</sub>、B<sub>709</sub>）をR G B B T . 2020サンプル（R<sub>2020</sub>、G<sub>2020</sub>、B<sub>2020</sub>）にコンバートするためのプロセスは、適切なコンバージョン行列を使用して、最初にX Y Z表現にコンバートし、その後にX Y ZからR G B B T . 2020へのコンバージョンが続くことを伴う、2ステップコンバージョンを用いて実装され得る。

## 【0091】

## 【数5】

$$\begin{aligned}
 X &= 0.412391 * R_{709} + 0.357584 * G_{709} + 0.180481 * B_{709} \\
 Y &= 0.212639 * R_{709} + 0.715169 * G_{709} + 0.072192 * B_{709} \\
 Z &= 0.019331 * R_{709} + 0.119195 * G_{709} + 0.950532 * B_{709}
 \end{aligned} \tag{5}$$

## 【0092】

[0096] X Y ZからR<sub>2020</sub> G<sub>2020</sub> B<sub>2020</sub>（B T . 2020）へのコンバージョン

## 【0093】

10

20

30

40

## 【数6】

$$\begin{aligned}
 R_{2020} &= \text{clipRGB}( 1.716651 * X - 0.355671 * Y - 0.253366 * Z ) \\
 G_{2020} &= \text{clipRGB}( -0.666684 * X + 1.616481 * Y + 0.015768 * Z ) \\
 B_{2020} &= \text{clipRGB}( 0.017640 * X - 0.042771 * Y + 0.942103 * Z )
 \end{aligned} \tag{6}$$

## 【0094】

同様に、単一のステップおよび推奨方法は以下の通りである。

## 【0095】

## 【数7】

$$\begin{aligned}
 R_{2020} &= \text{clipRGB}( 0.627404078626 * R_{709} + 0.329282097415 * G_{709} + 10 \\
 &\quad 0.043313797587 * B_{709} ) \\
 G_{2020} &= \text{clipRGB}( 0.069097233123 * R_{709} + 0.919541035593 * G_{709} + \\
 &\quad 0.011361189924 * B_{709} ) \tag{7} \\
 B_{2020} &= \text{clipRGB}( 0.016391587664 * R_{709} + 0.088013255546 * G_{709} + \\
 &\quad 0.895595009604 * B_{709} )
 \end{aligned}$$

## 【0096】

[0097] CGコンバージョンの後の得られたビデオデータが、図8中のRGBターゲットCGビデオデータ204として示されている。本開示の他の例では、入力データと出力HDR'データとのための色コンテナは同じであり得る。そのような例では、CGコンバータ202は、RGBネイティブCGビデオデータ200に対してコンバージョンを実行する必要はない。

## 【0097】

[0098] 次に、伝達関数ユニット206が、RGBターゲットCGビデオデータ204のダイナミックレンジを圧縮する。伝達関数ユニット206は、図5を参照しながら上記で説明されたのと同じ様式でダイナミックレンジを圧縮するために、伝達関数を適用するように構成され得る。色コンバージョンユニット208が、RGBターゲットCG色データ204を、入力色コンテナの色空間（たとえば、RGB）からターゲット色コンテナの色空間（たとえば、YCrCb）にコンバートする。図5を参照しながら上記で説明されたように、色コンバージョンユニット208は、ハイブリッドビデオエンコーダ（たとえば、ビデオエンコーダ20）による圧縮により好適であるより圧縮したまたはロバストな色空間（たとえば、YUVまたはYCrCb色空間）に、圧縮されたデータをコンバートする。

## 【0098】

[0099] 調整ユニット210が、DRAパラメータ推定ユニット212によって導出されたDRAパラメータに従って、色コンバートされたビデオデータのダイナミックレンジ調整（DRA）を実行するように構成される。概して、CGコンバータ202によるCGコンバージョンおよび伝達関数ユニット206によるダイナミックレンジ圧縮の後に、得られたビデオデータの実際の色値は、特定のターゲット色コンテナの色域のために割り振られたすべての利用可能なコードワード（たとえば、各色を表す特有のビットシーケンス）を使用しないことがある。すなわち、いくつかの状況では、入力色コンテナから出力色コンテナへのRGBネイティブCGビデオデータ200のコンバージョンは、得られた圧縮されたビデオデータがすべての可能な色表現を効率的に利用しないように、ビデオデータの色値（たとえば、CrおよびCb）を過度に圧縮し得る。上記で説明されたように、色のための値の低減されたレンジを用いて信号をコーディングすることは、コード化クロマコンポーネントのための正確さの著しい喪失をもたらし得、コーディングアーティファクト、たとえば、色ミスマッチおよび/または色ブリーディングとして視聴者によって観測されることになる。

## 【0099】

[0100] 調整ユニット210は、特定のターゲット色コンテナのために利用可能なコ-

10

20

30

40

50

ドワードを十分に利用するために、ダイナミックレンジ圧縮および色コンバージョンの後に、ビデオデータ、たとえば、RGBターゲットCGビデオデータ204の色コンポーネント（たとえば、YCrCb）にDRAパラメータを適用するように構成され得る。調整ユニット210は、ピクセルレベルにおいてビデオデータにDRAパラメータを適用し得る。概して、DRAパラメータは、ターゲット色コンテナのために利用可能なだけ多くのコードワードに実際のビデオデータを表すために使用されるコードワードを拡大する関数を定義する。

#### 【0100】

[0101] 本開示の一例では、DRAパラメータは、ビデオデータのコンポーネントに適用されるスケールおよびオフセット値を含む。概して、ビデオデータの色コンポーネントの値レンジが低いほど、大きいスケーリングファクタが使用され得る。オフセットパラメータ（offset parameter）は、色コンポーネントの値の中心をターゲット色コンテナのための利用可能なコードワードの中心に置くために使用され得る。たとえば、ターゲット色コンテナが色コンポーネントごとに1024個のコードワードを含む場合、オフセット値は、中心コードワードがコードワード512（たとえば、最中間コードワード）に移動されるように選定され得る。他の例では、オフセットパラメータは、ターゲット色コンテナ中の全体的な表現が、コーディングアーティファクトをなくす際により効率的であるように、出力コードワードへの入力コードワードのより良いマッピングを与えるために使用され得る。

#### 【0101】

[0102] 一例では、調整ユニット210は、以下のように、ターゲット色空間（たとえば、YCrCb）中のビデオデータにDRAパラメータを適用する。

#### 【0102】

#### 【数8】

$$\begin{aligned}
 - & Y'' = scale1 * Y' + offset1 \\
 - & Cb'' = scale2 * Cb' + offset2 \\
 - & Cr'' = scale3 * Cr' + offset3
 \end{aligned} \tag{8}$$

#### 【0103】

ここで、信号コンポーネントY'、Cb'およびCr'は、RGBからYCrCbへのコンバージョン（式3中の例）から生成された信号である。Y'、Cr'およびCb'は、ビデオデコーダ30によって復号されたビデオ信号でもあり得ることに留意されたい。Y''、Cb''およびCr''は、DRAパラメータが各色コンポーネントに適用された後のビデオ信号の色コンポーネントである。上記の例に見られるように、各色コンポーネントは、異なるスケールおよびオフセットパラメータに関係する。たとえば、scale1およびoffset1がY'コンポーネントのために使用され、scale2およびoffset2がCb'コンポーネントのために使用され、scale3およびoffset3がCr'コンポーネントのために使用される。これはただの一例であることを理解されたい。他の例では、同じスケールおよびオフセット値が、あらゆる色コンポーネントのために使用され得る。

#### 【0104】

[0103] 他の例では、各色コンポーネントは、複数のスケールおよびオフセットパラメータに関連し得る。たとえば、CrまたはCb色コンポーネントのためのクロマ値の実際の分散は、コードワードの異なるパーティションまたはレンジについて異なり得る。一例として、中心コードワード（たとえば、コードワード512）の下にあるよりも、中心コードワードの上で使用されるより多くの特有のコードワードがあり得る。そのような例では、調整ユニット210は、中心コードワードを上回る（たとえば、中心コードワードよりも大きい値を有する）クロマ値のためにスケールおよびオフセットパラメータの1つのセットを適用し、中心コードワードを下回る（たとえば、中心コードワードよりも小さい値を有する）クロマ値のためにスケールおよびオフセットパラメータの異なるセットを適

10

20

30

40

50

用するように構成され得る。

【0105】

[0104] 上記の例に見られるように、調整ユニット210は、線形関数としてスケールおよびオフセットDRAパラメータを適用する。したがって、調整ユニット210が、色コンバージョンユニット208による色コンバージョンの後にターゲット色空間中でDRAパラメータを適用することは必要でない。これは、色コンバージョンが、それ自体が線形プロセスであるからである。したがって、他の例では、調整ユニット210は、色コンバージョンプロセスの前にネイティブ色空間（たとえば、RGB）中のビデオデータにDRAパラメータを適用し得る。この例では、色コンバージョンユニット208は、調整ユニット210がDRAパラメータを適用した後に色コンバージョンを適用することになる。

10

【0106】

[0105] 本開示の別の例では、調整ユニット210は、以下のように、ターゲット色空間またはネイティブ色空間のいずれか中でDRAパラメータを適用し得る。

【0107】

【数9】

$$\begin{aligned} - & Y'' = (\text{scale1} * (Y' - \text{offsetY}) + \text{offset1}) + \text{offsetY}; \\ - & Cb'' = \text{scale2} * Cb' + \text{offset2} \\ - & Cr'' = \text{scale3} * Cr' + \text{offset3} \end{aligned} \quad (9)$$

20

【0108】

この例では、パラメータscale1、scale2、scale3、offset1、offset2、およびoffset3は、上記で説明されたのと同じ意味を有する。パラメータoffsetYは、信号の輝度を反映するパラメータであり、Y'の平均値に等しくなり得る。他の例では、オフセットYと同様のオフセットパラメータが、入力および出力表現中の中心値のマッピングをより良く保存するために、Cb'およびCr'コンポーネントのために適用され得る。

【0109】

[0106] 本開示の別の例では、調整ユニット210は、ネイティブ色空間またはターゲット色空間以外の色空間中でDRAパラメータを適用するように構成され得る。概して、調整ユニット210は、以下のように、DRAパラメータを適用するように構成され得る。

30

【0110】

【数10】

$$\begin{aligned} - & A' = \text{scale1} * A + \text{offset1}; \\ - & B' = \text{scale2} * B + \text{offset2} \\ - & C' = \text{scale3} * C + \text{offset3} \end{aligned} \quad (10)$$

【0111】

ここで、信号コンポーネントA、BおよびCは、ターゲット色空間とは異なる色空間、たとえば、RGBまたは中間色空間中の信号コンポーネントである。

40

【0112】

[0107] 本開示の他の例では、調整ユニット210は、DRAを実行するためにビデオに線形伝達関数を適用するように構成される。そのような伝達関数は、ダイナミックレンジを圧縮するために伝達関数ユニット206によって使用される伝達関数とは異なる。上記で定義されたスケールおよびオフセットという用語と同様に、調整ユニット210によって適用される伝達関数は、ターゲット色コンテナ中の利用可能なコードワードに色値を拡大し、色値をその中心に置くために使用され得る。DRAを実行するために伝達関数を適用することの一例が、以下に示されている。

【0113】

50

## 【数11】

- $Y'' = TF2(Y')$
- $Cb'' = TF2(Cb')$
- $Cr'' = TF2(Cr')$

## 【0114】

TF2という用語は、調整ユニット210によって適用される伝達関数を指定する。いくつかの例では、調整ユニット210は、コンポーネントの各々に異なる伝達関数を適用するように構成され得る。

## 【0115】

[0108] 本開示の別の例では、調整ユニット210は、単一のプロセスにおいて、色コンバージョンユニット208の色コンバージョンと一緒にDRAパラメータを適用するように構成され得る。すなわち、調整ユニット210と色コンバージョンユニット208との線形関数は組み合わせられ得る。f1およびf2が、RGBからYCbCrへの行列とDRAスケーリングファクタとの組合せである、組み合わせられた適用例の一例が、以下に示されている。

## 【0116】

## 【数12】

$$Cb = \frac{B' - Y'}{f1}; Cr = \frac{R' - Y'}{f2}$$

## 【0117】

[0109] 本開示の別の例では、DRAパラメータを適用した後に、調整ユニット210は、ビデオデータが、あるターゲット色コンテナのために指定されたコードワードのレンジ外に値を有するのを防ぐために、クリッピングプロセスを実行するように構成され得る。いくつかの状況では、調整ユニット210によって適用されるスケールおよびオフセットパラメータは、いくつかの色コンポーネント値が、許容可能なコードワードのレンジを超えることを引き起こし得る。この場合、調整ユニット210は、レンジを超えるコンポーネントの値をレンジ中の最大値にクリッピングするように構成され得る。

## 【0118】

[0110] 調整ユニット210によって適用されるDRAパラメータは、DRAパラメータ推定ユニット212によって決定され得る。DRAパラメータ推定ユニット212がDRAパラメータを更新する頻度および時間インスタンスは、フレキシブルである。たとえば、DRAパラメータ推定ユニット212は、時間レベルでDRAパラメータを更新し得る。すなわち、ピクチャのグループ(GOP)、または単一のピクチャ(フレーム)について、新しいDRAパラメータが決定され得る。この例では、RGBネイティブCGビデオデータ200は、GOPまたは単一のピクチャであり得る。他の例では、DRAパラメータ推定ユニット212は、空間レベルで、たとえば、スライスタイルにおいて、またはブロックレベルでDRAパラメータを更新し得る。このコンテキストでは、ビデオデータのブロックは、マクロブロック、コーディングツリーユニット(CTU)、コーディングユニット、またはブロックの任意の他のサイズおよび形状であり得る。ブロックは、正方形、矩形、または任意の他の形状であり得る。したがって、DRAパラメータは、より効率的な時間および空間予測ならびにコーディングのために使用され得る。

## 【0119】

[0111] 本開示の一例では、DRAパラメータ推定ユニット212は、RGBネイティブCGビデオデータ200のネイティブ色域とターゲット色コンテナの色域との対応に基づいてDRAパラメータを導出し得る。たとえば、DRAパラメータ推定ユニット212は、ネイティブ色域(たとえば、BT.709)およびターゲット色コンテナの色域(たとえば、BT.2020)を前提として、スケールおよびオフセット値を決定するためにあらかじめ定義されたルールのセットを使用し得る。

## 【0120】

10

20

30

40

50

[0112] たとえば、ネイティブ色域およびターゲット色コンテナが、 $x$   $y$  空間中の色プライマリ座標および白色点座標の形態で定義されると仮定する。BT.709およびBT.2020のためのそのような情報の一例が、以下の表2に示されている。

【0121】

【表2】

表2-RGB色空間パラメータ

色空間	RGB色空間パラメータ							
	白色点		原色					
	xxw	yyw	xxr	yyr	xxg	yyg	xxb	yyb
DCI-P3	0.314	0.351	0.680	0.320	0.265	0.690	0.150	0.060
ITU-R BT.709	0.3127	0.3290	0.64	0.33	0.30	0.60	0.15	0.06
ITU-R BT.2020	0.3127	0.3290	0.708	0.292	0.170	0.797	0.131	0.046

【0122】

[0113] 一例では、BT.2020はターゲット色コンテナの色域であり、BT.709はネイティブ色コンテナの色域である。この例では、調整ユニット210は、YCbCrターゲット色空間にDRAパラメータを適用する。DRAパラメータ推定ユニット212は、以下のように、DRAパラメータを推定し、それを調整ユニット210にフォワーディングするように構成され得る。

【0123】

【数13】

```
scale1 = 1;          offset1 = 0;
scale2 = 1.0698;    offset2 = 0;
scale3 = 2.1735;    offset3 = 0;
```

【0124】

[0114] 別の例として、BT.2020がターゲット色域であり、P3がネイティブ色域であり、DRAがYCbCrターゲット色空間中で適用される場合、DRAパラメータ推定ユニット212は、次のようにDRAパラメータを推定するように構成され得る。

【0125】

【数14】

```
scale1 = 1;          offset1 = 0;
scale2 = 1.0068;    offset2 = 0;
scale3 = 1.7913;    offset3 = 0;
```

【0126】

[0115] 上記の例では、DRAパラメータ推定ユニット212は、あるネイティブ色域およびあるターゲット色域を前提として、使用すべきDRAパラメータを示すルックアップテーブルを調べることによって、上記のスケールおよびオフセット値を決定するように構成され得る。他の例では、DRAパラメータ推定ユニット212は、たとえば、表2に示されているように、ネイティブ色域およびターゲット色域のプライマリおよび白色空間値からDRAパラメータを計算するように構成され得る。

【0127】

[0116] たとえば、プライマリ座標( $x_{Xt}$ 、 $y_{Xt}$ )によって指定されたターゲット( $T$ )色コンテナを考慮し、ここで、 $X$ はR、G、B色コンポーネントを示す。

【0128】

10

20

30

40

50

【数15】

$$primeT = \begin{bmatrix} xRt & yRt \\ xGt & yGt \\ xBt & yBt \end{bmatrix}$$

【0129】

および、プライマリ座標 (  $x \times n$  、  $y \times n$  ) によって指定されたネイティブ ( N ) 色域を考慮し、ここで、 X は R 、 G 、 B 色コンポーネントを示す。

【0130】

【数16】

$$primeN = \begin{bmatrix} xRn & yRn \\ xGn & yGn \\ xBn & yBn \end{bmatrix}$$

10

【0131】

両方の色域のための白色点座標は、  $w h i t e P = ( x W 、 y W )$  に等しい。 D R A パラメータ推定ユニット 212 は、プライマリ座標～白色点の間の距離の関数として D R A のための  $s c a l e 2$  および  $s c a l e 3$  パラメータを導出し得る。そのような推定の一例が、以下に与えられる。

【0132】

【数17】

$$rdT = \sqrt{(primeT(1,1) - whiteP(1,1))^2 + (primeN(1,2) - whiteP(1,2))^2}$$

20

$$gdT = \sqrt{(primeT(2,1) - whiteP(1,1))^2 + (primeN(2,2) - whiteP(1,2))^2}$$

$$bdT = \sqrt{(primeT(3,1) - whiteP(1,1))^2 + (primeN(3,2) - whiteP(1,2))^2}$$

$$rdN = \sqrt{(primeN(1,1) - whiteP(1,1))^2 + (primeN(1,2) - whiteP(1,2))^2}$$

$$gdN = \sqrt{(primeN(2,1) - whiteP(1,1))^2 + (primeN(2,2) - whiteP(1,2))^2}$$

$$bdN = \sqrt{(primeN(3,1) - whiteP(1,1))^2 + (primeN(3,2) - whiteP(1,2))^2}$$

$$scale2 = bdT/bdN$$

30

$$scale3 = \sqrt{(rdT/rdN)^2 + (gdT/gdN)^2}$$

【0133】

[0117] いくつかの例では、 D R A パラメータ推定ユニット 212 は、ネイティブ色域のあらかじめ定義されたプライマリ値からでなく、 R G B ネイティブ C G ビデオデータ 200 中の色値の実際の分散から  $p r i m e N$  中のプライマリ座標を決定することによって D R A パラメータを推定するように構成され得る。すなわち、 D R A パラメータ推定ユニット 212 は、 R G B ネイティブ C G ビデオデータ 200 中に存在する実際の色を分析し、 D R A パラメータを計算するために、上記で説明された関数におけるそのような分析から決定されたプライマリ色値および白色点を使用するように構成され得る。算出を容易にするために、上記で定義されたいいくつかのパラメータの近似が D R A として使用され得る。たとえば、  $s c a l e 3 = 2.1735$  は  $s c a l e 3 = 2$  に近似され得、これは、いくつかのアーキテクチャにおけるより容易な実装を可能にする。

40

【0134】

[0118] 本開示の他の例では、 D R A パラメータ推定ユニット 212 は、ターゲット色コンテナの色域だけでなく、ターゲット色空間に基づいて D R A パラメータを決定するように構成され得る。コンポーネント値の値の実際の分散は色空間ごとに異なり得る。たとえば、クロマ値分散は、一定でないルミナンスを有する Y C b C r 色空間と比較して一定のルミナンスを有する Y C b C r 色空間について異なり得る。 D R A パラメータ推定ユニット 212 は、 D R A パラメータを決定するために異なる色空間の色分散を使用し得る。

【0135】

50

[0119] 本開示の他の例では、D R A パラメータ推定ユニット 212 は、ビデオデータを前処理および / または符号化することに関連するいくつかのコスト関数を最小限に抑えるように、D R A パラメータのための値を導出するように構成され得る。一例として、D R A パラメータ推定ユニット 212 は、量子化ユニット 214 によってもたらされる量子化誤差を最小限に抑えた D R A パラメータを推定するように構成され得る上記の（たとえば、式（4）を参照）。D R A パラメータ推定ユニット 212 は、適用された D R A パラメータの異なるセットを有しているビデオデータに対して量子化誤差テストを実行することによって、そのような誤差を最小限に抑え得る。別の例では、D R A パラメータ推定ユニット 212 は、知覚的に、量子化ユニット 214 によってもたらされる量子化誤差を最小限に抑える D R A パラメータを推定するように構成され得る。D R A パラメータ推定ユニット 212 は、適用される D R A パラメータの異なるセットを有しているビデオデータに対する知覚的誤差テストに基づいて、そのような誤差を最小限に抑え得る。D R A パラメータ推定ユニット 212 は、次いで、最も低い量子化誤差を生成した D R A パラメータを選択し得る。  
10

#### 【 0 1 3 6 】

[0120] 別の例では、D R A パラメータ推定ユニット 212 は、調整ユニット 210 によって実行された D R A とビデオエンコーダ 20 によって実行されたビデオ符号化の両方に関連するコスト関数を最小限に抑える D R A パラメータを選択し得る。たとえば、D R A パラメータ推定ユニット 212 は、D R A パラメータの複数の異なるセットを用いて、D R A を実行し、ビデオデータを符号化し得る。D R A パラメータ推定ユニット 212 は、次いで、D R A およびビデオ符号化から生じるビットレート、ならびにこれらの 2 つの不可逆プロセスによってもたらされるひずみの重み付き和を形成することによって、D R A パラメータの各セットのためのコスト関数を計算し得る。D R A パラメータ推定ユニット 212 は、次いで、コスト関数を最小限に抑える D R A パラメータのセットを選択し得る。  
20

#### 【 0 1 3 7 】

[0121] D R A パラメータ推定のための上記の技法の各々では、D R A パラメータ推定ユニット 212 は、そのコンポーネントに関する情報を使用して、各コンポーネントのために別個に D R A パラメータを決定し得る。他の例では、D R A パラメータ推定ユニット 212 は、クロスコンポーネント情報 (cross-component information) を使用して D R A パラメータを決定し得る。たとえば、C r コンポーネントのために導出された D R A パラメータは、C b コンポーネントのための D R A パラメータを導出するために使用され得る。  
30

#### 【 0 1 3 8 】

[0122] D R A パラメータを導出することに加えて、D R A パラメータ推定ユニット 212 は、符号化ビットストリーム中で D R A パラメータをシグナリングするように構成され得る。D R A パラメータ推定ユニット 212 は、D R A パラメータを示す 1 つまたは複数のシンタックス要素を直接シグナリングし得るか、またはシグナリングのためにビデオエンコーダ 20 に 1 つまたは複数のシンタックス要素を与えるように構成され得る。パラメータのそのようなシンタックス要素は、ビデオデコーダ 30 および / またはビデオボストプロセッサユニット 31 が、そのネイティブ色コンテナ中のビデオデータを再構成するために、ビデオプリプロセッサユニット 19 のプロセスの逆を実行し得るように、ビットストリーム中でシグナリングされ得る。D R A パラメータをシグナリングするための例示的な技法は、以下で説明される。  
40

#### 【 0 1 3 9 】

[0123] 一例では、D R A パラメータ推定ユニット 212 は、メタデータとして符号化ビデオビットストリーム中で、補足エンハンスメント情報 (S E I) メッセージ中で、ビデオユーザビリティ情報 (V U I) 中で、ビデオパラメータセット (V P S) 中で、シーケンスパラメータセット (S P S) 中で、ピクチャパラメータセット中で、スライスヘッダ中で、C T U ヘッダ中で、またはビデオデータのサイズ（たとえば、G O P、ピクチャ  
50

、ブロック、マクロブロック、C TUなど)のためのDRAパラメータを示すのに好適な任意の他のシンタックス構造中で1つまたは複数のシンタックス要素をシグナリングし得る。

#### 【0140】

[0124] いくつかの例では、1つまたは複数のシンタックス要素は、DRAパラメータを明示的に示す。たとえば、1つまたは複数のシンタックス要素は、DRAのための様々なスケールおよびオフセット値であり得る。他の例では、1つまたは複数のシンタックス要素は、DRAのためのスケールおよびオフセット値を含むルックアップテーブルへの1つまたは複数のインデックス(index)であり得る。さらに別の例では、1つまたは複数のシンタックス要素は、DRAのために使用するための線形伝達関数を指定するルックアップテーブルへのインデックスであり得る。 10

#### 【0141】

[0125] 他の例では、DRAパラメータは明示的にシグナリングされず、むしろ、ビデオプリプロセッサユニット19とビデオポストプロセッサユニット31の両方が、ビットストリーム識別可能な形態であるビデオデータの同じ情報および/または特性を使用する同じあらかじめ定義されたプロセスを使用してDRAパラメータを導出するように構成される。一例として、ビデオポストプロセッサユニット31は、符号化ビットストリーム中でビデオデータのネイティブ色コンテナならびに符号化ビデオデータのターゲット色コンテナを示すように構成され得る。ビデオポストプロセッサユニット31は、次いで、上記で定義されたのと同じプロセスを使用してそのような情報からDRAパラメータを導出するように構成され得る。いくつかの例では、ネイティブおよびターゲット色コンテナを識別する1つまたは複数のシンタックス要素は、シンタックス構造で供給される。そのようなシンタックス要素は明示的に色コンテナを示し得るか、またはルックアップテーブルへのインデックスであり得る。別の例では、ビデオプリプロセッサユニット19は、特定の色コンテナのための色プライマリおよび白色点のXY値を示す1つまたは複数のシンタックス要素をシグナリングするように構成され得る。別の例では、ビデオプリプロセッサユニット19は、DRAパラメータ推定ユニット212によって実行された分析に基づいて、ビデオデータ中の実際の色値の色プライマリおよび白色点(コンテンツプライマリおよびコンテンツ白色点)のXY値を示す1つまたは複数のシンタックス要素をシグナリングするように構成され得る。 20

#### 【0142】

[0126] 一例として、コンテンツ中の色を含んでいる最も小さい色域の色プライマリがシグナリングされ得、ビデオデコーダ30および/またはビデオポストプロセッサユニット31において、DRAパラメータは、コンテナプライマリとコンテンツプライマリの両方を使用して導出される。一例では、コンテンツプライマリは、上記で説明されたように、R、GおよびBのためのxおよびyコンポーネントを使用してシグナリングされ得る。別の例では、コンテンツプライマリは、2つの知られているプライマリセット間の比としてシグナリングされ得る。たとえば、コンテンツプライマリは、(GおよびBコンポーネントのための $a_1 f a_g$ および $a_1 f a_b$ の場合と同様の式を用いて) BT.709プライマリとBT.2020プライマリとの間の線形位置、 $x_{r\_content} = a_1 f a_r * x_{r_bt709} + (1 - a_1 f a_r) * x_{r_bt2020}$ としてシグナリングされ得、ここで、パラメータ $a_1 f a_r$ は、2つの知られているプライマリセット間の比を指定する。いくつかの例では、シグナリングされたおよび/または導出されたDRAパラメータは、HDR/WCGビデオデータのコーディングのために利用される重み付け予測ベースの技法を可能にするために、ビデオエンコーダ20および/またはビデオデコーダ30によって使用され得る。 40

#### 【0143】

[0127] 重み付け予測を利用するビデオコーディング方式では、現在コーディングされているピクチャS<sub>c</sub>のサンプルが、重み(W<sub>wp</sub>)およびオフセット(O<sub>wp</sub>)を用いてとられた参照ピクチャS<sub>r</sub>の(单方向予測のための)サンプルから予測され、これは、予測サンプルS<sub>p</sub>を生じる。 50

【 0 1 4 4 】

【数 1 8 】

$$Sp = Sr \cdot * W_{wp} + O_{wp}$$

【 0 1 4 5 】

[0128] D R A を利用するいくつかの例では、参照ピクチャのサンプルと現在コーディングされているピクチャのサンプルとは、異なるパラメータ、すなわち現在ピクチャのための { scale1<sub>cur</sub>, offset1<sub>cur</sub> } および参照ピクチャのための { scale1<sub>ref</sub>, offset1<sub>ref</sub> } を採用する D R A を用いて処理され得る。そのような実施形態では、重み付け予測のパラメータは、たとえば、以下のように、D R A から導出され得る。

10

【 0 1 4 6 】

【数 1 9 】

$$W_{wp} = scale1_{cur} / scale1_{ref}$$

$$O_{wp} = offset1_{cur} - offset1_{ref}$$

【 0 1 4 7 】

[0129] 調整ユニット 210 が D R A パラメータを適用した後に、ビデオプリプロセッサユニット 19 は、次いで、量子化ユニット 214 を使用してビデオデータを量子化し得る。量子化ユニット 214 は、図 4 に関して上記で説明されたのと同じ様式で動作し得る。量子化後に、ビデオデータは、次に、H D R ' データ 216 のターゲットプライマリのターゲット色空間およびターゲット色域において調整される。H D R ' データ 216 は、次いで、圧縮のためにビデオエンコーダ 20 に送られ得る。

20

【 0 1 4 8 】

[0130] 図 9 は、本開示の技法による例示的な H D R / W C G 逆コンバージョン装置を示すブロック図である。図 9 に示されているように、ビデオポストプロセッサユニット 31 が、図 8 のビデオプリプロセッサユニット 19 によって実行された技法の逆を適用するように構成され得る。他の例では、ビデオポストプロセッサユニット 31 の技法は、ビデオデコーダ 30 に組み込まれ、それによって実行され得る。

【 0 1 4 9 】

[0131] 一例では、ビデオデコーダ 30 は、ビデオエンコーダ 20 によって符号化されたビデオデータを復号するように構成され得る。次いで、復号ビデオデータ（ターゲット色コンテナ中の H D R ' データ 316 ）は、ビデオポストプロセッサユニット 31 にフォワーディングされる。逆量子化ユニット 314 が、図 8 の量子化ユニット 214 によって実行された量子化プロセスを反転させるために、H D R ' データ 316 に対して逆量子化プロセスを実行する。

30

【 0 1 5 0 】

[0132] ビデオデコーダ 30 はまた、図 8 の D R A パラメータ推定ユニット 212 によって生成された 1 つまたは複数のシンタックス要素のうちのいずれかを復号し、それをビデオポストプロセッサユニット 31 の D R A パラメータ導出ユニット 312 に送るように構成され得る。D R A パラメータ導出ユニット 312 は、上記で説明されたように、1 つまたは複数のシンタックス要素に基づいて D R A パラメータを決定するように構成され得る。いくつかの例では、1 つまたは複数のシンタックス要素は、D R A パラメータを明示的に示す。他の例では、D R A パラメータ導出ユニット 312 は、図 8 の D R A パラメータ推定ユニット 212 によって使用された同じ技法を使用して D R A パラメータを導出するように構成される。

40

【 0 1 5 1 】

[0133] D R A パラメータ導出ユニット 312 によって導出されたパラメータは、逆調整ユニット 310 に送られる。逆調整ユニット 310 は、調整ユニット 210 によって実行された線形 D R A 調整の逆を実行するために D R A パラメータを使用する。逆調整ユニット 310 は、調整ユニット 210 のための上記で説明された調整技法のいずれかの逆を

50

適用し得る。さらに、調整ユニット210の場合と同様に、逆調整ユニット310は、逆色コンバージョンの前または後に逆DRAを適用し得る。したがって、逆調整ユニット310は、ターゲット色コンテナまたはネイティブ色コンテナ中のビデオデータに対してDRAパラメータを適用し得る。いくつかの例では、逆調整ユニット310は、逆量子化ユニット314の前に逆調整を適用するように配置され得る。

#### 【0152】

[0134] 逆色コンバージョンユニット(inverse color conversion unit)308が、ターゲット色空間(たとえば、YCbCr)からネイティブ色空間(たとえば、RGB)にビデオデータをコンバートする。次いで、逆伝達関数306が、ビデオデータのダイナミックレンジを圧縮解除(uncompact)するために、伝達関数206によって適用された伝達関数の逆を適用する。いくつかの例では、彼得られたビデオデータ(RGBターゲットCG304)は、依然としてターゲット色域中にあるが、現在、ネイティブダイナミックレンジおよびネイティブ色空間中にある。次に、逆CGコンバータ302が、RGBネイティブCG300を再構成するために、ネイティブ色域にRGBターゲットCG304をコンバートする。

10

#### 【0153】

[0135] いくつかの例では、追加の後処理技法が、ビデオポストプロセッサユニット31によって採用され得る。DRAを適用することは、ビデオを、その実際のネイティブ色域の外側に置き得る。量子化ユニット214および逆量子化ユニット314によって実行される量子化ステップ、ならびに調整ユニット210および逆調整ユニット310によって実行されるアップサンプリングおよびダウンサンプリング技法は、ネイティブ色コンテナ中の得られた色値が、ネイティブ色域の外側になることの原因となり得る。ネイティブ色域(または、上記で説明されたように、シグナリングされた場合、実際の最も小さいコンテンツプライマリ)が知られているとき、DRAのための後処理として意図された色域に色値(たとえば、RGBまたはCbおよびCr)を変換するために、追加のプロセスが、RGBネイティブCGビデオデータ304に適用され得る。他の例では、そのような後処理は、量子化の後に、またはDRA適用の後に適用され得る。

20

#### 【0154】

[0136] 上述のように、いくつかのSEIメッセージが、ビデオデータの様々な色コンポーネントのためのダイナミックレンジ調整情報に関する情報を伝達するために使用され得る。上記で説明されたおよび以下でより詳細に説明されるようなコンポーネントスケーリングSEIメッセージは、ビデオデータの様々な色コンポーネントのためのマッピング情報を示すために使用され得るスケールファクタ、オフセット、およびレンジ(たとえば、コードワード値のパーティション)のセットを伝達し得る。マッピング情報は、再構成されたHDRビデオデータの全体的な品質、または後方互換性が望まれるいくつかの場合には再構成されたSDRビデオデータの品質が改善されるように、あるいは表示能力のためにより好適な再構成された出力を製作するために、ビデオデコーダ30および/またはビデオポストプロセッサユニット31に、サンプル値(sample value)の異なるレンジをどのように拡大または縮小すべきかを示すために使用され得る。

30

#### 【0155】

[0137] 以下の表3は、コンポーネントスケーリングSEIメッセージのシンタックス構造の一変形形態を与える。以下のシンタックス要素の名称が、以下の例で説明されるものとは異なるプレフィックス「hdr\_recon\_」を含んでいるが、シンタックス要素の名称がcomponent\_scalingとしてプレフィックスされる場合、シンタクステーブルは場合によっては同じであることに留意されたい。

40

#### 【0156】

[0138]

#### 【0157】

## 【表3】

表3—例示的なレンジ調整SEIシンタックス

hdr_reconstruction_info( payloadSize ) {	記述子
<b>hdr_recon_id</b>	ue(v)
<b>hdr_recon_cancel_flag</b>	u(1)
if( !hdr_recon_cancel_flag ) {	
<b>hdr_recon_persistence_flag</b>	u(1)
<b>hdr_recon_transfer_characteristics</b>	u(8)
<b>hdr_recon_default_flag</b>	u(1)
if( !hdr_recon_default_flag ) {	
<b>hdr_recon_scale_bit_depth</b>	u(4)
<b>hdr_recon_offset_bit_depth</b>	u(4)
<b>hdr_recon_scale_frac_bit_depth</b>	u(4)
<b>hdr_recon_offset_frac_bit_depth</b>	u(4)
<b>hdr_recon_num_comps_minus1</b>	ue(v)
}	
for( c = 0; c <= hdr_recon_num_comps_minus1; c++ ) {	
<b>hdr_recon_num_ranges[ c ]</b>	ue(v)
<b>hdr_recon_equal_ranges_flag[ c ]</b>	u(1)
<b>hdr_recon_global_offset_val[ c ]</b>	u(v)
for( i = 0; i <= hdr_recon_num_ranges[ c ]; i++ )	
<b>hdr_recon_scale_val[ c ][ i ]</b>	u(v)
if( !hdr_recon_equal_ranges[ c ] )	
for( i = 0; i <= hdr_recon_num_ranges[ c ]; i++ )	
<b>hdr_recon_range_val[ c ][ i ]</b>	u(v)
}	
}	
}	

## 【0158】

[0139] 表3のSEIシンタックスのセマンティクスは、以下で提示される。

## 【0159】

[0140] マッピングプロセスは、区分線形関数マップ[ c ]( )、 $c = 0 \dots \text{hdr\_recon\_num\_comps\_minus1}$ の場合に基づき、これは、以下のように、 $[0, 1]$ における値 $x$ を値 $y = \text{map}[ c ]( x )$ にマッピングする。

10

20

30

40

50

- 両端値を含む、 $0 \sim \text{hdr\_recon\_num\_ranges}[c] - 1$  のレンジ中の  $i$  の場合、以下が適用される。
- 値  $\text{ScaleValue}[c][i]$  は、シンタックス要素  $\text{hdr\_recon\_scale\_val}[c][i]$  のセマンティクスにおいて説明されるように導出される。
- 値  $\text{RangeValue}[c][i]$  は、シンタックス要素  $\text{hdr\_recon\_range\_val}[c][i]$  のセマンティクスにおいて説明されるように導出される。
- 両端値を含む、 $0 \sim \text{hdr\_recon\_num\_ranges}[c] - 1$  のレンジ中の  $i$  の場合、値  $\text{InputRanges}[c][i]$  および  $\text{OutputRanges}[c][i]$  は、以下のように導出される。
- $i$  が 0 に等しい場合、以下が適用される。

【0160】

【数20】

$$\text{OutputRanges}[c][i] = -\text{hdr\_recon\_global\_offset\_val}[c] * \text{ScaleValue}[c][i-1] \quad (\text{D-xx})$$

$$\text{InputRanges}[c][i] = 0 \quad (\text{D-xx})$$

【0161】

- 他の場合 ( $i$  が 0 に等しくない)、以下が適用される。

【0162】

【数21】

$$\text{InputRanges}[c][i] = \text{InputRanges}[c][i-1] + \text{RangeValue}[c][i-1] \quad (\text{D-xx})$$

$$\text{OutputRanges}[c][i] = \text{OutputRanges}[c][i-1] +$$

$$\text{RangeValue}[c][i-1] * \text{ScaleValue}[c][i-1] \quad (\text{D-xx})$$

【0163】

- 両端値を含む、 $0 \sim \text{hdr\_recon\_num\_ranges}[c] - 1$  のレンジ中の  $i$  の場合、値  $\text{OffsetValue}[c][i]$  は、以下のように導出される。

【0164】

【数22】

$$\text{OffsetValue}[c][i] = \text{InputRanges}[c][i+1] - \text{OutputRanges}[c][i+1] \quad (\text{D-xx})$$

$$\text{ScaleValue}[c][i-1] \quad (\text{D-xx})$$

【0165】

- パラメータ  $y = \text{map}[c](x)$  は、以下のように導出される。
- $x$  が  $\text{OutputRanges}[c][0]$  よりも小さいかまたはそれに等しい場合、以下が適用される。

【0166】

【数23】

$$y = \text{InputRanges}[c][0] \quad (\text{D-xx})$$

【0167】

- そうではなく、 $x$  が  $\text{OutputRanges}[c][\text{hdr\_recon\_num\_ranges}[c]]$  よりも大きい場合、以下が適用される。

【0168】

【数24】

$$y = \text{InputRanges}[c][\text{hdr\_recon\_num\_ranges}[c]] \quad (\text{D-xx})$$

【0169】

- 他の場合、以下が適用される。

【0170】

10

20

30

40

## 【数25】

```

for( i = 1; i <= hdr_recon_num_ranges[ c ]; i++ )
    if( OutputRanges[ i - 1 ] < x && x <= OutputRanges[ i ] )
        y = x ÷ ScaleValue[ c ][ i - 1 ] + OffsetValue[ c ][ i - 1 ](D-xx)

```

## 【0171】

[0141] コンポーネントスケーリング情報 S E I メッセージ、およびコンポーネントのダイナミックレンジを調整するために適用可能である他のパラメータに関連するいくつかの問題が識別されている。特に、スケールおよびオフセット値、ならびにサンプル値（たとえば、R G B 値、Y C r C b 値、Y U V 値、X Y Z 値など）のためのコードワードのレンジを導出するための浮動小数点数の使用に関する問題が識別されている。たとえば、ビットストリーム中でシグナリングされるスケール値は、逆ダイナミックレンジ調整プロセスを実行するために、デコーダ側において、たとえば、ビデオポストプロセッサ 3 1 によって使用される。しかしながら、サンプル値のレンジを算出するためおよびマッピングプロセスを算出するために、スケール値を使用するために、逆方向演算（reciprocal operation）が、ビデオポストプロセッサ 3 1 において実行される。コンポーネントスケーリング S E I メッセージのための前の例示的なセマンティクスは、サンプル値を用いて乗算されるべき逆数（たとえば、スケール値の逆数、またはスケール値および加算されたオフセット値の逆数）の使用を指定する。そのような逆方向演算においてもたらされる誤差は、逆数が、発生されたあらゆるサンプル値に適用されるので、順方向演算（forward operation）における潜在的誤差よりも有意になることになる。1020

## 【0172】

[0142] コンポーネントスケーリング S E I メッセージのセマンティクスは、サンプル値のレンジの導出プロセス、および浮動小数点演算に関する色コンポーネント（たとえば、サンプル値）のコードワードの各レンジに対するマッピングプロセス（たとえば、スケールおよびオフセット値の適用）を示す。これは、異なるコンピューティングシステムにおける様々な浮動小数点演算実装に基づく再構成された H D R 出力における差分をもたらすことがある。

## 【0173】

[0143] 本出願は、S E I シグナリングおよび処理を使用してコンポーネントスケーリング情報の通信を改善するためのいくつかの技法、または H . 2 6 5 / H E V C 、 H . 2 6 4 / A V C 、 B D A 、 M P E G または他のものなど、ビデオコーディング規格において指定され得る他の同様のシグナリング技法について説明する。後続の態様のうちの 1 つまたは複数は、独立して、または、特定の例におけるこれらの態様のうちの他のものとの好適な組合せで適用され得ることを認識されたい。30

## 【0174】

[0144] 概して、本開示は、ビデオエンコーダ 2 0 および / またはビデオプリプロセッサユニット 1 9 が、コンポーネントサンプル値（たとえば、色コンポーネント値）の 1 つまたは複数のサンプル値レンジのためのスケール値をシグナリングするように構成され得る技法について説明する。スケール値は、ビデオデコーダ 3 0 およびビデオポストプロセッサユニット 3 1 が、入力サンプル値を含んでいる特定のサンプル値レンジのために指定されたスケール値を入力サンプル値に乗算し、コンポーネントスケーリング情報の一部としてパラメータに基づいて算出されたオフセットを加算することによって、コンポーネントの入力サンプル値から出力サンプル値を取得するためのマッピングプロセスを実行するように構成され得るように指定される。40

## 【0175】

[0145] 本開示の別の例では、色コンポーネントのコードワードのレンジのサイズおよび数を算出するために浮動小数点実装を使用するのではなく、ビデオエンコーダ 2 0 および / またはビデオプリプロセッサユニット 1 9 は、固定小数点算出実装を使用して、色コンポーネントのコードワードのレンジのサイズおよび数を導出するように構成され得る。50

たとえば、ビデオエンコーダ 20 および / またはビデオプリプロセッサユニット 19 は、ダイナミックレンジ調整マッピングプロセスのパラメータを決定および適用するために、所定の数の小数ビット (fractional bit) を使用するように構成され得る。小数ビットの数は、ダイナミックレンジ調整プロセスの各パラメータ (たとえば、各色コンポーネント (コードワード) のための値のレンジ、スケール値、およびオフセット値) について異なり得ることに留意されたい。

#### 【0176】

[0146] たとえば、ビデオプリプロセッサユニット 19 は、色コンポーネントのためのコードワードのレンジのサイズおよび数を通信するために使用される任意のパラメータまたはシンタックス要素 (たとえば `h d r _ r e c o n _ n u m _ r a n g e s [ c ]`) に対して整数演算を実行するように構成され得る。ビデオプリプロセッサユニット 19 は、使用される固定小数点実装におけるレンジのサイズおよび数の任意の計算の小数部分によって使用されるビット数を追跡するように構成され得る。ビデオプリプロセッサユニット 19 および / またはビデオエンコーダ 20 は、SEI メッセージ (たとえば、`h d r _ r e c o n _ o f f s e t _ f r a c _ b i t _ d e p t h`、`h d r _ r e c o n _ s c a l e _ f r a c _ b i t _ d e p t h`) 中の小数部分において使用されるビット数をシグナリングするように構成され得るか、または、小数部分において使用されるビット数は所定の値であり得る。ビデオデコーダ 30 は、小数部分におけるビット数を示す SEI メッセージ中のシンタックス要素を復号するように構成され得、ビデオポストプロセッサユニット 31 は、逆ダイナミックレンジ調整プロセスのパラメータのうちの 1 つまたは複数のための小数部分における同じビット数を使用して、逆ダイナミックレンジ調整を実行するように構成され得る。

#### 【0177】

[0147] 本開示の一例では、マッピングプロセスのためのレンジおよび / または他のパラメータを決定するとき、ビデオデコーダ 30 および / またはビデオポストプロセッサユニット 31 は、異なるパラメータのシグナリングされた小数ビット深度が異なるとき、パラメータのために実行された算出の正確さができるだけ保持されるように、そのようなパラメータを決定するように構成され得る。たとえば、ビデオデコーダ 30 および / またはビデオポストプロセッサユニット 31 は、特定のパラメータを決定するために使用される任意の中間計算ステップにおける小数ビットの数を累算する (accumulate) ことによって、最小値に丸めることによりもたらされる誤差を保持するように構成され得る。ビデオデコーダ 30 および / またはビデオポストプロセッサユニット 31 は、次いで、特定のパラメータの最終値を、特定のパラメータを決定および / または計算する最後のステップにおいて所望の小数正確さ (desired fractional accuracy) にするために、クリッピングプロセスを実行し得る。別の例では、パラメータのシグナリングされた小数ビット深度が同じであるとき、ビデオデコーダ 30 および / またはビデオポストプロセッサユニット 31 は、中間ステップにおける小数ビットの数を累算し、パラメータの最終値を (1 つまたは複数の) 最後のステップにおいて所望の正確さにするために、クリッピングを実行するように構成され得る。

#### 【0178】

[0148] 別の例では、ビデオデコーダ 30 および / またはビデオポストプロセッサユニット 31 は、計算プロセスの 1 つまたは複数の中間ステップにおいてパラメータの値を、またはパラメータのために取得された値の小数部分が所定の値に低減されるようにパラメータの値をクリッピングおよび / またはトランケートする (truncate) ように構成され得る。すなわち、クリッピングを実行するためにパラメータのための最終値を決定するまで待つのではなく、ビデオデコーダ 30 および / またはビデオポストプロセッサユニット 31 は、パラメータを決定するために実行される計算の中間値をクリッピングし得る。そのようなクリッピングまたはトランケーションは、SEI メッセージ中で示される小数ビットの数に基づき得る。別の例では、ビデオデコーダ 30 および / またはビデオポストプロセッサユニット 31 は、動作 / ステップがクリッピングなしに実行されると、小数ビット

10

20

30

40

50

の累算された数が、ある所定の値、たとえば中間値を記憶するために使用されるレジスタのビット深度を超えることになると決定されたとき、特定の動作 / ステップの前に特定のパラメータを計算するときに使用される中間値をクリッピングおよび / またはトランケートするように構成され得る。

【 0 1 7 9 】

[0149] 本開示の別の例では、ビデオプリプロセッサユニット 19 および / またはビデオポストプロセッサユニット 31 は、色コンポーネントの固定表現のために定義された定義された最小値と最大値に基づく所定のサンプル値レンジに従って、スケール、オフセットおよびレンジ値を導出するように構成され得る。たとえば、色コンポーネントの固定表現は、定義された値の複数のレンジ、たとえば、値の「標準」レンジ、値の「フル」レンジ、および「制限付き」レンジ値を有し得る。値の「フル」レンジは、値の「標準」レンジ (たとえば、Y C b C r 色空間の 8 ビット標準レンジ表現、Y コンポーネントは、両端値を含む、16 ~ 235 のレンジ中の値をとり得、C b および C r コンポーネントは、両端値を含む、16 から 240 の間の値をとり得る) と比較して、特定のコンポーネントの最小値と最大値との間のより大きいスパンを有し得る (たとえば、Y C b C r 色空間の 8 ビットフルレンジ表現の場合、Y、C b、および C r コンポーネントは、両端値を含む、0 ~ 255 の激怒中の値をとることができる)。値の「制限付き」レンジは、値の「標準」レンジと比較して、特定のコンポーネントの最小値と最大値との間のより小さいスパンを有し得る (たとえば、Y C b C r 色空間の 10 ビット制限付きレンジ表現の場合、Y、C b、および C r コンポーネントは、両端値を含む、4 ~ 1019 のレンジ中の値をとり得る)。 10

【 0 1 8 0 】

[0150] 一例では、ビデオエンコーダ 20 および / またはビデオプリプロセッサユニット 19 は、どんなサンプルレンジ (たとえばフル、制限付き、標準、または他のもの) が使用されるかに基づいて、ビデオデコーダ 30 および / またはビデオポストプロセッサユニット 31 に、サンプルの最小および最大許容値 (たとえば、色コンポーネント値) を示すために (たとえば、S E I メッセージ中で) シンタックス要素をシグナリングするように構成され得る。別の例では、ビデオエンコーダ 20 および / またはビデオプリプロセッサユニット 19 は、どんなサンプルレンジ (たとえばフル、制限付き、標準) が使用されるかに基づいて、ビデオデコーダに、サンプルの最小および最大許容値を示す 1 つまたは複数のシンタックス値を (たとえば、S E I メッセージ中で) シグナリングするように構成され得る。ビデオデコーダ 30 および / またはビデオポストプロセッサユニット 31 は、次いで、受信された最小値と受信された最大値に基づいて逆ダイナミックレンジ調整プロセスのために許可されるコンポーネント値のレンジを決定し得る。 20

【 0 1 8 1 】

[0151] 別の例では、ビデオエンコーダ 20 および / またはビデオプリプロセッサユニット 19 は、スケール値が符号付き (signed) であるのか符号なし (unsigned) であるのかを示すために、(たとえば、S E I メッセージ中で) フラグをシグナリングするように構成され得る。この例では、任意の S E I メッセージのパースプロセス (parsing process) は、フラグの値にかかわらず同じである。 30

【 0 1 8 2 】

[0152] 後続のセクションは、前のセクションにおいて開示された例示的な技法を使用する実施形態のいくつかの例を含む。この実施形態では、コンポーネントスケーリング関数がルックアップテーブルとしてシグナリングされ、ルックアップテーブルを定義する点をシグナリングするために使用されるビット数もシグナリングされる。一例では、ルックアップは、区分線形マッピング関数を定義する。ルックアップテーブルのための点は、区分線形マッピングを定義する (x, y) 座標に対応する。シグナリングされる明示的点を有しないサンプル値のために、値は隣接ビット点に基づいて補間される。

【 0 1 8 3 】

[0153] レンジおよび出力サンプル値の導出プロセスは、以下のように定義される。

50

## 【0184】

[0154] サンプル  $y = \text{map}[c](x)$  への、コンポーネント  $c$  からのサンプル  $x$  のマッピングは、以下のように指定される。

- Default Prec Shift の値を 9 に等しく設定する
- 変数  $\text{minSampleVal}$  および  $\text{maxSampleVal}$  を、コンテンツのサンプルレンジによって定義された最小および最大サンプル値を示すものとする。
- 両端値を含む、 $0 \sim \text{hdr_recon_num_ranges}[c] - 1$  のレンジ中の  $i$  の場合、変数  $\text{ScaleValue}[c][i]$  は、以下のように導出される。

## 【0185】

## 【数26】

10

```

SignValue[c][i]=0//正について0、負について1
hdrReconScaleBitDepth = hdr_recon_scale_bit_depth -
( hdr_negative_scales_present_flag ? 1 : 0 )
if( hdr_negative_scales_present_flag )
    ScaleValue[ c ][ i ] = hdr_recon_scale_val[ c ][ i ] & ( (1 <<
hdrReconScaleBitDepth ) - 1 )      (D-xx)
    SignValue[ c ][ i ] = hdr_recon_scale_val[ c ][ i ] & ( 1 <<
hdrReconScaleBitDepth )
else
    ScaleValue[ c ][ i ] = hdr_recon_scale_val[ c ][ i ] (D-xx)
    shiftInvScale = 1 << hdrReconScaleBitDepth
    InvScaleValue[ c ][ i ] = ( 1 << (DefaultPrecShift + hdrReconScaleBitDepth)
+ shiftInvScale ) /
    ScaleValue[ c ][ i ]

```

20

## 【0186】

30

- 両端値を含む、 $0 \sim \text{hdr_recon_num_ranges}[c] - 1$  のレンジ中の  $i$  の場合、変数  $\text{RangeValue}[c][i]$  は、以下のように導出される。

## 【0187】

- $\text{hdr_recon_equal_ranges_flag}[c]$  が 0 に等しい場合、以下が適用される。

## 【0188】

## 【数27】

```
RangeValue[ c ][ i ] = hdr_recon_range_val[ c ][ i ] (D-xx)
```

## 【0189】

- 他の場合 ( $\text{hdr_recon_equal_ranges_flag}[c]$  が 1 に等しい)、以下が適用される。

40

## 【0190】

## 【数28】

```

RangeValue[ c ][ i ] = ( ( InputDynamicRangeValue <<
hdr_recon_offset_frac_bit_depth ) +
( ( hdr_recon_num_ranges[ c ] + 1 ) >> 1 ) ) /
hdr_recon_num_ranges[ c ]      (D-xx)

```

## 【0191】

50

ここで、サンプルレンジが 0 ~ 1 に正規化されたとき、InputDynamicRanges[*c*] は 1 に等しい。

- 両端値を含む、0 ~ hdr\_recon\_num\_ranges[*c*] のレンジ中の *i* の場合、変数 InputRanges[*c*][*i*] および OutputRanges[*c*][*i*] は、以下のように導出される。

【0192】

- *i* が 0 に等しい場合、以下が適用される。

【0193】

【数39】

$$\text{OutputRanges}[c][i] = -\text{hdr_recon_global_offset_val}[c] * \quad 10$$

$$\text{InvScaleValue}[c][i-1] \quad (\text{D-xx})$$

【0194】

【数30】

$$\text{InputRanges}[c][i] = 0 \quad (\text{D-xx})$$

【0195】

- 他の場合 (*i* が 0 に等しくない)、以下が適用される。

【0196】

【数31】

$$\text{InputRanges}[c][i] = \text{InputRanges}[c][i-1] + \text{RangeValue}[c][i-1] \quad (\text{D-} \quad 20$$

$$\text{xx})$$

【0197】

【数32】

$$\text{OutputRanges}[c][i] = \text{OutputRanges}[c][i-1] +$$

$$\text{RangeValue}[c][i-1] * \text{InvScaleValue}[c][i-1]$$

$$(\text{D-xx})$$

【0198】

- 両端値を含む、0 ~ hdr\_recon\_num\_ranges[*c*] - 1 のレンジ中の *i* の場合、パラメータ OffsetSetValue[*c*][*i*] は、以下のように導出される。

【0199】

【数33】

$$\text{precOffsetDeltaBits} = \text{DefaultPrecShift} + \text{hdr_recon_scale_frac_bit_depth}$$

$$\text{OffsetValue}[c][i] = \text{InputRanges}[c][i+1] * (1 << \text{precOffsetDeltaBits})$$

-

$$\text{OutputRanges}[c][i+1] * \text{ScaleValue}[c][i-1] \quad (\text{D-} \quad 40$$

$$\text{xx})$$

$$\text{OffsetValue}[c][i] = ((\text{OffsetValue}[c][i] + (1 << (\text{BitDepth} - 1))) >>$$

$$\text{BitDepth}) *$$

$$(\text{maxSampleVal} - \text{minSampleVal})$$

【0200】

- パラメータ *y* = map[*c*] (x) は、以下のように導出される。

【0201】

- 変数 bitDepthDelta は、DefaultPrecShift + hdr\_recon\_offset\_frac\_bit\_depth - BitDepth に等し

〈設定される

- ( $x << \text{bitDepthDelta}$ ) が  $\text{OutputRanges}[c][0]$  よりも小さいかまたはそれに等しい場合、以下が適用される。

【0202】

【数34】

$$y = \text{InputRanges}[c][0] \quad (\text{D-xx})$$

【0203】

- そうではなく、( $x << \text{bitDepthDelta}$ ) が  $\text{OutputRanges}[c][\text{hdr\_recon\_num\_ranges}[c]]$  よりも大きい場合、以下が適用される。

【0204】

【数35】

$$y = \text{InputRanges}[c][\text{hdr\_recon\_num\_ranges}[c]] \quad (\text{D-xx})$$

$$\text{fracBitDepth} = \text{hdr\_recon\_offset\_frac\_bit\_depth}$$

【0205】

- 他の場合、以下が適用される。

【0206】

【数36】

$$\text{fracBitDepth} = \text{DefaultPrecShift} + \text{hdr\_recon\_scale\_frac\_bit\_depth} +$$

$$\text{hdr\_recon\_offset\_frac\_bit\_depth} - \text{BitDepth}$$

$$\text{for}(i = 1; i \leq \text{hdr\_recon\_num\_ranges}[c]; i++)$$

$$\text{if}( \text{OutputRanges}[i - 1] < (x \ll \text{bitDepthDelta}) \&\&$$

$$(x \ll \text{bitDepthDelta}) \leq \text{OutputRanges}[i] ) \{$$

$$\text{rangeBitShift} = \text{DefaultPrecShift} +$$

$$\text{hdr\_recon\_offset\_frac\_bit\_depth} - \text{BitDepth}$$

$$y = (x - \text{minSampleVal}) * \text{ScaleValue}[c][i - 1] * (1 \ll$$

$$\text{rangeBitDepth}) +$$

$$\text{OffsetValue}[c][i - 1] +$$

$$\text{minSampleVal} * (1 \ll \text{fracBitDepth}) \quad (\text{D-xx})$$

}

$$- \quad \text{fracShiftOffset} = 1 \ll (\text{fracBitDepth} - 1)$$

$$y = (y + \text{fracShiftOffset}) \gg \text{fracBitDepth}$$

【0207】

代替的に、 $\text{minSampleVal}$  および  $\text{maxSampleVal}$  に基づくサンプルレンジの調整は、 $\text{OffsetValue}$  に対して実行されず、むしろ以下のように、 $\text{InputRanges}$  および  $\text{OutputRanges}$  に対して実行される。

【0208】

10

20

30

40

## 【数37】

```

deltaSampleVal = maxSampleVal - minSampleVal
deltaBitShift = DefaultPrecShift + hdr_recon_offset_frac_bit_depth
sampleShift = ( 1 << ( BitDepth - 1 ) ) + ( minSampleVal << deltaBitShift )
OutputRanges[ c ][ i ] = ( ( OutputRanges[ c ][ i ] * deltaSampleVal ) +
sampleShift ) >>
                                         BitDepth
                                         10

deltaBitShift = DefaultPrecShift + hdr_recon_offset_frac_bit_depth
sampleShift = ( 1 << ( BitDepth - 1 ) ) + ( minSampleVal << deltaBitShift )
InputRanges[ c ][ i ] = ( ( InputRanges[ c ][ i ] * deltaSampleVal ) + sampleShift )
>>
                                         BitDepth

```

## 【0209】

[0155] 本開示は、S E I シグナリングおよび処理または H . 2 6 5 / H E V C 、 H . 2 6 4 / A V C 、 B D A 、 M P E G または他のものなど、ビデオコーディング規格において指定されるまたは指定されることになる他の手段を使用してコンポーネントスケーリング情報の搬送を改善するためのいくつかの技法を与える。これらの技法のうちの 1 つまたは複数は、独立して、または、他のものとの組合せで適用され得る。さらに、ダイナミックレンジプロセスの固定小数点実装を実行するための S E I メッセージ中の情報をシグナリングおよび / または使用するための上記で説明された技法は、情報をシグナリング / 受信するための以下で説明されるシンタックス構造のうちの 1 つまたは複数を利用し得る。 20

## 【0210】

[0156] いくつかの例では、ビデオエンコーダ 2 0 は、各コンポーネントのために、本開示で説明されるように、ダイナミックレンジ調整を適用する前に、それを下回るすべてのコンポーネント値が第 1 のコンポーネント値にクリッピングされる第 1 の調整されていないコンポーネント値を決定する第 1 のオフセット値を含む、グローバルオフセット値を含む 1 つまたは複数の S E I メッセージをシグナリングし得る。デコーダ 3 0 は、そのような S E I メッセージのうちの 1 つまたは複数を受信し、 S E I メッセージ中の情報をベースおよび / または復号し、ならびにビデオポストプロセッサ 3 1 に情報を受け渡し得る。 30

## 【0211】

[0157] いくつかの例では、各コンポーネントのために、ビデオエンコーダ 2 0 は、第 1 のオフセット値がダイナミックレンジ調整の後にそれにマッピングされる調整された値を指定する、第 2 のオフセット値を含む 1 つまたは複数の S E I メッセージをシグナリングし得る。ビデオデコーダ 3 0 は、そのような S E I メッセージを受信し、情報をベースおよび / または復号し、ビデオポストプロセッサ 3 1 にその情報を受け渡し得る。 40

## 【0212】

[0158] 別の例では、第 1 のグローバルオフセット値も第 2 のグローバルオフセット値も、 S E I メッセージ中でシグナリングされない。代わりに、デコーダ 3 0 は、第 1 のグローバルオフセットおよび第 2 のグローバルオフセットの値が、デコーダ 3 0 がシーケンスごとに決定するかまたは外部手段によって受信するかのいずれかの一定値、所定の値、あるいはシグナリングされた値であると仮定する。別の例では、ビデオエンコーダ 2 0 は S E I メッセージ中で第 1 のグローバルオフセット値をシグナリングするが、第 2 のグローバルオフセット値は S E I メッセージ中でシグナリングされない。代わりに、ビデオデコーダ 3 0 は、その値が、デコーダ 3 0 がシーケンスごとに決定するかまたは外部手段 50

によって受信されたかのいずれかの一定値、所定の値、あるいはシグナリングされた値であると推論する。またさらなる例では、ビデオエンコーダ20はSEIメッセージ中で第2のグローバルオフセット値をシグナリングするが、第1のグローバルオフセット値はSEIメッセージ中でシグナリングされない。代わりに、ビデオデコーダ30は、第1のグローバルオフセット値が、デコーダ30がシーケンスごとに決定するかまたは外部手段によって受信されたかのいずれかの一定値、所定の値、あるいはシグナリングされた値であると推論する。

#### 【0213】

[0159] いくつかの例では、ビデオエンコーダ20は、デコーダ30によって受信され、グローバルオフセット値とローカルスケールおよびオフセット値の両方、ならびに調整されていない値のレンジのパーティション、および調整された値のレンジのパーティションを含む、他のグローバルまたはローカルパラメータを導出するためにデコーダ30によって使用されるオフセット値をシグナリングし得る。 10

#### 【0214】

[0160] いくつかの例では、ビデオエンコーダ20は、入力表現値（すなわち、コンポーネント値）のレンジがダイナミックレンジ調整中に分割されたパーティションの数を含む1つまたは複数のSEIメッセージをシグナリングし得る。一例では、パーティションの数は、2の累乗（すなわち1、2、4、8、16など）であるように制約され得、パーティションの数は、対数としてシグナリングされる（たとえば、8つのパーティションは $3 = 10g_2 8$ としてシグナリングされる）。ビデオデコーダ30は、そのようなSEIメッセージを受信し、情報をペースおよび/または復号し、ビデオポストプロセッサ31にその情報を受け渡し得る。 20

#### 【0215】

[0161] いくつかの例では、クロマコンポーネントのためのパーティションの数は、ルーマコンポーネントのためのパーティションの数とは異なり得る。パーティションの数は、 $2 + 1$ の累乗であるように制約され、対数としてシグナリングされ、マイナス0に向けて丸め得る。このようにして、中間クロマをもつピクセルはそれら自体の値を有することができ、そのパーティションのサイズは他のパーティションよりも小さくなり得る。そのような例では、中間クロマは、中央値の周りのクロマの値（たとえば、クロマ値が、-0.5から0.5の間、または、10ビット表現において-512から511の間にわたるとき、0）を指し得る。2の累乗としてパーティションの数を制約することは、エンコーダ20が整数値のための実際の値よりも少ないビットで値の対数を表すことが可能であり得るので、エンコーダ20がビットを節約することを可能にし得る。 $2 + 1$ の累乗にパーティションの数を制約することは、少なくとも1つのパーティションが中間クロマ値に専用であり得ることを保証し得、いくつかの例では、中間クロマ値に対応するパーティションの幅は、残りよりも小さくなり得る。他の例では、そのようなパーティションは、他のパーティションのうちの1つまたは複数よりも大きくなり得る。 30

#### 【0216】

[0162] いくつかの例では、デコーダ30は、グローバルオフセット値とローカルスケールおよびオフセット値の両方、ならびに調整されていないコンポーネント値のレンジのパーティションの実際のサイズ、および/または調整されたコンポーネント値のレンジのパーティションのサイズを含む、他のグローバルまたはローカルパラメータを導出するためにパーティションのシグナリングされた数を使用し得る。 40

#### 【0217】

[0163] いくつかの例では、エンコーダ20は、各パーティションのために、入力コンポーネント値および対応するマッピングされた出力コンポーネント値のレンジを指定するローカルスケールおよびローカルオフセット値を含み得る1つまたは複数のSEIメッセージをシグナリングし得る。いくつかの例では、エンコーダ20は、スケールおよびオフセットをシグナリングするためにシンタックス要素によって使用されるビット数を含むSEIメッセージをシグナリングし得る。他の例では、エンコーダ20は、シンタックス要 50

素におけるスケールおよびオフセットの小数部分を表すために使用されるビット数を示す S E I メッセージをシグナリングし得る。他の例では、エンコーダ 2 0 は、スケールパラメータ (scale parameter) の整数部分が符号付き表現中でシグナリングされることを示す 1 つまたは複数の S E I メッセージまたはシンタックス要素をシグナリングし得る。いくつかの例では、符号付き表現は 2 の補数である。他の例では、符号付き表現は符号付き絶対値 (signed magnitude) 表現である。ビデオデコーダ 3 0 は、そのような S E I メッセージおよび / またはシンタックス要素を受信し、情報をペースおよび / または復号し、ビデオポストプロセッサ 3 1 にその情報を受け渡し得る。

#### 【 0 2 1 8 】

[0164] 他の例では、エンコーダ 2 0 は、最初に、調整されたコンポーネントまたは表現値のレンジを算出し、次いで、スケール値を使用して、調整されていない表現中の対応するレンジを算出するために、各オフセット値を連続的に使用し得る。たとえば、1 つのオフセット値が使用されて、調整されたコンポーネントのために導出またはシグナリングされたグローバルオフセット値の値を使用して、調整されたコンポーネント中の第 1 のパーティションのレンジを算出し、続いて、スケール値および調整された表現の第 1 のパーティションのレンジを使用して、調整されていない表現の対応するパーティション中のレンジを導出し、調整された表現の第 1 のパーティションと調整されていない表現の対応するパーティションとのそれぞれのレンジを用いて、パーティションの境界を示す調整されたレンジの第 1 のパーティションと調整されていない表現の対応するパーティションとのために導出されたそれぞれの値を導出し得る。これに続いて、別のオフセット値が使用されて、前のステップにおいて導出された調整されたコンポーネント中の第 1 のパーティションの境界値を使用して、調整されたコンポーネント中の第 2 のパーティションのレンジを算出し、続いて、スケール値および調整された表現の第 2 のパーティションのレンジを使用して、調整されていない表現のレンジを導出し得、調整された表現の第 2 のパーティションと調整されていない表現の対応するパーティションとのそれぞれのレンジを用いて、それぞれの表現の境界を示す調整された表現および調整されていない表現中のパーティションのためのそれぞれの値が導出される。この方法は、すべてのレンジおよび境界が、調整された表現および調整されていない表現中のすべてのパーティションのために導出されるまで、繰り返される。別の例では、エンコーダ 2 0 は、最初に、調整されていないコンポーネントまたは表現値のレンジを算出し、次いで、スケール値を使用して、調整された表現中の対応するレンジを算出するために、各オフセット値を連続的に使用し得る。言い換えれば、スケールおよびオフセット値が適用されるコンポーネントまたは表現は、調整されていない表現と調整された表現との間でスワップされ得る。

#### 【 0 2 1 9 】

[0165] いくつかの例では、スケールおよびオフセット値をシグナリングするためにシンタックス要素によって使用されるビット数は、コンポーネントに依存し得る。他の例では、デフォルトビット数が定義され、これらの数が明示的にシグナリングされないときに使用される。

#### 【 0 2 2 0 】

[0166] いくつかの例では、エンコーダ 2 0 は、出力表現 (すなわち、出力コンポーネント) のパーティションの長さが等しいかどうかを示すシンタックス要素をシグナリングし得る。そのような例では、エンコーダ 2 0 は、1 つまたは複数のパーティションのためにオフセット値をシグナリングしないことがある。デコーダ 3 0 は、いくつかの例では、オフセット値が等しいと推論し得る。別の例では、デコーダ 3 0 は、パーティションが等しい長さのものであると仮定し得、そのように示すシンタックス要素を受信しないことがある。いくつかの例では、デコーダ 3 0 は、シグナリングされたシンタックス要素および表現のあらかじめ定義された総動的レンジから各パーティションのサイズを導出し得る。

#### 【 0 2 2 1 】

[0167] 他の例では、各パーティションのためのピボット点ならびに各パーティションのためのスケールおよびオフセット値をシグナリングするのではなく、ビデオエンコーダ

10

20

30

30

40

50

20は、1つまたは複数またはすべてのパーティションのサイズとともに、各パーティションのための導関数またはスケール値を示す1つまたは複数のSEIメッセージをシグナリングし得る。この手法は、エンコーダ20が、各パーティションのためのローカルオフセット値をシグナリングすることを回避することを可能にし得る。代わりに、いくつかの例では、エンコーダ20は、1つまたは複数のSEIメッセージ中で、1つまたは複数のパーティションのためのパーティションサイズおよびスケール値（または導関数）をシグナリングすることが可能であり得る。（より高い正確さを必要とし得る）各パーティションまたは区分のためのローカルオフセット値は、デコーダ30によって決定または導出され得る。

## 【0222】

10

[0168] いくつかの例では、エンコーダ20は、いくつかのパーティションのためのオフセットおよびスケール値のためのいくつかのデフォルト値を指定するモード値を示す1つまたは複数のSEIメッセージをシグナリングし得る。ビデオデコーダ30は、そのようなSEIメッセージを受信し、情報をパースおよび／または復号し、ビデオポストプロセッサ31にその情報を受け渡し得る。

## 【0223】

[0169] いくつかの例では、エンコーダ20は、コンポーネントのサブセットの持続性が定義され得、コンポーネントのサブセットのコンポーネントスケール値が更新され得るように、SEIメッセージの持続性を定義する値を示す1つまたは複数のSEIメッセージをシグナリングし得る。SEIメッセージの持続性は、SEIのインスタンスにおいてシグナリングされた値が適用され得るピクチャを示す。いくつかの例では、SEIメッセージの持続性は、SEIメッセージの1つのインスタンスにおいてシグナリングされた値が、SEIメッセージが適用されるピクチャのすべてのコンポーネントに対応して適用され得るように定義される。他の例では、SEIメッセージの持続性は、SEIの1つのインスタンスにおいてシグナリングされた値が、コンポーネントのサブセットに対応して適用されるように示され得るように、定義され、ここにおいて、SEIのインスタンスにおける値が適用されないコンポーネントは、適用可能な値を有しないことがあるか、または、SEIメッセージの別のインスタンスにおいてシグナリングされる適用可能な値を有し得るかのいずれかである。ビデオデコーダ30は、そのようなSEIメッセージを受信し、情報をパースおよび／または復号し、ビデオポストプロセッサ31にその情報を受け渡し得る。

20

## 【0224】

30

[0170] いくつかの例では、エンコーダ20は、復号された出力に実行されるべき後処理ステップを示すシンタックス要素を含む1つまたは複数のSEIメッセージをシグナリングし得る。各シンタックス要素は、特定のプロセス（たとえばコンポーネントをスケーリングすること、色変換、アップサンプリング／ダウンサンプリングフィルタなど）に関連し得、シンタックス要素の各値は、プロセスに関連する特定のパラメータのセットが使用されることを指定し得る。いくつかの例では、プロセスに関連するパラメータは、ビットストリームの一部であるSEIメッセージを使用して、または他の手段を通して送信され得るメタデータとして、ビデオエンコーダ20によってシグナリングされる。ビデオデコーダ30は、そのようなSEIメッセージを受信し、情報をパースおよび／または復号し、ビデオポストプロセッサ31にその情報を受け渡し得る。

40

## 【0225】

[0171] いくつかの例では、エンコーダ20は、入力表現（すなわち、入力コンポーネント値）を出力表現（すなわち、出力コンポーネント値）にマッピングするための区分線形モデル関数を記述するおよび／またはそれを構築するために使用され得るシンタックス要素または1つまたは複数のSEIメッセージをシグナリングし得る。ビデオデコーダ30は、そのようなSEIメッセージを受信し、情報をパースおよび／または復号し、ビデオポストプロセッサ31にその情報を受け渡し得る。他の例では、あらかじめ定義された仮定が、入力表現を出力表現にマッピングするための区分線形モデル関数を記述するおよ

50

び / または構築するために使用され得る。

【 0 2 2 6 】

[0172] いくつかの例では、エンコーダ 2 0 は、S E I メッセージ中でシグナリングされるスケールおよびオフセットパラメータが、第 2 のコンポーネントの異なる値の関数として第 1 のコンポーネントに適用されるべきスケールの変動を表すことを示す 1 つまたは複数のシンタックス要素を含み得る 1 つまたは複数の S E I メッセージをシグナリングし得る。

【 0 2 2 7 】

[0173] いくつかの例では、エンコーダ 2 0 は、第 2 のコンポーネントの異なる値の関数として第 1 のコンポーネントに対してスケールとともに適用されるべきであるかまたは適用され得るオフセットパラメータを示す 1 つまたは複数の S E I メッセージをシグナリングし得る。いくつかの例では、エンコーダ 2 0 は、第 2 のコンポーネントの異なる値の関数として第 1 のコンポーネントに対してスケールとともに適用されるべきであるかまたは適用され得るオフセットパラメータを示す 1 つまたは複数の追加のシンタックス要素を含み得る 1 つまたは複数の S E I メッセージをシグナリングし得る。ビデオデコーダ 3 0 は、そのような S E I メッセージを受信し、情報をパースおよび / または復号し、ビデオポストプロセッサ 3 1 にその情報を受け渡し得る。

10

【 0 2 2 8 】

[0174] いくつかの例では、エンコーダ 2 0 は、デコーダ側で使用される電気光学伝達関数特性が電気光学伝達関数特性の第 1 のセットと同様であるとき、S E I メッセージシグナリングされたスケール、オフセット、および他のダイナミックレンジ調整パラメータが適用されるように、その電気光学伝達関数特性の第 1 のセットを示す第 1 のシンタックス要素を含む 1 つまたは複数の S E I メッセージをシグナリングし得る。

20

【 0 2 2 9 】

[0175] 別の例では、エンコーダ 2 0 は、電気光学伝達関数特性の第 1 のセットまたは同様の特性をもつものがデコーダ 3 0 によって使用されるとき、( 1 つまたは複数の ) S E I メッセージ中のシグナリングされたオフセット、スケールおよび他のダイナミックレンジパラメータが、H D R 出力の最良の再構成のために適用されるべきであることを示す 1 つまたは複数の S E I メッセージをシグナリングし得る。ビデオデコーダ 3 0 は、そのような S E I メッセージを受信し、情報をパースおよび / または復号し、ビデオポストプロセッサ 3 1 にその情報を受け渡し得る。

30

【 0 2 3 0 】

[0176] 別の例では、エンコーダ 2 0 は、対応する逆電気光学伝達関数特性がデコーダ側において適用されるとき、光電子伝達関数特性の第 1 のセット、およびシグナリングされたスケール、オフセットならびに他のダイナミックレンジ調整パラメータが、デコーダ 3 0 によって適用されることを示す 1 つまたは複数の S E I メッセージをシグナリングし得る。ビデオデコーダ 3 0 は、そのような S E I メッセージを受信し、情報をパースおよび / または復号し、ビデオポストプロセッサ 3 1 にその情報を受け渡し得る。

【 0 2 3 1 】

[0177] 他の例では、エンコーダ 2 0 は、電気光学 / 光電子特性の異なるセットを示す 2 つ以上の S E I メッセージが存在し、現在ピクチャ適用可能であるとき、1 つの S E I メッセージのみが適用されるような状況をシグナリングし得る。エンコーダは、異なるタイプのデコーダまたは異なる能力をもつデコーダを満たすために、電気光学 / 光電子特性の異なるセットをシグナリングし得る。たとえば、デコーダ側におけるいくつかのディスプレイは、適切な領域中のコード化コンポーネント値を線形光にコンバートするために、P T E O T F を適用し得るが、他のディスプレイ、たとえばレガシーディスプレイは、線形光にコンバートするためにガンマ E O T F を適用し得る。エンコーダが送る特定の特性をもつ各 S E I は、いくつかのタイプのディスプレイに適するかまたはそれに有益であり得、他のタイプのディスプレイに適さないかまたはそれに有益でないことがあり、たとえば、P Q E O T F 特性をもつ S E I メッセージが、コーディングされたビデオを線形

40

50

光にコンバートするために P Q E O T F を適用するディスプレイに好適であり得る。デコーダ 3 0 は、どの S E I メッセージが適用されるべきであるかを決定し、適用規格に基づいて、エンドユーザデバイスに基づいて、受信される信号に基づいて、または外部手段を通して受信された別の指示に基づいてそのような選定を行う。たとえば、デコーダ 3 0 は、現在ピクチャに適用される第 1 の S E I メッセージ中の第 1 のシンタックス要素が、S E I メッセージが P Q O E T F の逆を用いて適用されるべきであることを示し、現在ピクチャに適用される第 2 の S E I メッセージ中の第 1 のシンタックス要素が、S E I メッセージが ( B B C 、または P H などの ) 別の伝達関数を用いて適用されるべきであることを示すと決定し得、デコーダ 3 0 またはエンドユーザデバイスは、デバイスが P Q E O T F を使用するので、第 1 の S E I メッセージ中のパラメータを適用することを選定し得る。いくつかの例では、デコーダが準拠する適用規格が、特性の特定のセットをもつ S E I メッセージが使用されるべきであることを指定し得る。  
10

#### 【 0 2 3 2 】

[0178] 他の例では、エンコーダ 2 0 は、伝達特性の複数のセットに対応するパラメータを搬送する S E I メッセージをシグナリングし得る。他の例では、エンコーダ 2 0 は、その目的で異なる S E I メッセージをシグナリングし得る。ビデオデコーダ 3 0 は、そのような S E I メッセージを受信し、情報をパースおよび / または復号し、ビデオポストプロセッサ 3 1 にその情報を受け渡し得る

[0179] いくつかの例では、エンコーダ 2 0 は、S E I メッセージの適用可能性を示すシンタックス要素を含む 1 つまたは複数の S E I メッセージをシグナリングし得る。S E I メッセージの適用可能性は、限定はしないが、( 1 ) スケールおよびオフセットが適用されるコンポーネント、( 2 ) コンポーネントスケーリングが適用される位置、ならびに / または ( 3 ) 追加のスケーリングパラメータがシグナリングされるかどうかを含み得る。  
20

#### 【 0 2 3 3 】

[0180] 説明されたように、エンコーダ 2 0 は、スケールおよびオフセットが適用されるコンポーネントを示すシンタックス要素を含む 1 つまたは複数の S E I メッセージをシグナリングし得る。以下は、そのような適用例のいくつかの例を記載する。たとえば、シンタックス要素の 1 つの値は、第 1 のコンポーネントインデックスのためのシグナリングされたパラメータが、R G B コンポーネントに適用されるべきであることを示し得る。別の値は、第 1 のコンポーネントインデックスのためのシグナリングされたパラメータが、ルーマコンポーネントに適用されるべきであり、第 2 のインデックスおよび第 3 のインデックスのためのシグナリングされたパラメータが、C b および C r コンポーネントに適用されるべきであることを示し得る。別の値は、第 1 のコンポーネントインデックスのためのシグナリングされたパラメータが、R 、 G および B コンポーネントに適用されるべきであり、第 2 のインデックスおよび第 3 のインデックスのためのシグナリングされたパラメータが、C b および C r コンポーネントに適用されるべきであることを示し得る。別の値は、最初の 3 つのインデックスのためのシグナリングされたパラメータが、ルーマ、 C b および C r コンポーネントに適用され、残りのインデックスに対応するシグナリングされたパラメータが、色補正のために適用されることを示し得る。ビデオデコーダ 3 0 は、そのような S E I メッセージを受信し、情報をパースおよび / または復号し、ビデオポストプロセッサ 3 1 にその情報を受け渡し得る。  
30 40

#### 【 0 2 3 4 】

[0181] また説明されたように、エンコーダ 2 0 は、コンポーネントスケーリングが適用される位置を示すシンタックス要素を含む 1 つまたは複数の S E I メッセージをシグナリングし得る。ビデオの復号の後に、およびビデオポストプロセッサ 3 1 中で、デコーダ側でいくつかのプロセスが行われる。S E I に関連するプロセスが適用されるべきである位置を示すシンタックス要素のシグナリング、言い換えれば、S E I 中の情報を使用することに関連するプロセスの先行するまたは後続の動作のサブセットの指示は、ビデオを処理するために、ビデオデコーダ 3 0 またはビデオポストプロセッサ 3 1 にとって役立つこ  
50

とになる。たとえば、そのようなシンタックス要素は、たとえば、アップサンプリングの前または後に Y C b C r コンポーネントにコンポーネントスケーリングが適用される位置を示し得る。別の例では、シンタックス要素は、コンポーネントスケーリングがデコーダ側でない量子化の前に適用されることを示し得る。ビデオデコーダ 30 は、そのような S E I メッセージを受信し、情報をパースおよび / または復号し、ビデオポストプロセッサ 31 にその情報を受け渡し得る。

#### 【 0 2 3 5 】

[0182] また説明されたように、エンコーダ 20 は、たとえば色補正のために、スケーリングおよびパラメータの追加のセットがシグナリングされるかどうかを示すシンタックス要素を含む 1 つまたは複数の S E I メッセージをシグナリングし得る。パラメータの追加のセットは、特定の色域を適合させるために、色コンポーネントをマッピングするための色補正のために、または、V U I 中の transfer\_characteristics シンタックス要素によって示された伝達関数とは異なる伝達関数が適用されるとき、コンポーネント値の補正のために使用され得る。 10

#### 【 0 2 3 6 】

[0183] 他の例では、エンコーダ 20 は、上記の態様を示すために、異なるシンタックス要素、たとえば、S E I がどの ( 1 つまたは複数の ) コンポーネントに適用されるかを示す 1 つのシンタックス要素、S E I が S D R 互換コンテンツの H D R 互換に適用されるかどうかを示す 1 つのシンタックス要素、およびコンポーネントスケーリング S E I メッセージが適用されるべきである ( 1 つまたは複数の ) 位置を示す 1 つのシンタックス要素をシグナリングし得る。 20

#### 【 0 2 3 7 】

[0184] コンポーネントスケーリング S E I メッセージパラメータが適用されるコンポーネントの数が 2 つ以上であるとき、エンコーダ 20 は、スケールおよびオフセットパラメータの適用がコンポーネントのインデックスに基づいて連続的に行われ得ることを示すシンタックス要素を含む 1 つまたは複数の S E I メッセージをシグナリングし得る。たとえば、第 1 のコンポーネントのスケールおよびオフセットパラメータに基づくマッピングが適用され得、次いで、たとえば、第 2 のコンポーネントのためにシグナリングされたスケールおよびオフセットを使用する第 2 のコンポーネントのマッピングが、第 1 のコンポーネントの値に依存し得る。いくつかの例では、これは、たとえば、第 1 のコンポーネントのマッピングされた値が使用されるべきであることを指定するシンタックス要素によって示される。ビデオデコーダ 30 は、そのような S E I メッセージを受信し、情報をパースおよび / または復号し、ビデオポストプロセッサ 31 にその情報を受け渡し得る。 30

#### 【 0 2 3 8 】

[0185] 別の例では、ビデオエンコーダ 20 は、S E I 後処理が適用されない場合でも、H D R 1 0 受信機が閲覧可能な H D R ビデオを復号し、見せることができるように、1 つまたは複数の S E I メッセージ中で、または、ビットストリーム中でシグナリングされる値を制約し得る。 ( 1 つまたは複数の ) S E I メッセージは、そうあること ( たとえば、ビットストリームが H D R 1 0 後方互換性があるビットストリームであること ) を示すためのシンタックス要素を含み得る。 40

#### 【 0 2 3 9 】

[0186] このセクションは、本開示の 1 つまたは複数の態様によって開示される技法を使用するいくつかの例を含む。

例 1

[0187] この例 1 では、コンポーネントスケーリング関数がルックアップテーブルとしてシグナリングされ、ルックアップテーブルを定義する点をシグナリングするために使用されるビット数もシグナリングされる。シグナリングされる明示的点を有しないサンプル値のために、値は隣接ピボット点に基づいて補間される。

#### 【 0 2 4 0 】

【表4】

コンポーネントスケーリングSEIメッセージのシンタックス

component_scale_info( payloadSize ) {	記述子
<b>comp_scale_id</b>	ue(v)
<b>comp_scale_cancel_flag</b>	u(1)
if( !comp_scale_cancel_flag ) {	
<b>comp_scale_persistence_flag</b>	u(1)
<b>comp_scale_num_comps_minus1</b>	ue(v)
<b>comp_scale_input_bit_depth</b>	ue(v)
<b>comp_scale_output_bit_depth</b>	ue(v)
for( c = 0; c <= comp_scale_num_comps_minus1; c++ ) {	
<b>comp_scale_num_points_minus1[ c ]</b>	ue(v)
for( i = 0; i <= comp_scale_num_points_minus1[ c ]; i++ ) {	
<b>comp_scale_input_point[ c ][ i ]</b>	u(v)
<b>comp_scale_output_point[ c ][ i ]</b>	u(v)
}	
}	
}	
}	

## 【0241】

コンポーネントスケーリングSEIメッセージのセマンティクス

コンポーネントスケーリングSEIメッセージは、復号ピクチャの様々なコンポーネントに対してスケーリング動作を実行するための情報を与える。スケーリング動作が実行されるべきである色空間およびコンポーネントは、SEIメッセージ中でシグナリングされるシンタックス要素の値によって決定される。

**comp\_scale\_id**は、コンポーネントスケーリングSEIメッセージの目的を識別するために使用され得る識別番号を含んでいる。**comp\_scale\_id**の値は、両端値を含む、 $0 \sim 2^{32} - 2$ のレンジ中にあるものとする。**comp\_scale\_id**の値は、コンポーネントスケーリングSEIメッセージ色空間を指定する、または、コンポーネントスケーリングSEIメッセージが線形領域中で適用されるのか、非線形領域中で適用されるのかを指定するために使用され得る。

両端値を含む、 $0 \sim 255$ の、および、両端値を含む、 $512 \sim 2^{31} - 1$ の**comp\_scale\_id**の値は、適用によって決定されたように使用され得る。両端値を含む、 $256 \sim 511$ の、および、両端値を含む、 $2^{31} \sim 2^{32} - 2$ の**comp\_scale\_id**の値は、ITU-T | ISO / IECによる将来の使用のために予約される。デコーダは、両端値を含む、 $256 \sim 511$ のレンジ中の、または、両端値を含む、 $2^{31} \sim 2^{32} - 2$ のレンジ中の**comp\_scale\_id**の値を含んでいるすべてのコンポーネントスケール情報SEIメッセージを無視するものとし、ビットストリームは、そのような値を含んでいないものとする。

注1 - **comp\_scale\_id**は、異なるディスプレイシナリオに好適であるコンポーネントスケーリングプロセスをサポートするために使用され得る。たとえば、**comp\_scale\_id**の異なる値は、スケーリングが適用される異なるディスプレイビット深度または異なる色空間に対応し得る。

10

20

30

40

50

代替的に、`comp_scale_id` はまた、スケーリングがいくつかのタイプのディスプレイまたはデコーダ、たとえば HDR、SDR への互換性のために実行されるかどうかを識別するために使用され得る。

1 に等しい `comp_scale_cancel_flag` は、コンポーネントスケーリング情報 SEI メッセージが、現在レイヤに適用される出力順序で前のコンポーネント情報 SEI メッセージの持続性を消去することを示す。0 に等しい `comp_scale_cancel_flag` は、コンポーネントスケーリング情報が続くことを示す。

`comp_scale_persistence_flag` は、現在レイヤのためのコンポーネントスケーリング情報 SEI メッセージの持続性を指定する。

0 に等しい `comp_scale_persistence_flag` は、コンポーネントスケーリング情報が現在復号ピクチャのみに適用されることを指定する。

`picA` を現在ピクチャとする。1 に等しい `comp_scale_persistence_flag` は、以下の条件のいずれかが真になるまで、コンポーネントスケーリング情報が出力順序で現在レイヤのために持続することを指定する。

- 現在レイヤの新しい CLVS が開始する。
- ビットストリームが終了する。

- `comp_scale_id` の同じ値をもち、現在レイヤに適用可能な、コンポーネントスケーリング情報 SEI メッセージを含んでいるアクセスユニット中の現在レイヤ中のピクチャ `picB` が出力され、それについて、`PicOrderCnt(picB)` が `PicOrderCnt(picA)` よりも大きく、ここで、`PicOrderCnt(picB)` および `PicOrderCnt(picA)` が、それぞれ、`picB` のためのピクチャ順序カウントのための復号プロセスの呼出しの直後の `picB` および `picA` の `PicOrderCntVal` 値である。

`comp_scale_num_comps_minus1 + 1` は、コンポーネントスケーリング関数が指定されるコンポーネントの数を指定する。`comp_scale_num_comps_minus1` は、両端値を含む、0 ~ 2 のレンジ中にあるものとする。

`comp_scale_num_comps_minus1` が 2 よりも小さく、`c` 番目のコンポーネントのコンポーネントスケーリングパラメータがシグナリングされないとき、`(c - 1)` 番目のコンポーネントのコンポーネントスケーリングパラメータに等しいと推論される。

代替的に、`comp_scale_num_comps_minus1` が 2 よりも小さく、`c` 番目のコンポーネントのコンポーネントスケーリングパラメータがシグナリングされないとき、`c` 番目のコンポーネントのコンポーネントスケーリングパラメータは、効果的にそのコンポーネントのスケーリングがないように、デフォルト値に等しいと推論される。

代替的に、コンポーネントスケーリングパラメータの推論は、SEI メッセージが適用される色空間に基づいて指定され得る。

- 色空間が YCbCr であり、`comp_scale_num_comps_minus1` が 1 に等しいとき、コンポーネントスケーリングパラメータは Cb コンポーネントと Cr コンポーネントの両方に適用される。

- 色空間が YCbCr であり、`comp_scale_num_comps_minus1` が 2 に等しいとき、第 1 および第 2 のコンポーネントスケーリングパラメータが Cb および Cr コンポーネントに適用される。

一代替形態では、異なる推論は、`comp_scale_id` の値に基づいてまたは明示的シンタックス要素に基づいて指定される。

代替的に、制約が以下のように加えられる。

`comp_scale_num_comps_minus1` の値が、CLVS 内の `comp_scale_id` の所与の値をもつすべてのコンポーネントスケーリング SEI メッセージについて同じであるとすることが、ビットストリームコンフォーマンスのための制約である。

10

20

30

40

50

`comp_scale_input_bit_depth_minus8 + 8` は、シンタックス要素 `comp_scale_input_point[c][i]` をシグナリングするために使用されるビット数を指定する。`comp_scale_input_bit_depth_minus8` の値は、両端値を含む、0 ~ 8 のレンジ中にあるものとする。

コンポーネントスケーリング SEI メッセージがレンジ 0 . 0 ~ 1 . 0 中の正規化された浮動小数点表現中にある入力に適用されるとき、SEI メッセージは、入力ビデオを、`colour_remap_input_bit_depth_minus8 + 8` に等しいビット深度をもつコンバートされたビデオ表現にコンバートするために実行される量子化動作の仮想結果を指す。

コンポーネントスケーリング SEI メッセージが、`comp_scale_input_bit_depth_minus8 + 8` に等しくないビット深度を有する入力に適用されるとき、SEI メッセージは、入力ビデオ表現を、`colour_remap_input_bit_depth_minus8 + 8` に等しいビット深度をもつコンバートされたビデオ表現にコンバートするために実行されるトランスコーディング動作の仮想結果を指す。

`comp_scale_output_bit_depth_minus8 + 8` は、シンタックス要素 `comp_scale_output_point[c][i]` をシグナリングするために使用されるビット数を指定する。`comp_scale_output_bit_depth_minus8` の値は、両端値を含む、0 ~ 8 のレンジ中にあるものとする。

コンポーネントスケーリング SEI メッセージが浮動小数点表現中にある入力に適用されるとき、SEI メッセージは、コンポーネントスケーリング SEI メッセージの処理の後に取得される `colour_remap_output_bit_depth_minus8 + 8` に等しいビット深度をもつビデオ表現をレンジ 0 . 0 ~ 1 . 0 中の浮動小数点表現にコンバートするために実行される逆量子化動作の仮想結果を指す。

代替的に、`comp_scale_input_point[c][i]` および `comp_scale_output_point[c][i]` をシグナリングするために使用されるビット数は、それぞれ、`comp_scale_input_bit_depth` および `comp_scale_output_bit_depth` として、または言い換えると、8 を減算することなしにシグナリングされる。

`comp_scale_num_points_minus1[c] + 1` は、コンポーネントスケーリング関数を定義するために使用されるピボット点の数を指定する。`comp_scale_num_points_minus1[c]` は、両端値を含む、0 ~ (1 << `Min(comp_scale_input_bit_depth_minus8 + 8, comp_scale_output_bit_depth_minus8 + 8)`) - 1 のレンジ中にあるものとする。

`comp_scale_input_point[c][i]` は、入力ピクチャの `c` 番目のコンポーネントの `i` 番目のピボット点を指定する。`comp_scale_input_point[c][i]` の値は、両端値を含む、0 ~ (1 << `comp_scale_input_bit_depth_minus8[c] + 8`) - 1 のレンジ中にあるものとする。`comp_scale_input_point[c][i]` の値は、両端値を含む、1 ~ `comp_scale_points_minus1[c]` のレンジ中の `i` の場合、`comp_scale_input_point[c][i - 1]` の値よりも大きいかまたはそれに等しくなるものとする。

`comp_scale_output_point[c][i]` は、出力ピクチャの `c` 番目のコンポーネントの `i` 番目のピボット点を指定する。`comp_scale_output_point[c][i]` の値は、両端値を含む、1 ~ (1 << `comp_scale_output_bit_depth_minus8[c] + 8`) - 1 のレンジ中にあるものとする。`comp_scale_output_point[c][i]` の値は、

10

20

30

40

50

両端値を含む、`1 ~ comp_scale_points_minus1[c]` のレンジ中の `i` の場合、`comp_scale_output_point[c][i - 1]` の値よりも大きいかまたはそれに等しくなるものとする。

入力および出力の両方のためのサンプル値が、それぞれ、両端値を含む、`0 ~ (1 << comp_scale_input_bit_depth_minus8[c] + 8) - 1`、および、両端値を含む、`0 ~ (1 << comp_scale_output_bit_depth_minus8[c] + 8) - 1` のレンジ中にある、入力信号表現 `x` および出力信号表現 `y` をマッピングするプロセスは、以下のように指定される。

【0242】

【数38】

```
if( x <= comp_scale_input_point[c][0])
    y = comp_scale_output_point[c][0]
else if( x > comp_scale_input_point[c][comp_scale_input_point_minus1[c]])
    y = comp_scale_output_point[c][comp_scale_output_point_minus1[c]]
else
    for( i = 1; i <= comp_scale_output_point_minus1[c]; i++)
        if( comp_scale_input_point[i - 1] < x && x <=
comp_scale_input_point[i])
```

10

`y = ((comp_scale_output_point[c][i] - comp_scale_output_point[c][i - 1]) /`

20

`(comp_scale_input_point[c][i] - comp_scale_input_point[c][i - 1])) *  
(x - comp_scale_input_point[c][i - 1]) +  
(comp_scale_output_point[c][i - 1]))`

30

【0243】

[0188] 一代替形態では、入力ピボット点 `comp_scale_input_point[c][i]` および出力ピボット点 `comp_scale_output_point[c][i]` は、隣接する値の差分、たとえば、`delta_comp_scale_input_point[]` および `delta_comp_scale_output_point[]` としてコーディングされ、シンタックス要素は、指数ゴロムコードを使用してコーディングされる。

別の代替形態では、入力および出力表現値をマッピングするプロセスは、限定はしないが、スプラインおよび3次補間を含む他の補間方法によって指定される。

【0244】

例2

[0189] この例2は、例1で説明されたSEIシンタックス構造と比較して異なるシンタックス構造を示す。このシンタックス構造では、マッピング関数は、ピボット点の代わりにスケールおよびオフセットに関して説明される。

【0245】

40

【表 5】

コンポーネントスケーリングSEIメッセージのシンタックス

component_scale_info( payloadSize ) {	記述子
<b>comp_scale_id</b>	ue(v)
<b>comp_scale_cancel_flag</b>	u(1)
if( !comp_scale_cancel_flag ) {	
<b>comp_scale_persistence_flag</b>	u(1)
<b>comp_scale_num_comps</b>	ue(v)
<b>comp_scale_input_bit_depth</b>	ue(v)
<b>comp_scale_output_bit_depth</b>	ue(v)
<b>comp_scale_bit_depth_scale_val</b>	ue(v)
<b>comp_scale_log2_denom_scale_val</b>	ue(v)
for( c = 0; c < comp_scale_num_comps; c++ ) {	
<b>comp_scale_num_points_minus1[ c ]</b>	ue(v)
<b>comp_scale_global_offset_input_val[ c ]</b>	u(v)
<b>comp_scale_global_offset_output_val[ c ]</b>	u(v)
for( i = 0; i < comp_scale_num_points_minus1[ c ]; i++ ) {	
<b>comp_scale_offset_val[ c ][ i ]</b>	u(v)
<b>comp_scale_val[ c ][ i ]</b>	u(v)
}	
}	
}	
}	

【0 2 4 6】

**comp\_scale\_bit\_depth\_scale\_val** は、シンタックス要素 **comp\_scale\_val[ c ][ i ]** をシグナリングするために使用されるビット数を指定する。**comp\_scale\_bit\_depth\_scale\_val** の値は、両端値を含む、0 ~ 24 のレンジ中にあるものとする。

**comp\_scale\_log2\_denom\_scale\_val** は、スケール値の底2の分母を指定する。**comp\_scale\_log2\_denom\_scale\_val** の値は、両端値を含む、0 ~ 16 のレンジ中にあるものとする。

**comp\_scale\_global\_offset\_input\_val[ c ] + 1** は、コンポーネントスケーリング関数を定義するために使用される、それを下回るすべての入力表現値が **CompScaleOffsetOutputVal[ c ][ 0 ]** にクリッピングされる入力サンプル値を指定する。**comp\_scale\_num\_points\_minus1[ c ]** は、両端値を含む、0 ~ (1 << **comp\_scale\_input\_bit\_depth**) - 1 のレンジ中にあるものとする。**comp\_scale\_global\_offset\_input\_val[ c ]** を表すために使用されるビット数は、**comp\_scale\_input\_bit\_depth** である。

**comp\_scale\_global\_offset\_output\_val[ c ] + 1** は、**comp\_scale\_global\_offset\_input\_val[ c ]** を下回るすべての入力表現値がクリッピングされるべきである出力サンプル値を指定する。

10

20

30

40

50

`comp_scale_num_points_minus1[c]` は、両端値を含む、  
 $0 \sim (1 << \text{comp\_scale\_output\_bit\_depth}) - 1$  のレンジ  
 中にあるものとする。`comp_scale_global_offset_output_val[c]` を表すために使用されるビット数は、`comp_scale_output_bit_depth` である。

`comp_scale_num_points_minus1[c] + 1` は、コンポーネントスケーリング関数を定義するために使用されるピボット点の数を指定する。`comp_scale_num_points_minus1[c]` は、両端値を含む、 $0 \sim (1 << \text{Min}(\text{comp\_scale\_input\_bit\_depth}, \text{comp\_scale\_output\_bit\_depth})) - 1$  のレンジ中にあるものとする。 10

両方のためのサンプル値、入力表現が、両端値を含む、 $0 \sim (1 << \text{comp\_scale\_input\_bit\_depth}) - 1$  のレンジ中にあり、出力表現が、両端値を含む、 $0 \sim (1 << \text{comp\_scale\_output\_bit\_depth}) - 1$  のレンジ中にある入力信号表現  $x$  および出力信号表現  $y$  をマッピングするプロセスは、以下のように指定される。

【0247】

【数39】

```
if( x <= CompScaleOffsetInputVal[c][0] )
    y = CompScaleOffsetOutputVal[c][0]
else if( x > CompScaleOffsetInputVal[c][comp_scale_output_point_minus1] )
    y = CompScaleOffsetOutputVal[c][comp_scale_output_point_minus1]
else
    for( i = 1; i <= comp_scale_output_point_minus1; i++ )
        if( CompScaleOffsetInputVal[i - 1] < x && x <=
CompScaleOffsetInputVal[i] )
            y = ( x -
CompScaleOffsetInputVal[i - 1] * ( comp_scale_val[c][i] +
CompScaleOffsetOutputVal[c][i] ) 20
30
```

【0248】

`comp_scale_offset_set_val[c][i]` は、 $c$  番目のコンポーネントの  $i$  番目のサンプル値領域のオフセット値を指定する。`comp_scale_offset_set_val[c]` を表すために使用されるビット数は、`comp_scale_input_bit_depth` に等しい。

`comp_scale_val[c][i]` は、 $c$  番目のコンポーネントの  $i$  番目のサンプル値領域点のスケール値を指定する。`comp_scale_val[c]` を表すために使用されるビット数は、`comp_scale_bit_depth_scale_val` に等しい。 40

両端値を含む、 $0 \sim \text{comp\_scale\_num\_points\_minus1}[c]$  のレンジ中の  $i$  の場合の変数 `CompScaleOffsetOutputVal[c][i]` および `CompScaleOffsetInputVal[c][i]` は、以下のように導出される。

【0249】

## 【数40】

```

roundingOffset = (comp_scale_log2_denom_scale_val == 0) ? 0 : (1 <<
                comp_scale_log2_denom_scale_val - 1)
for( i = 0; i <= comp_scale_num_points_minus1[ c ]; i++ )
    if( i == 0 )
        CompScaleOffsetOutputVal[ c ][ i ] =
            comp_scale_global_offset_output_val[ c ]
        CompScaleOffsetInputVal[ c ][ i ] =
            comp_scale_global_offset_input_val[ c ]10
    else
        CompScaleOffsetOutputVal[ c ][ i ] = CompScaleOffsetOutputVal[ c ][ i -
1 ] +20
            (comp_scale_offset_val[ c ][ i - 1 ] << comp_scale_val[ c ][ i - 1 ]
            + roundingOffset ) >>
        comp_scale_log2_denom_scale_val
        CompScaleOffsetInputVal[ c ][ i ] = CompScaleOffsetInputVal[ c ][ i - 1 ]
        +
            comp_scale_offset_val[ c ][ i - 1 ]

```

## 【0250】

一代替形態では、comp\_scale\_offset\_val[ c ][ i ]が、以下のように、0 ~ comp\_scale\_num\_points\_minus1[ c ]のレンジ中の i の場合の、CompScaleOffsetOutputVal[ ][ i ]を直接計算するために、およびCompScaleOffsetInputVal[ ][ i ]を間接的に計算するために使用される。30

## 【0251】

## 【数4 1】

```

for( i = 0; i < comp_scale_num_points_minus1[ c ]; i++ )
    if( i == 0 )
        CompScaleOffsetOutputVal[ c ][ i ] =
            comp_scale_global_offset_output_val[ c ]
        CompScaleOffsetInputVal[ c ][ i ] =
            comp_scale_global_offset_input_val[ c ]
    else
        CompScaleOffsetInputVal[ c ][ i ] = CompScaleOffsetInputVal[ c ][ i - 1 ]
        (comp_scale_offset_val[ c ][ i - 1 ] *
            comp_scale_val[ c ][ i - 1 ]
        + roundingOffset ) >>
        comp_scale_log2_denom_scale_val )
        CompScaleOffsetOutputVal[ c ][ i ] = CompScaleOffsetOutputVal[ c ][ i -
    1 ] +
        comp_scale_offset_val[ c ][ i - 1 ]

```

## 【0252】

一代替形態では、両端値を含む、`0 ~ comp_scale_num_points_minus1[ c ]` のレンジ中の `i` の場合の `comp_scale_offset_val[ c ][ i ]` はシグナリングされず、`comp_scale_offset_val[ c ][ i ]` の値は、スケールが指定される、`comp_scale_num_points_minus1[ c ]` の等間隔に離間した間隔に基づいて導出される。両端値を含む、`0 ~ comp_scale_num_points_minus1[ c ] - 1` のレンジ中の `i` の場合の `comp_scale_offset_val[ c ][ i ]` の値は、以下のように導出される。

## 【0253】

## 【数4 2】

```

comp_scale_offset_val[ c ][ i ] = ((1 << comp_scale_output_bit_depth ) -
    comp_scale_global_offset_output_val[ c ]) ÷
    (comp_scale_num_points_minus1[ c ])

```

## 【0254】

別の代替形態では、`0 ~ comp_scale_num_points_minus1[ c ]` のレンジ中の `i` の場合の `comp_scale_offset_val[ c ][ i ]` は、以下のように計算される。

## 【0255】

## 【数4 3】

```

comp_scale_offset_val[ c ][ i ] = (1 << comp_scale_output_bit_depth ) ÷
    (comp_scale_num_points_minus1[ c ])

```

## 【0256】

一代替形態では、`comp_scale_num_points_minus1[ c ]` をシグナリングする代わりに、ピボット点の数が、`log2_comp_scale_num_points[ c ]` を使用してシグナリングされ、ここで、`(1 << log2_comp`

10

20

30

40

50

`mp_scale_num_points[ c ]` は、`c` 番目のコンポーネントのためのピボット点の数を指定する。

代替的に、`comp_scale_offset_val[ c ][ ]` および `comp_scale_val[ c ][ ]` の各々は、浮動小数点数として、または指数および仮数をもつ 2 つのシンタックス要素としてシグナリングされる。

別の代替形態では、`comp_scale_val[ c ][ i ]` のシグナリングは、`comp_scale_output_point[ c ][ i ]` によって置き換えられる。

シンタックス要素の残りのセマンティクスは、例 1 で説明されたものと同様である。

例 3

[0190] 例 3 で説明されるこの方法は、コンポーネントスケーリング関数が独立して更新されることを可能にされることを除いて、例 2 で説明された代替形態の方法と同様である。 10

【0257】

【表 6】

コンポーネントスケーリング SEI メッセージのシンタックス

記述子
component_scale_info( payloadSize ) {
<b>comp_scale_id</b>
<b>comp_scale_cancel_flag</b>
if( !comp_scale_cancel_flag ) {
<b>comp_scale_persistence_flag</b>
<b>comp_scale_num_comps</b>
<b>comp_scale_input_bit_depth</b>
<b>comp_scale_output_bit_depth</b>
for( c = 0; c < comp_scale_num_comps; c++ ) {
<b>comp_scale_persist_component_flag[ c ]</b>
if( !comp_scale_persist_component_flag[ c ] )
<b>comp_scale_num_scale_regions[ c ]</b>
<b>comp_scale_global_offset_input_val[ c ]</b>
<b>comp_scale_global_offset_output_val[ c ]</b>
for( i = 0; i < comp_scale_num_scale_regions[ c ]; i++ ) {
<b>comp_scale_offset_val[ c ][ i ]</b>
<b>comp_scale_val[ c ][ i ]</b>
}
}
}
}
}

【0258】

コンポーネントスケーリング SEI メッセージのセマンティクス

以下のシンタックス要素を除いて、セマンティクスは例 2 と同様である。

`comp_scale_num_scale_regions[ c ]` は、シンタックス要素 `comp_scale_val[ c ][ i ]` が `c` のコンポーネントのためにシグナリン 50

グされる領域の数を指定する。`comp_scale_num_scale_regions[c]`は、両端値を含む、`0 ~ (1 << comp_scale_input_bit_depth) - 1`のレンジ中にあるものとする。

`0`に等しい`comp_scale_persist_component_flag[c]`は、`c`番目のコンポーネントのためのコンポーネントスケーリングパラメータが、SEIメッセージ中で明示的にシグナリングされることを指定する。`1`に等しい`comp_scale_persist_component_flag[c]`は、`c`番目のコンポーネントのためのコンポーネントスケーリングパラメータが、SEIメッセージ中で明示的にシグナリングされず、それが、出力順序で、前のピクチャに適用されるコンポーネントスケーリングSEIメッセージの`c`番目のコンポーネントのコンポーネントスケーリングパラメータから持続することを指定する。  
10

コンポーネントスケーリングSEIメッセージがI R A Pアクセスユニット中に存在するとき、`comp_scale_persist_component_flag[c]`の値は、存在するとき、`0`に等しいものとすることが、ビットストリームコンフォーマンスの要件である。

代替的に、以下の条件が加えられる。

コンポーネントスケーリングSEIメッセージが、I R A Pアクセスユニットでないアクセスユニット中に存在し、`comp_scale_persist_component_flag[c]`が`1`に等しいとき、両端値を含む、出力順序で現在ピクチャに先行し、復号順序で前のI R A Pピクチャに出力順序で続く、少なくとも1つのピクチャがあり、その結果、その1つのピクチャが、`1`に等しい`comp_scale_persistence_flag`をもつコンポーネントスケーリングSEIメッセージに関連することが、ビットストリームコンフォーマンスの要件である。  
20

`comp_scale_persistence_flag`は、現在レイヤのためのコンポーネントスケーリング情報SEIメッセージの持続性を指定する。

`0`に等しい`comp_scale_persistence_flag`は、コンポーネントスケーリング情報が現在復号ピクチャのみに適用されることを指定する。

`pic A`を現在ピクチャとする`1`に等しい`comp_scale_persistence_flag`は、以下の条件のいずれかが真になるまで、`c`番目のコンポーネントのコンポーネントスケーリング情報が出力順序で現在レイヤのために持続することを指定する。  
30

- 現在レイヤの新しいCLVSが開始する。
- ビットストリームが終了する。
- `comp_scale_id`の同じ値および`0`に等しい`comp_scale_persist_component_flag[c]`をもち、現在レイヤに適用可能な、コンポーネントスケーリング情報SEIメッセージを含んでいるアクセスユニット中の現在レイヤ中のピクチャ`pic B`が出力され、それについて、`PicOrderCnt(pic B)`が`PicOrderCnt(pic A)`よりも大きく、ここで、`PicOrderCnt(pic B)`および`PicOrderCnt(pic A)`が、それぞれ、`pic B`のためのピクチャ順序カウントのための復号プロセスの呼び出しの直後の`pic B`および`pic A`の`PicOrderCntVal`値である。  
40

例 4

[0191] この例 4 では、スケール領域をシグナリングするための異なる方法が開示される。

【0259】

【表7】

コンポーネントスケーリングSELメッセージシングタックスの変更

component_scale_info( payloadSize ) {	記述子
<b>comp_scale_id</b>	ue(v)
<b>comp_scale_cancel_flag</b>	u(1)
if( !comp_scale_cancel_flag ) {	
<b>comp_scale_persistence_flag</b>	u(1)
<b>comp_scale_num_comps</b>	ue(v)
<b>comp_scale_input_bit_depth</b>	ue(v)
<b>comp_scale_output_bit_depth</b>	ue(v)
for( c = 0; c < comp_scale_num_comps; c++ ) {	
<b>comp_scale_persist_component_flag[ c ]</b>	u(1)
if( !comp_scale_persist_component_flag[ c ] )	
<b>comp_scale_global_offset_input_val[ c ]</b>	u(v)
<b>comp_scale_global_offset_output_val[ c ]</b>	u(v)
<b>comp_scale_num_scale_regions[ c ]</b>	ue(v)
for( i = 0; i < comp_scale_num_scale_regions[ c ]; i++ ) {	
<b>comp_scale_offset_begin_val[ c ][ i ]</b>	u(v)
<b>comp_scale_offset_end_val[ c ][ i ]</b>	u(v)
<b>comp_scale_val[ c ][ i ]</b>	u(v)
}	
}	
}	
}	
}	

## 【0260】

コンポーネントスケーリングSELメッセージセマンティクスの変更

シンタックス要素のセマンティクスは、以下を除いて、前の例で説明されたものと同様である。

**comp\_scale\_offset\_set\_begin\_val[ c ][ i ]** は、スケール値 **comp\_scale\_val[ c ][ i ]** が適用可能であるサンプル値レンジの開始を指定する。**comp\_scale\_offset\_set\_begin\_val[ c ]** を表すために使用されるビット数は、**comp\_scale\_input\_bit\_depth** に等しい。

**comp\_scale\_offset\_set\_end\_val[ c ][ i ]** は、スケール値 **comp\_scale\_val[ c ][ i ]** が適用可能である例示的な値レンジの終了を指定する。**comp\_scale\_offset\_set\_end\_val[ c ]** を表すために使用されるビット数は、**comp\_scale\_input\_bit\_depth** に等しい。

**comp\_scale\_offset\_set\_begin\_val** および **comp\_scale\_offset\_set\_end\_val** によって明示的に指定されない領域のために、それらの領域のための **comp\_scale\_value[ c ][ i ]** は、0 に等しいと推論され

10

20

30

40

50

る。

【0261】

[0192] 代替的に、`comp_scale_offset_set_end_val[c][i]` はシグナリングされず、代わりに、`comp_scale_offset_set_end_val[c][i]` と `comp_scale_offset_set_begin_val[c][i]` との間の差分がシグナリングされ、デコーダ側において導出された `comp_scale_offset_set_end_val[c][i]` の値。

【0262】

[0193] 別の代替形態では、出力サンプルレンジがそれにスプリットされる領域の総数が指定され、スケール領域がそれのために明示的にシグナリングされる領域の数がシグナリングされる。

【0263】

【表8】

...	u(v)
<code>comp_scale_global_offset_output_val[c]</code>	u(v)
<code>comp_scale_tot_scale_regions[c]</code>	ue(v)
<code>comp_scale_num_scale_regions[c]</code>	ue(v)
<code>for(i = 0; i &lt; comp_scale_num_scale_regions[c]; i++) {</code>	
<code>comp_scale_region_idx[c][i]</code>	u(v)
<code>comp_scale_val[c][i]</code>	u(v)
}	
...	

【0264】

`comp_scale_tot_scale_regions[c]` は、サンプル値がそれにスプリットされる等長サンプル値レンジの総数を指定する。`comp_scale_tot_scale_regions[c]` を表すために使用されるビット数は、`comp_scale_input_bit_depth` に等しい。

一代替形態では、`comp_scale_tot_scale_regions[c]` サンプル値レンジは、長さが正確に等しくないことがあるが、領域長さの整数正確さを説明することに極めてほぼ等しい。

`comp_scale_region_idx[c][i]` は、スケール値 `comp_scale_val[c][i]` が適用されるサンプル値レンジのインデックスを指定する。シンタックス要素 `comp_scale_region_idx[c]` の長さは、 $Ceil(Log2(comp_scale_tot_scale_regions[c]))$  ビットである。

代替形態

代替的に、クロマ中間の周りの領域(10ビットデータの場合511)は、より小さいサイズ、p.e.他の領域の半分のサイズを有する。

例5

【0265】

10

20

30

40

【表9】

コンポーネントスケールSEIメッセージのシンタックス

component_scale_info( payloadSize ) {	記述子	
<b>comp_scale_id</b>	ue(v)	
<b>comp_scale_cancel_flag</b>	u(1)	
if( !comp_scale_cancel_flag ) {		
<b>comp_scale_persistence_flag</b>	u(1)	
<b>comp_scale_scale_bit_depth</b>	u(4)	10
<b>comp_scale_offset_bit_depth</b>	u(4)	
<b>comp_scale_scale_frac_bit_depth</b>	u(4)	
<b>comp_scale_offset_frac_bit_depth</b>	u(4)	
<b>comp_scale_num_comps_minus1</b>	ue(v)	
for( c = 0; c <= comp_scale_num_comps_minus1; c++ ) {		
<b>comp_scale_num_ranges[ c ]</b>	ue(v)	
<b>comp_scale_equal_ranges_flag[ c ]</b>	u(1)	20
<b>comp_scale_global_offset_val[ c ]</b>	u(v)	
for( i = 0; i <= comp_scale_num_ranges[ c ]; i++ )		
<b>comp_scale_scale_val[ c ][ i ]</b>	u(v)	
if( !comp_scale_equal_ranges[ c ] )	u(v)	
for( i = 0; i <= comp_scale_num_ranges[ c ]; i++ )		
<b>comp_scale_offset_val[ c ][ i ]</b>	u(v)	
}		
}		30

【0266】

コンポーネントスケールSEIメッセージのセマンティクス

コンポーネントスケーリングSEIメッセージは、復号ピクチャの様々なコンポーネントに対してスケーリング動作を実行するための情報を与える。スケーリング動作が実行されるべきである色空間およびコンポーネントは、SEIメッセージ中でシグナリングされるシンタックス要素の値によって決定される。

`comp_scale_id`は、コンポーネントスケーリングSEIメッセージの目的を識別するために使用され得る識別番号を含んでいる。`comp_scale_id`の値は、両端値を含む、 $0 \sim 2^{32} - 2$ のレンジ中にあるものとする。`comp_scale_id`の値は、コンポーネントスケーリングSEIメッセージ色空間を指定する、または、コンポーネントスケーリングSEIメッセージが線形領域中で適用されるのか、非線形領域中で適用されるのかを指定するために使用され得る。

いくつかの例では、`comp_scale_id`は、HDR再構成プロセスの構成を指定することができる。いくつかの例では、`comp_scale_id`の特定の値は、3つのコンポーネントのためのスケーリングパラメータのシグナリングに関連し得る。R'、G'、B'色空間のサンプルに適用されるべき第1のコンポーネントのスケーリング、後続の2つのコンポーネントのパラメータは、CrおよびCbのスケーリングのために適用される。

また別の`comp_scale_id`値の場合、hdr再構成プロセスは、3つのコンポ

ーネントのためのパラメータを利用することができ、スケーリングは、ルーマ、Cr およびCb 色コンポーネントのサンプルに適用される。

また別のcomp\_scale\_id 値では、hdr 再構成プロセスは、4 つのコンポーネントのためのシグナリングを利用することができ、ルーマ、Cr およびCb スケーリングに適用されるべき4 つのコンポーネントのうちの3 つ、ならびに色補正のパラメータをもたらすための第4 のコンポーネント。

いくつかの例では、comp\_scale\_id 値のいくつかのレンジは、SDR 後方互換がある構成において行われるHDR 再構成に関連し得るが、comp\_scale\_id 値の別のレンジは、後方互換性がない構成に行われるHDR 再構成に関連し得る。

両端値を含む、0 ~ 255 の、および、両端値を含む、512 ~ 2<sup>31</sup> - 1 のcomp\_scale\_id の値は、適用によって決定されたように使用され得る。両端値を含む、256 ~ 511 の、および、両端値を含む、2<sup>31</sup> ~ 2<sup>32</sup> - 2 のcomp\_scale\_id の値は、ITU-T | ISO/IEC による将来の使用のために予約される。デコーダは、両端値を含む、256 ~ 511 のレンジ中の、または、両端値を含む、2<sup>31</sup> ~ 2<sup>32</sup> - 2 のレンジ中のcomp\_scale\_id の値を含んでいるすべてのコンポーネントスケール情報SEI メッセージを無視するものとし、ビットストリームは、そのような値を含んでいないものとする。

注1 - comp\_scale\_id は、異なるディスプレイシナリオに好適であるコンポーネントスケーリングプロセスをサポートするために使用され得る。たとえば、comp\_scale\_id の異なる値は、スケーリングが適用される異なるディスプレイビット深度または異なる色空間に対応し得る。

代替的に、comp\_scale\_id はまた、スケーリングがいくつかのタイプのディスプレイまたはデコーダ、たとえばHDR、SDR への互換性のために実行されるかどうかを識別するために使用され得る。

1に等しいcomp\_scale\_cancel\_flag は、コンポーネントスケーリング情報SEI メッセージが、現在レイヤに適用される出力順序で前のコンポーネント情報SEI メッセージの持続性を消去することを示す。0 に等しいcomp\_scale\_cancel\_flag は、コンポーネントスケーリング情報が続くことを示す。

comp\_scale\_persistence\_flag は、現在レイヤのためのコンポーネントスケーリング情報SEI メッセージの持続性を指定する。

0 に等しいcomp\_scale\_persistence\_flag は、コンポーネントスケーリング情報が現在復号ピクチャのみに適用されることを指定する。

picA を現在ピクチャとする。1 に等しいcomp\_scale\_persistence\_flag は、以下の条件のいずれかが真になるまで、コンポーネントスケーリング情報が出力順序で現在レイヤのために持続することを指定する。

- 現在レイヤの新しいCLVS が開始する。
- ビットストリームが終了する。
- comp\_scale\_id の同じ値をもち、現在レイヤに適用可能な、コンポーネントスケーリング情報SEI メッセージを含んでいるアクセスユニット中の現在レイヤ中のピクチャpicB が output され、それについて、PicOrderCnt(picB) が PicOrderCnt(picA) よりも大きく、ここで、PicOrderCnt(picB) およびPicOrderCnt(picA) が、それぞれ、picB のためのピクチャ順序カウントのための復号プロセスの呼出しの直後のpicB およびpicA のPicOrderCntVal 値である。

comp\_scale\_scale\_bit\_depth は、シンタックス要素comp\_scale\_scale\_val[c][i] をシグナリングするために使用されるビット数を指定する。comp\_scale\_scale\_bit\_depth の値は、両端値を含む、0 ~ 15 のレンジ中にあるものとする。

comp\_scale\_offset\_bit\_depth は、シンタックス要素comp\_scale\_global\_offset\_val[c] およびcomp\_sca

10

20

30

40

50

`e_offset_val[c][i]` をシグナリングするために使用されるビット数を指定する。`comp_scale_offset_bit_depth` の値は、両端値を含む、0 ~ 15 のレンジ中にあるものとする。

`comp_scale_scale_frac_bit_depth` は、`c` 番目のコンポーネントの `i` 番目のパーティションのスケールパラメータの小数部分を示すために使用される LSB の数を指定する。`comp_scale_scale_frac_bit_depth` の値は、両端値を含む、0 ~ 15 のレンジ中にあるものとする。`comp_scale_scale_bit_depth` の値は、`comp_scale_scale_bit_depth` の値よりも小さいかまたはそれに等しいものとする。

`comp_scale_offset_frac_bit_depth` は、`c` 番目のコンポーネントの `i` 番目のパーティションのオフセットパラメータおよび `c` 番目のコンポーネントのグローバルオフセットの小数部分を示すために使用される LSB の数を指定する。

`comp_scale_offset_frac_bit_depth` の値は、`comp_scale_offset_bit_depth` の値よりも小さいかまたはそれに等しいものとする。

`comp_scale_num_comps_minus1 + 1` は、コンポーネントスケーリング関数が指定されるコンポーネントの数を指定する。`comp_scale_num_comps_minus1` は、両端値を含む、0 ~ 2 のレンジ中にあるものとする。

`comp_scale_num_ranges[c]` は、出力サンプルレンジがそれに区分されるレンジの数を指定する。`comp_scale_num_ranges[c]` の値は、両端値を含む、0 ~ 63 のレンジ中にあるものとする。。

1 に等しい `comp_scale_equal_ranges_flag[c]` は、その出力サンプルレンジが、`comp_scale_num_ranges[c]` のほぼ等しいパーティションに区分されることを示し、パーティション幅は明示的にシグナリングされない。0 に等しい `comp_scale_equal_ranges_flag[c]` は、出力サンプルレンジが、そのすべてが同じサイズのものであるとは限らない `comp_scale_num_ranges[c]` パーティションに区分され得ることを示し、パーティション幅は明示的にシグナリングされる。

`comp_scale_global_offset_val[c]` は、`c` 番目のコンポーネントのための有効な入力データレンジの最も小さい値をマッピングするために使用されるオフセット値を導出するために使用される。`comp_scale_global_offset_val[c]` の長さは、`comp_scale_offset_bit_depth` ビットである。

`comp_scale_scale_val[c][i]` は、`c` 番目のコンポーネントの `i` 番目のパーティションのその幅を導出するために使用されるオフセット値を導出するために使用される。`comp_scale_global_offset_val[c]` の長さは、`comp_scale_offset_bit_depth` ビットである。

変数 `CompScaleScaleVal[c][i]` は以下のように導出される。

【0267】

【数44】

```
CompScaleScaleVal[c][i] = (comp_scale_scale_val[c][i] >>
    comp_scale_scale_frac_bit_depth) +
    (comp_scale_scale_val[c][i] &
    ((1 << comp_scale_scale_frac_bit_depth) - 1))
)
```

(1 << comp\_scale\_scale\_frac\_bit\_depth)

【0268】

10

20

30

40

50

`comp_scale_offset_val[c][i]` は、`c` 番目のコンポーネントの `i` 番目のパーティションのその幅を導出するために使用されるオフセット値を導出するために使用される。`comp_scale_global_offset_val[c]` の長さは、`comp_scale_offset_bit_depth` ビットである。`comp_scale_offset_val[c][i]` がシグナリングされるとき、`CompScaleOffsetVal[c][i]` の値は以下のように導出される。

【0269】

【数45】

```
CompScaleOffsetVal[c][i] = (comp_scale_offset_val[c][i] >>
                           comp_scale_offset_frac_bit_depth) +
                           (comp_scale_offset_val[c][i] &
                            ((1 << comp_scale_offset_frac_bit_depth) - 1))
                           )) ÷
                           (1 << comp_scale_offset_frac_bit_depth)
```

【0270】

代替的に、変数 `CompScaleScaleVal[c][i]` および `CompScaleOffsetVal[c][i]` は、以下のように導出される。

【0271】

【数46】

```
CompScaleScaleVal[c][i] = comp_scale_val[c][i] &
                           (1 << comp_scale_scale_frac_bit_depth)
CompScaleOffsetVal[c][i] = comp_scale_offset_val[c][i] ÷
                           (1 << comp_scale_offset_frac_bit_depth)
```

【0272】

`comp_scale_equal_ranges_flag[c]` が 1 に等しいとき、`comp_scale_offset_val[c][i]` はシグナリングされず、`CompScaleOffsetVal[c][i]` の値は以下のように導出される。

【0273】

【数47】

`CompScaleOffsetVal[c][i] = 1 ÷ comp_scale_num_ranges[c]`

【0274】

`0 ~ comp_scale_num_ranges[c]` のレンジ中の `i` の場合の変数 `CompScaleOutputRanges[c][i]` および `CompScaleOutputRanges[c][i]` は、以下のように導出される。

【0275】

## 【数4 8】

```

for( i = 0; i <= comp_scale_num_ranges[ c ]; i++ )
    if( i == 0 )
        CompScaleOutputRanges[ c ][ i ] = comp_scale_global_offset_val[ c ]÷
            (1 << comp_scale_offset_frac_bit_depth )
        CompScaleInputRanges[ c ][ i ] = 0
    else
        CompScaleInputRanges[ c ][ i ] = CompScaleOffsetInputRanges[ c ][ i - 10
1 ] +
            (CompScaleOffsetVal[ c ][ i - 1 ] *
        CompScaleScaleVal[ c ][ i - 1 ]
        CompScaleOutputRanges[ c ][ i ] = CompScaleOutputRanges[ c ][ i - 1 ] +
            CompScaleOffsetVal[ c ][ i - 1 ]

```

## 【0276】

一代替形態では、CompScaleOutputRanges[ ][ ] および CompScaleOutputRanges[ ][ ] の値は、以下のように導出される。 20

## 【0277】

## 【数4 9】

```

for( i = 0; i <= comp_scale_num_ranges[ c ]; i++ )
    if( i == 0 )
        CompScaleInputRanges[ c ][ i ] = comp_scale_global_offset_val[ c ]÷
            (1 << comp_scale_offset_frac_bit_depth )
        CompScaleOutputRanges[ c ][ i ] = 0
    else
        CompScaleInputRanges[ c ][ i ] = CompScaleOffsetInputRanges[ c ][ i - 30
1 ] +
            (CompScaleOffsetVal[ c ][ i - 1 ] *
        CompScaleScaleVal[ c ][ i - 1 ]
        CompScaleOutputRanges[ c ][ i ] = CompScaleOutputRanges[ c ][ i - 1 ] +
            CompScaleOffsetVal[ c ][ i - 1 ]

```

## 【0278】

両方のためのサンプル値、入力表現が、0 ~ 1 のレンジ中に正規化され、出力表現が、および 0 ~ 1 のレンジ中にある、( 整数点ならびに浮動小数点の両方をカバーするために使用され得る ) 入力信号表現 x および出力信号表現 y をマッピングするプロセスは、以下のように指定される。 40

## 【0279】

## 【数50】

```

if( x <= CompScaleInputRanges[ c ][ 0 ] )
    y = CompScaleOutputRanges[ c ][ 0 ]
else if( x > CompScaleInputRanges[ c ][ comp_scale_num_ranges[ c ] ] )
    y = CompScaleOutputRanges[ c ][ comp_scale_num_ranges[ c ]; ]
else
    for( i = 1; i <= comp_scale_num_ranges[ c ]; i++ )
        if( CompScaleInputRanges[ i - 1 ] < x && x <=
10
CompScaleInputRanges[ i ] )
            y = ( x -
CompScaleInputRanges[ i - 1 ] ) □ comp_scale_val[ c ][ i ] +
CompScaleOutputRanges[ c ][ i - 1 ]

```

## 【0280】

一代替形態では、CompScaleOutputRanges[ c ][ 0 ] の値は、許容サンプル値レンジに基づいて設定される。代替的に、入力値 valIn を出力値 valOut にマッピングするプロセスは、以下のように定義される。

20

## 【0281】

## 【数51】

```

m_pAtfRangeIn[ 0 ] = 0;
m_pAtfRangeOut[ 0 ] = -m_offset2 * m_pAtfScale2[ c ][ 0 ];
for (int j = 1; j < m_atfNumberRanges + 1; j++)
{
    m_pAtfRangeIn[ j ] = m_pAtfRangeIn[ j - 1 ] + m_pAtfDelta[ j - 1 ];
    m_pAtfRangeOut[ j ] = m_pAtfRangeOut[ j - 1 ] + m_pAtfScale2[ c ][ j -
30
1 ] *
    m_pAtfDelta[ j - 1 ];
}

```

```

for (int j = 0; j < numRanges && skip == 0; j++)
{
    if (valIn <= pAtfRangeIn[ j + 1 ])

```

40

```

    {
        valOut = (valIn - pOffset[ component ][ j ]) *
pScale[ component ][ j ];
        skip = 1;
    }
}

```

## 【0282】

一代替形態では、m\_offset2 は、comp\_scale\_global\_offset\_set\_val[ c ] ÷ ( 1 << comp\_scale\_offset\_setfracbi

50

`t_depth`) に等しく、`m_PAtfScale[c][i]` は、`CompScaleScaleVal[c][i]` に等しく、`m_PAtfDelta[i]` は、`c` 番目のコンポーネントのための`CompScaleOffsetSetVal[c][i]` に等しく、`pScale` および `pOffset` は、`m_Atfscale` および `m_PAtfDelta` から導出されたスケールおよびオフセットパラメータである。逆演算はそれに応じて定義されることになる。

#### 例 6

いくつかの例では、たとえば例 5 における、上記で説明されたシグナリング方法のうちのいくつかは、後続の擬似コードに示されているように利用され得る。

`m_atfNumberRanges` は、マッピングされたデータのためのダイナミックレンジ区分の数を指定する、所与の `c` のためのシンタックス要素 `comp_scale_number_ranges[c]` のための項である。10

`m_PAtfRangeIn` は、`CompScaleInputRanges` のための項であり、2つの連結されたパーティション、たとえば、`i` および `i + 1` 間の境界を指定する入力サンプル値を含む `m_atfNumberRanges + 1` のアレイサイズである。。

`m_PAtfRangeOut` は、`CompScaleOutputRanges` のための項であり、2つの連結されたパーティション、たとえば `i` および `i + 1` 間の境界を指定する出力サンプル値を含む `m_atfNumberRanges + 1` のアレイサイズである。20

`m_PAtfScale2` は、変数 `CompScaleScaleVal[c]` のための項であり、各パーティションのためのスケール値を含む `m_atfNumberRanges` のアレイサイズである。

`m_PAtfOffset2` は、各パーティションのためのオフセット値を含む `m_atfNumberRanges` のアレイアレイサイズである。

`m_offset2` は、`comp_scale_global_offset_val` のための項である。

この例では、区分線形モデルのパラメータは、アルゴリズム 1 の場合のように決定された形態シンタックス要素であり得る。

アルゴリズム 1 :

【0 2 8 3】

## 【数52】

```

m_pAtfRangeIn[0] = 0;
m_pAtfRangeOut[0] = -m_offset2 * m_pAtfScale2[c][0];
for (int j = 1; j < m_atfNumberRanges + 1; j++)
{
    m_pAtfRangeIn[j] = m_pAtfRangeIn[j - 1] + m_pAtfDelta[j - 1];
    m_pAtfRangeOut[j] = m_pAtfRangeOut[j - 1] + m_pAtfScale2[c][j - 1]
    * m_pAtfDelta[j - 1];
}
10

for (int j = 0; j < m_atfNumberRanges; j++)
{
    temp = m_pAtfRangeIn[j + 1] - m_pAtfRangeOut[j + 1] /
m_pAtfScale2[c][j];
    m_pAtfOffset2[c][j] = temp;
}
20
}

```

## 【0284】

決定されると、区分線形モデルは、アルゴリズム2の場合のように出力サンプル値 output value を決定するために、入力サンプル値 input value に適用され得る。

アルゴリズム2：

## 【0285】

## 【数53】

```

for (int j = 0; j < m_atfNumberRanges && skip == 0; j++)
{
    if (inValue <= m_pAtfRangeIn[j + 1])
    {
        outValue = (inValue - m_pAtfOffset2[j]) *
m_pAtfScale2[j];
        skip = 1;
    }
}

```

## 【0286】

アルゴリズム3の場合のように行われるべき逆プロセス。

アルゴリズム3：

## 【0287】

30

40

10

20

30

## 【数54】

```

for (int j = 0; j < m_atfNumberRanges && skip == 0; j++)
{
    if (inValue <= m_pAtfRangeOut[j + 1])
    {
        outValue = inValue / m_pAtfScale2[j] + m_pAtfOffset2
[j];
        skip = 1;
    }
}

```

10

## 【0288】

[0194] いくつかの例では、2つの連結されたパーティション  $i$  および  $i + 1$  間の境界サンプル値 ( $m_pAtfRangeIn$  または  $m_pAtfRangeOut$  のエントリ) は、アルゴリズム 2 およびアルゴリズム 3 に示されているように、 $i$  パーティションに属する代わりに、 $i + 1$  に属するとして、別様に解釈され得る。

## 【0289】

[0195] いくつかの例では、アルゴリズム 3 に示されている逆プロセスは、 $m_pAtfScale2[j]$  による除算の代わりに、 $m_pAtfInverseScale2$  値による乗算を用いて実装され得る。そのような例では、 $m_pAtfScale2[j]$  の値は、事前に  $m_pAtfScale2[j]$  から決定される。

20

## 【0290】

[0196] いくつかの例では、 $m_pAtfInverseScale2[j]$  は、 $1 / m_pAtfScale2[j]$  としてデコーダ側において決定される。

## 【0291】

[0197] いくつかの例では、 $m_pAtfInverseScale2[j]$  は、エンコーダ側において算出され、ビットストリームを通してシグナリングされ得る。そのような例では、アルゴリズム 1、2 および 3 において与えられた動作は、それに応じて調整される。

30

## 【0292】

[0198] 様々な例

[0199] いくつかの例では、提案されたシグナリング機構は、たとえばビデオコーディングシステムの圧縮効率を改善するために、入力信号のサンプルのための動的レンジ調整を可能にするために利用され得る区分関数をモデル化するために使用され得る。

## 【0293】

[0200] いくつかの例では、提案されたシグナリング機構は、OETF によって、たとえば、ST.2084 の PQTF、または他のものによって生成されたコードワード (R、G、B サンプルの非線形表現) に適用され得る区分関数をモデル化するために使用され得る。

40

## 【0294】

[0201] いくつかの例では、提案されたシグナリング機構は、YCbCr 色表現のサンプルに適用され得る区分関数をモデル化するために使用され得る。

## 【0295】

[0202] いくつかの例では、提案されたシグナリング機構は、SDR 互換性をもつ HDR / WCG ソリューションに利用され得る区分関数をモデル化するために使用され得る。

## 【0296】

[0203] いくつかの例では、提案されたシグナリング機構は、浮動小数点表現中のサンプルに適用され得る区分関数をモデル化するために使用され得る。また別の例では、提案

50

されたシグナリング機構および得られた関数は、整数表現、たとえば10ビットにおけるサンプルに適用され得る。

【0297】

[0204] いくつかの例では、提案されたシグナリング機構は、ルックアップテーブルの形態でサンプルに適用され得る区分関数をモデル化するために使用され得る。また別の例では、提案されたシグナリングは、乗算器の形態でサンプルに適用され得る関数をモデル化するために使用され得る。

組合せおよび拡張

[0205] 上記の例では、線形モデルは、各領域（すなわち、スケール+オフセット）のために仮定され、本開示の技法はまた、たとえば、2つの代わりに3つのパラメータを必要とする二次の多項式をもつ、より高次の多項式モデルのために適用可能であり得る。シグナリングおよびシンタックスは、このシナリオのために適切に拡張されることになる。

【0298】

[0206] 上記で説明された様の組合せおよび本開示の技法の一部が可能である。

【0299】

[0207] ツールボックス組合せ：本開示で説明されるSEIの目標とやや同様の目標をターゲットにすることがあるいくつかのHDR方法がある。HDR方法のうちの2つ以上を適応するが、同時に、フレームごとに適用可能なSEI処理の数を限定するために、単一のSEI中にこれらの方法（のうちの1つまたは複数）を組み合わせることが提案される。提案されたシンタックス要素は、各インスタンスにおいて適用されるための特定の方法を示すことになる。たとえば、2つの可能な方法がSEIにある場合、シンタックス要素は使用されるべき1つを示すフラグであることになる。

例7

[0208] この例では、スケールパラメータのシグナリングは、負スケールが送信され得るように変更され、シグナリングされたスケールパラメータは、様々なコンポーネントの異なるレンジのために適用されるべきスケールの変数を示す。例5に関する変更は以下である。

【0300】

10

20

## 【表10】

SEIメッセージのシンタックスの変更

component_scale_info( payloadSize ) {	記述子
<b>comp_scale_id</b>	ue(v)
<b>comp_scale_cancel_flag</b>	u(1)
if( !comp_scale_cancel_flag ) {	
<b>comp_scale_persistence_flag</b>	u(1)
<b>comp_scale_scale_bit_depth</b>	u(4)
<b>comp_scale_offset_bit_depth</b>	u(4)
<b>comp_scale_scale_frac_bit_depth</b>	u(4)
<b>comp_scale_offset_frac_bit_depth</b>	u(4)
<b>comp_scale_negative_scales_present_flag</b>	u(1)
<b>comp_scale_dep_component_id</b>	ue(v)
<b>comp_scale_num_comps_minus1</b>	ue(v)
for( c = 0; c <= comp_scale_num_comps_minus1; c++ ) {	
<b>comp_scale_num_ranges[ c ]</b>	ue(v)
<b>comp_scale_equal_ranges_flag[ c ]</b>	u(1)
<b>comp_scale_global_offset_val[ c ]</b>	u(v)
for( i = 0; i <= comp_scale_num_ranges[ c ]; i++ )	
<b>comp_scale_scale_val[ c ][ i ]</b>	u(v)
if( !comp_scale_equal_ranges[ c ] )	u(v)
for( i = 0; i <= comp_scale_num_ranges[ c ]; i++ )	
<b>comp_scale_offset_val[ c ][ i ]</b>	u(v)
}	
}	

## 【0301】

SEIメッセージのセマンティクスの変更

1に等しい**comp\_scale\_negative\_scales\_present\_flag**は、**comp\_scale\_scale\_val[ c ][ i ]**から導出されたスケールパラメータの整数部分が符号付き整数として表されることを指定する。0に等しい**comp\_scale\_negative\_scales\_present\_flag**は、**comp\_scale\_scale\_val[ c ][ i ]**から導出された整数部分スケールパラメータが符号なし整数として表されることを指定する。

## 【0302】

[0209] 一代替形態では、オフセットパラメータの別のセットは、第2のコンポーネントの値の関数として第1のコンポーネントに対してスケールとともに適用されるオフセットを定義するために使用される**comp\_scale\_scale\_val**とともにシグナリングされる。

## 【0303】

[0210] 符号付き整数表現は、限定はしないが、2の補数表記法および符号付き絶対値表現（符号のための1ビットおよび整数部分中の残りのビット）を含む。下記の導出は、

10

20

30

40

50

符号付き絶対値表現のために与えられる。導出は、符号付き表現の他の形態のために同様に定義され得る。

変数 `CompScaleScaleEval[c][i]` は以下のように導出される。

【0304】

【数55】

```

compScaleScaleFracPart = ( comp_scale_scale_val[ c ][ i ] &
                           ( (1 << comp_scale_scale_frac_bit_depth ) - 1 )
                           )
                           (1 << comp_scale_scale_frac_bit_depth ) 10
if( comp_scale_negative_scales_present_flag ) {
    compScaleSignPart = comp_scale_scale_val[ c ][ i ] >>
    (comp_scale_scale_bit_depth - 1)
    compScaleIntegerPart = comp_scale_scale_val[ c ][ i ] - ( compScaleSignPart
                                                               << (comp_scale_scale_bit_depth - 1) )
    compScaleIntegerVal = ( ( compScaleSignPart == 1 ) : -1 : 1 ) *
    compScaleIntegerPart
} else 20
    compScaleIntegerVal = comp_scale_scale_val[ c ][ i ] >>
    comp_scale_scale_frac_bit_depth
    CompScaleScaleVal[ c ][ i ] = compScaleIntegerVal + compScaleScaleFracPart

```

【0305】

`comp_scale_negative_scales_present_flag` が 1 に等しいとき、`comp_scale_scale_bit_depth` の値が、`comp_scale_scale_frac_bit_depth` よりも大きいかまたはそれに等しいものとすることが、ビットストリームコンフォーマンスの要件である 30  
`comp_scale_dependent_component_id` は、ビデオの様々なコンポーネントへのスケールおよびオフセットパラメータの適用を指定する。`comp_scale_dependent_component_id` が 0 に等しいとき、シンタックス要素 `comp_scale_global_offset_val[c]`、`comp_scale_scale_val[c][i]` および `comp_scale_offset_val[c][i]` は、c 番目のコンポーネントの入力および出力値のマッピングを識別するために使用される。`comp_scale_dependent_component_id` が 0 よりも大きいとき、`comp_scale_dependent_component_id - 1` は、シンタックス要素 `comp_scale_global_offset_val[c]`、`comp_scale_scale_val[c]` および `comp_scale_offset_val[c][i]` が、サンプルの (`comp_scale_dependent_component_id - 1`) 番目のコンポーネントの値の関数として、サンプルの c 番目のコンポーネントに適用されるべきスケールパラメータのマッピングを指定するように、コンポーネントのインデックスを指定する。 40

セマンティクスの残りは、例 5 で説明されたものと同様である。

例 8

[0211] この例では、ATF パラメータのビット深度はコンポーネントに依存する。各コンポーネントのために、シンタックス要素のビット深度は明示的に信号である。さらに、それらのシンタックス要素のためのデフォルトビット深度がある。ビット深度が明示的

にシグナリングされないとき、デフォルト値が割り当てられる。フラグは、デフォルト値が適用されるのか、それらが明示的にシグナリングされるのかを示し得る。

### 【0306】

[0212] 下記の表はこれらの概念の一例を示す。ATFパラメータのシンタックス要素は、スケール `hdr_recon_scale_val` [ ] [ ] およびレンジ `hdr_recon_range_val` [ ] [ ] である。対応するビット深度（整数および小数部分）を示すシンタックス要素は、後続の要素である。

### 【0307】

#### 【数56】

- `hdr_recon_scale_bit_depth[c]`, 10
- `hdr_recon_offset_bit_depth[c]`,
- `hdr_recon_scale_frac_bit_depth[c]`,
- `hdr_recon_offset_frac_bit_depth[c]`,

### 【0308】

ここで、`c` はコンポーネントインデックスである。スケールおよびオフセット（レンジ）のためのデフォルトビット深度は、以下のように設定され得る。

### 【0309】

#### 【数57】

- `hdr_recon_scale_bit_depth[c] = 8`,
- `hdr_recon_offset_bit_depth[c] = 8`,
- `hdr_recon_scale_frac_bit_depth[c] = 6`,
- `hdr_recon_offset_frac_bit_depth[c] = 8`.

### 【0310】

[0213] また、パラメータの正確さは、ATFパラメータおよび色調整パラメータについて異なり得る。また、デフォルト値は、コンポーネントごとに、および色調整パラメータについて異なり得る。この例では、デフォルト値は同じであると仮定される。 30

### 【0311】

【表 1 1】

hdr_reconstruction_info( payloadSize ) {	記述子
<b>hdr_recon_id</b>	ue(v)
<b>hdr_recon_cancel_flag</b>	u(1)
if( !hdr_recon_cancel_flag ) {	
<b>hdr_recon_persistence_flag</b>	u(1)
if (hdr_recon_id == 1) {	
<b>hdr_output_full_range_flag</b>	10
<b>hdr_output_colour_primaries</b>	
<b>hdr_output_transfer_characteristics</b>	
<b>hdr_output_matrix_coeffs</b>	
}	
マッピングLUTのためのシンタックス	
<b>hdr_recon_num_comps_minus1</b>	ue(v)
for( c = 0; c <= hdr_recon_num_comps_minus1; c++ ) {	
<b>hdr_recon_default_bit_depth</b> [ c ]	u(1)
if ( hdr_recon_default_bit_depth [ c ] == 0) {	
<b>hdr_recon_scale_bit_depth</b> [ c ]	u(4)
<b>hdr_recon_offset_bit_depth</b> [ c ]	u(4)
<b>hdr_recon_scale_frac_bit_depth</b> [ c ]	u(4)
<b>hdr_recon_offset_frac_bit_depth</b> [ c ]	u(4)
}	
<b>hdr_recon_num_ranges</b> [ c ]	ue(v)
<b>hdr_recon_equal_ranges_flag</b> [ c ]	u(1)
<b>hdr_recon_global_offset_val</b> [ c ]	u(v)
for( i = 0; i <= hdr_recon_num_ranges[ c ]; i++ )	
<b>hdr_recon_scale_val</b> [ c ][ i ]	u(v)
if( !hdr_recon_equal_ranges[ c ] )	u(v)
for( i = 0; i <= hdr_recon_num_ranges[ c ]; i++ )	
<b>hdr_recon_range_val</b> [ c ][ i ]	u(v)
}	u(v)

【 0 3 1 2 】

色補正部分のためのシンタックス	
if (hdr_recon_id == 1) {	色補正に関する パラメータ
hdr_color_correction_type	U、Vに対して0、 R、G、Bに対して-1
<b>hdr_color_accuracy_flag</b>	色をコーディングするための シンタックス
if( !hdr_recon_color_accuracy_flag ) {	補正LUT
<b>hdr_color_scale_bit_depth</b>	u(4)
<b>hdr_color_offset_bit_depth</b>	u(4)
<b>hdr_color_scale_frac_bit_depth</b>	u(4)
<b>hdr_color_offset_frac_bit_depth</b>	u(4)
}	
color_correction_num_ranges	
color_correction_equal_len_ranges_flag	
color_correction_zero_offset_val	
for( i = 0; i < color_correction_num_ranges; i++ )	
color_correction_scale_val[ i ]	
if( !color_correction_equal_len_ranges_flag )	
for( i = 0; i < color_correction_num_ranges; i++ )	
color_correction_range_val[ i ]	
}	
}	
}	
}	

【0313】

例 9

[0214] 新しいH D Rソリューションの望ましいプロパティは、それが、H D R 1 0のように、以前のH D Rソリューションに後方互換性があることである。シンタックス要素は、そうであることを示し得る。これはビットストリームの特性を示し、H D Rデコーダは、非A T Fバージョンがすでに閲覧可能である場合、いくつかの状況下では、逆A T F処理に対して計算リソースを費やさないことを決定し得る。

【0314】

[0215] 一例では、h d r \_ r e c o n \_ i dシンタックス要素のいくつかの値は、H D R 1 0後方互換性を示すために、またはどの程度まで後方互換性があるかを示すために予約される。

【0315】

[0216] 別の例では、フラグ(h d r \_ r e c o n \_ h d r 1 0 \_ b c)がこの状況を示す。

10

20

30

40

50

## 【0316】

[0217] 一例では、シグナリングされたH D R 1 0 後方互換性は、ビットストリームが閲覧可能であることを示す。代替的に、それは、シグナリングされた値のいくつかの特定のプロパティ、たとえば、それらがこのプロパティを保証する値のレンジであることを示し得る。たとえば、制約は、スケールが0 . 9 ~ 1 . 1 の間であるということであり得る。

## 【0317】

[0218] 図10は、本開示の技法を実装し得るビデオエンコーダ20の一例を示すプロック図である。ビデオエンコーダ20は、ビデオプリプロセッサユニット19によって処理されたターゲット色コンテナ中のビデオスライス内のビデオブロックのイントラコーディングおよびインターフォーマンスを実行し得る。イントラコーディングは、所与のビデオフレームまたはピクチャ内のビデオの空間冗長性を低減または除去するために空間予測に依拠する。インターフォーマンスは、ビデオシーケンスの隣接フレームまたはピクチャ内のビデオの時間冗長性を低減または除去するために時間予測に依拠する。イントラモード(Iモード)は、いくつかの空間ベースコーディングモードのいずれかを指し得る。単方向予測(Pモード)または双予測(Bモード)などのインターモードは、いくつかの時間ベースコーディングモードのいずれかを指し得る。

## 【0318】

[0219] 図10に示されているように、ビデオエンコーダ20は、符号化されるべきビデオフレーム内の現在ビデオブロックを受信する。図10の例では、ビデオエンコーダ20は、モード選択ユニット40と、ビデオデータメモリ41と、復号ピクチャバッファ64と、加算器50と、変換処理ユニット52と、量子化ユニット54と、エントロピー符号化ユニット56とを含む。モード選択ユニット40は、動き補償ユニット44と、動き推定ユニット42と、イントラ予測ユニット46と、パーティションユニット48とを含む。ビデオブロック再構成のために、ビデオエンコーダ20はまた、逆量子化ユニット58と、逆変換処理ユニット60と、加算器62とを含む。再構成されたビデオからプロックキネスアーティファクトを除去するためにプロック境界をフィルタ処理するための(図10に示されていない)デブロッキングフィルタも含まれ得る。所望される場合、デブロッキングフィルタは、一般に、加算器62の出力をフィルタ処理することになる。(ループ中またはループ後の)追加のフィルタもデブロッキングフィルタに加えて使用され得る。そのようなフィルタは、簡潔のために示されていないが、所望される場合、(ループ内フィルタとして)加算器50の出力をフィルタ処理し得る。

## 【0319】

[0220] ビデオデータメモリ41は、ビデオエンコーダ20のコンポーネントによって符号化されるべきビデオデータを記憶し得る。ビデオデータメモリ41に記憶されるビデオデータは、たとえば、ビデオソース18から取得され得る。復号ピクチャバッファ64は、たとえば、イントラコーディングモードまたはインターフォーマンスモードでビデオエンコーダ20によってビデオデータを符号化する際に使用するための参照ビデオデータを記憶する参照ピクチャメモリであり得る。ビデオデータメモリ41および復号ピクチャバッファ64は、同期D R A M ( S D R A M )を含むダイナミックランダムアクセスメモリ(D R A M )、磁気抵抗R A M ( M R A M )、抵抗性R A M ( R R A M (登録商標) )、または他のタイプのメモリデバイスなど、様々なメモリデバイスのいずれかによって形成され得る。ビデオデータメモリ41および復号ピクチャバッファ64は、同じメモリデバイスまたは別個のメモリデバイスによって与えられ得る。様々な例では、ビデオデータメモリ41は、ビデオエンコーダ20の他のコンポーネントとともにオンチップであるか、またはそれらのコンポーネントに対してオフチップであり得る。

## 【0320】

[0221] 符号化プロセス中に、ビデオエンコーダ20は、コーディングされるべきビデオフレームまたはスライスを受信する。フレームまたはスライスは複数のビデオブロックに分割され得る。動き推定ユニット42および動き補償ユニット44は、時間予測を行う

10

20

30

40

50

ために、1つまたは複数の参照フレーム中の1つまたは複数のブロックに対する受信されたビデオブロックのインター予測コーディングを実行する。イントラ予測処理ユニット46は、代替的に、空間予測を行うために、コーディングされるべきブロックと同じフレームまたはスライス中の1つまたは複数の隣接ブロックに対する受信されたビデオブロックのイントラ予測コーディングを実行し得る。ビデオエンコーダ20は、たとえば、ビデオデータのブロックごとに適切なコーディングモードを選択するために、複数のコーディングパスを実行し得る。

#### 【0321】

[0222] その上、パーティションユニット48は、前のコーディングパスにおける前の区分方式の評価に基づいて、ビデオデータのブロックをサブブロックに区分し得る。たとえば、パーティションユニット48は、初めにフレームまたはスライスをLCUに区分し、レートひずみ分析（たとえば、レートひずみ最適化）に基づいてLCUの各々をサブCUに区分し得る。モード選択ユニット40は、さらに、サブCUへのLCUの区分を示す4分木データ構造を生成し得る。4分木のリーフノードCUは、1つまたは複数のPUと1つまたは複数のTUとを含み得る。

#### 【0322】

[0223] モード選択ユニット40は、たとえば、エラー結果に基づいてコーディングモード、すなわち、イントラまたはインターのうちの1つを選択し得、残差ブロックデータを発生するために、得られたイントラコード化ブロックまたはインターコード化ブロックを加算器50に与え、参照フレームとして使用するための符号化ブロックを再構成するために、得られたイントラコード化ブロックまたはインターコード化ブロックを加算器62に与える。モード選択ユニット40はまた、動きベクトル、イントラモードインジケータ、パーティション情報、および他のそのようなシンタックス情報など、シンタックス要素をエントロピー符号化ユニット56に与える。

#### 【0323】

[0224] 動き推定ユニット42と動き補償ユニット44とは、高度に統合され得るが、概念的な目的のために別個に示されている。動き推定ユニット42によって実行される動き推定は、ビデオブロックの動きを推定する動きベクトルを発生するプロセスである。動きベクトルは、たとえば、現在ピクチャ（または他のコード化ユニット）内でコーディングされている現在ブロックに対する参照ピクチャ（または他のコード化ユニット）内の予測ブロックに対する現在ビデオフレームまたはピクチャ内のビデオブロックのPUの変位を示し得る。予測ブロックは、絶対差分和（SAD：sum of absolute difference）、2乗差分和（SSD：sum of square difference）、または他の差分メトリックによって決定され得るピクセル差分に関して、コーディングされるべきブロックにぴったり一致することがわかるプロックである。いくつかの例では、ビデオエンコーダ20は、復号ピクチャバッファ64に記憶された参照ピクチャのサブ整数ピクセル位置の値を計算し得る。たとえば、ビデオエンコーダ20は、参照ピクチャの1/4ピクセル位置、1/8ピクセル位置、または他の分数ピクセル位置の値を補間し得る。したがって、動き推定ユニット42は、フルピクセル位置と分数ピクセル位置とに対して動き探索を実行し、分数ピクセル精度で動きベクトルを出力し得る。

#### 【0324】

[0225] 動き推定ユニット42は、PUの位置を参照ピクチャの予測ブロックの位置と比較することによって、インターコード化スライス中のビデオブロックのPUのための動きベクトルを計算する。参照ピクチャは、第1の参照ピクチャリスト（リスト0）または第2の参照ピクチャリスト（リスト1）から選択され得、それらの各々が、復号ピクチャバッファ64に記憶された1つまたは複数の参照ピクチャを識別する。動き推定ユニット42は、計算された動きベクトルをエントロピー符号化ユニット56と動き補償ユニット44とに送る。

#### 【0325】

[0226] 動き補償ユニット44によって実行される動き補償は、動き推定ユニット42

10

20

30

40

50

によって決定された動きベクトルに基づいて予測ブロックをフェッチまたは発生することを伴い得る。この場合も、動き推定ユニット42および動き補償ユニット44は、いくつかの例では、機能的に統合され得る。現在ビデオブロックのPUのための動きベクトルを受信すると、動き補償ユニット44は、動きベクトルが参照ピクチャリストのうちの1つにおいてそれを指す予測ブロックの位置を特定し得る。加算器50は、以下で説明されるように、コーディングされている現在ビデオブロックのピクセル値から予測ブロックのピクセル値を減算し、ピクセル差分値を形成することによって、残差ビデオブロックを形成する。概して、動き推定ユニット42はルーマコンポーネントに対して動き推定を実行し、動き補償ユニット44は、クロマコンポーネントとルーマコンポーネントの両方のためにルーマコンポーネントに基づいて計算された動きベクトルを使用する。モード選択ユニット40はまた、ビデオスライスのビデオブロックを復号する際にビデオデコーダ30が使用するためのビデオブロックとビデオスライスとに関連するシンタックス要素を発生し得る。

#### 【0326】

[0227] イントラ予測処理ユニット46は、上記で説明されたように、動き推定ユニット42と動き補償ユニット44とによって実行されるインター予測の代替として、現在ブロックをイントラ予測し得る。特に、イントラ予測処理ユニット46は、現在ブロックを符号化するために使用すべきイントラ予測モードを決定し得る。いくつかの例では、イントラ予測処理ユニット46は、たとえば、別個の符号化パス中に、様々なイントラ予測モードを使用して現在ブロックを符号化し得、イントラ予測処理ユニット46（または、いくつかの例では、モード選択ユニット40）は、テストされたモードから使用するのに適切なイントラ予測モードを選択し得る。

#### 【0327】

[0228] たとえば、イントラ予測処理ユニット46は、様々なテストされたイントラ予測モードのためのレートひずみ分析を使用してレートひずみ値を計算し、テストされたモードの中で最良のレートひずみ特性を有するイントラ予測モードを選択し得る。レートひずみ分析は、概して、符号化ブロックと、符号化ブロックを生成するために符号化された元の符号化されていないブロックとの間のひずみ（またはエラー）の量、ならびに符号化ブロックを生成するために使用されるビットレート（すなわち、ビット数）を決定する。イントラ予測処理ユニット46は、どのイントラ予測モードがブロックについて最良のレートひずみ値を呈するかを決定するために、様々な符号化ブロックのひずみおよびレートから比を計算し得る。

#### 【0328】

[0229] ブロックのためのイントラ予測モードを選択した後に、イントラ予測処理ユニット46は、ブロックのための選択されたイントラ予測モードを示す情報をエントロピー符号化ユニット56に与え得る。エントロピー符号化ユニット56は、選択されたイントラ予測モードを示す情報を符号化し得る。ビデオエンコーダ20は、複数のイントラ予測モードインデックステーブルおよび複数の変更されたイントラ予測モードインデックステーブル（コードワードマッピングテーブルとも呼ばれる）と、様々なブロックの符号化コンテキストの定義と、コンテキストの各々について使用すべき、最確イントラ予測モード、イントラ予測モードインデックステーブル、および変更されたイントラ予測モードインデックステーブルの指示とを含み得る構成データを送信ビットストリーム中に含み得る。

#### 【0329】

[0230] ビデオエンコーダ20は、コーディングされている元のビデオブロックから、モード選択ユニット40からの予測データを減算することによって残差ビデオブロックを形成する。加算器50は、この減算演算を実行する1つまたは複数のコンポーネントを表す。変換処理ユニット52は、離散コサイン変換（DCT）または概念的に同様の変換などの変換を残差ブロックに適用し、残差変換係数値を備えるビデオブロックを生成する。変換処理ユニット52は、DCTと概念的に同様である他の変換を実行し得る。ウェーブレット変換、整数変換、サブバンド変換または他のタイプの変換も使用され得る。いずれ

10

20

30

40

50

の場合も、変換処理ユニット 52 は、変換を残差ブロックに適用し、残差変換係数のブロックを生成する。変換は、残差情報をピクセル値領域から周波数領域などの変換領域に変換し得る。変換処理ユニット 52 は、得られた変換係数を量子化ユニット 54 に送り得る。

### 【 0 3 3 0 】

[0231] 量子化ユニット 54 は、ビットレートをさらに低減するために変換係数を量子化する。量子化プロセスは、係数の一部または全部に関連するビット深度を低減し得る。量子化の程度は、量子化パラメータを調整することによって修正され得る。いくつかの例では、量子化ユニット 54 は、次いで、量子化された変換係数を含む行列の走査を実行し得る。代替的に、エントロピー符号化ユニット 56 が走査を実行し得る。

10

### 【 0 3 3 1 】

[0232] 量子化の後に、エントロピー符号化ユニット 56 は量子化された変換係数をエントロピーコーディングする。たとえば、エントロピー符号化ユニット 56 は、コンテキスト適応型可変長コーディング (C A V L C)、コンテキスト適応型バイナリ算術コーディング (C A B A C)、シンタックスベースコンテキスト適応型バイナリ算術コーディング (S B A C)、確率間隔区分エントロピー (P I P E) コーディングまたは別のエントロピーコーディング技法を実行し得る。コンテキストベースエントロピーコーディングの場合、コンテキストは近隣ブロックに基づき得る。エントロピー符号化ユニット 56 によるエントロピーコーディングの後に、符号化ビットストリームは、別のデバイス (たとえば、ビデオデコーダ 30) に送信されるか、あるいは後で送信するかまたは取り出すためにアーカイブされ得る。

20

### 【 0 3 3 2 】

[0233] 逆量子化ユニット 58 および逆変換処理ユニット 60 は、たとえば、参照ブロックとして後で使用するために、ピクセル領域内で残差ブロックを再構成するために、それぞれ逆量子化および逆変換を適用する。動き補償ユニット 44 は、残差ブロックを復号ピクチャバッファ 64 のフレームのうちの 1 つの予測ブロックに加算することによって参照ブロックを計算し得る。動き補償ユニット 44 はまた、動き推定において使用するためのサブ整数ピクセル値を計算するために、再構成された残差ブロックに 1 つまたは複数の補間フィルタを適用し得る。加算器 62 は、復号ピクチャバッファ 64 に記憶するための再構成されたビデオブロックを生成するために、動き補償ユニット 44 によって生成された動き補償予測ブロックに再構成された残差ブロックを加算する。再構成されたビデオブロックは、後続のビデオフレーム中のブロックをインターコーディングするために、動き推定ユニット 42 および動き補償ユニット 44 によって参照ブロックとして使用され得る。

30

### 【 0 3 3 3 】

[0234] 図 11 は、本開示の技法を実装し得るビデオデコーダ 30 の一例を示すブロック図である。特に、ビデオデコーダ 30 は、上記で説明されたように、ビデオポストプロセッサユニット 31 によって次いで処理され得るターゲット色コンテナにビデオデータを復号し得る。図 11 の例では、ビデオデコーダ 30 は、エントロピー復号ユニット 70 と、ビデオデータメモリ 71 と、動き補償ユニット 72 と、イントラ予測処理ユニット 74 と、逆量子化ユニット 76 と、逆変換処理ユニット 78 と、復号ピクチャバッファ 82 と、加算器 80 とを含む。ビデオデコーダ 30 は、いくつかの例では、ビデオエンコーダ 20 (図 10) に関して説明された符号化パスに概して相反する復号パスを実行し得る。動き補償ユニット 72 は、エントロピー復号ユニット 70 から受信された動きベクトルに基づいて予測データを発生し得、イントラ予測ユニット 74 は、エントロピー復号ユニット 70 から受信されたイントラ予測モードインジケータに基づいて予測データを発生し得る。

40

### 【 0 3 3 4 】

[0235] ビデオデータメモリ 71 は、ビデオデコーダ 30 のコンポーネントによって復号されるべき、符号化ビデオビットストリームなどのビデオデータを記憶し得る。ビデオ

50

データメモリ 7 1 に記憶されるビデオデータは、たとえば、コンピュータ可読媒体 1 6 から、たとえば、カメラなどのローカルビデオソースから、ビデオデータのワイヤードまたはワイヤレスネットワーク通信を介して、あるいは物理データ記憶媒体にアクセスすることによって取得され得る。ビデオデータメモリ 7 1 は、符号化ビデオビットストリームからの符号化ビデオデータを記憶するコード化ピクチャバッファ ( C P B ) を形成し得る。復号ピクチャバッファ 8 2 は、たとえば、イントラコーディングモードまたはインターコーディングモードでビデオデコーダ 3 0 によってビデオデータを復号する際に使用するための参照ビデオデータを記憶する参照ピクチャメモリであり得る。ビデオデータメモリ 7 1 および復号ピクチャバッファ 8 2 は、同期 D R A M ( S D R A M ) を含むダイナミックランダムアクセスメモリ ( D R A M ) 、磁気抵抗 R A M ( M R A M ) 、抵抗性 R A M ( R R A M ) 、または他のタイプのメモリデバイスなど、様々なメモリデバイスのいずれかによって形成され得る。ビデオデータメモリ 7 1 および復号ピクチャバッファ 8 2 は、同じメモリデバイスまたは別個のメモリデバイスによって与えられ得る。様々な例では、ビデオデータメモリ 7 1 は、ビデオデコーダ 3 0 の他のコンポーネントとともにオンチップであるか、またはそれらのコンポーネントに対してオフチップであり得る。

### 【 0 3 3 5 】

[0236] 復号プロセス中に、ビデオデコーダ 3 0 は、ビデオエンコーダ 2 0 から、符号化ビデオスライスのビデオブロックと、関連するシンタックス要素とを表す符号化ビットストリームを受信する。ビデオデコーダ 3 0 のエントロピー復号ユニット 7 0 は、量子化された係数と、動きベクトルまたはイントラ予測モードインジケータと、他のシンタックス要素とを発生するために、ビットストリームをエントロピー復号する。エントロピー復号ユニット 7 0 は、動きベクトルと他のシンタックス要素とを動き補償ユニット 7 2 に転送する。ビデオデコーダ 3 0 は、ビデオスライスレベルおよび / またはビデオブロックレベルでシンタックス要素を受信し得る。

### 【 0 3 3 6 】

[0237] ビデオスライスがイントラコード化 ( I ) スライスとしてコーディングされるとき、イントラ予測処理ユニット 7 4 は、シグナリングされたイントラ予測モードと、現在フレームまたはピクチャの、前に復号されたブロックからのデータとに基づいて、現在ビデオスライスのビデオブロックのための予測データを発生し得る。ビデオフレームがインターコード化 ( すなわち、 B または P ) スライスとしてコーディングされるとき、動き補償ユニット 7 2 は、エントロピー復号ユニット 7 0 から受信された動きベクトルと他のシンタックス要素とに基づいて、現在ビデオスライスのビデオブロックのための予測ブロックを生成する。予測ブロックは、参照ピクチャリストのうちの 1 つ内の参照ピクチャのうちの 1 つから生成され得る。ビデオデコーダ 3 0 は、復号ピクチャバッファ 8 2 に記憶された参照ピクチャに基づいて、デフォルト構成技法を使用して、参照ピクチャリスト、リスト 0 およびリスト 1 を構成し得る。動き補償ユニット 7 2 は、動きベクトルと他のシンタックス要素とをパースすることによって現在ビデオスライスのビデオブロックのための予測情報を決定し、復号されている現在ビデオブロックのための予測ブロックを生成するために、その予測情報を使用する。たとえば、動き補償ユニット 7 2 は、ビデオスライスのビデオブロックをコーディングするために使用される予測モード ( たとえば、イントラまたはインター予測 ) と、インター予測スライスタイプ ( たとえば、 B スライスまたは P スライス ) と、スライスのための参照ピクチャリストのうちの 1 つまたは複数のための構成情報と、スライスの各インター符号化ビデオブロックのための動きベクトルと、スライスの各インターコード化ビデオブロックのためのインター予測ステータスと、現在ビデオスライス中のビデオブロックを復号するための他の情報とを決定するために、受信されたシンタックス要素のうちのいくつかを使用する。

### 【 0 3 3 7 】

[0238] 動き補償ユニット 7 2 はまた、補間フィルタに基づいて補間を実行し得る。動き補償ユニット 7 2 は、参照ブロックのサブ整数ピクセルのための補間値を計算するために、ビデオブロックの符号化中にビデオエンコーダ 2 0 によって使用された補間フィルタ

10

20

30

40

50

を使用し得る。この場合、動き補償ユニット72は、受信されたシンタックス要素からビデオエンコーダ20によって使用された補間フィルタを決定し、予測ブロックを生成するためにその補間フィルタを使用し得る。

【0338】

[0239] 逆量子化ユニット76は、ビットストリーム中で与えられ、エントロピー復号ユニット70によって復号された量子化された変換係数を逆量子化(inverse quantize)、すなわち、量子化解除(de-quantize)する。逆量子化プロセスは、量子化の程度を決定し、同様に、適用されるべき逆量子化の程度を決定するための、ビデオライス中のビデオブロックごとにビデオデコーダ30によって計算される量子化パラメータ $Q_{P_Y}$ の使用を含み得る。逆変換処理ユニット78は、ピクセル領域において残差ブロックを生成するために、逆変換、たとえば、逆DCT、逆整数変換、または概念的に同様の逆変換プロセスを変換係数に適用する。

10

【0339】

[0240] 動き補償ユニット72が、動きベクトルと他のシンタックス要素とに基づいて現在ビデオブロックのための予測ブロックを発生した後に、ビデオデコーダ30は、逆変換処理ユニット78からの残差ブロックを動き補償ユニット72によって生成された対応する予測ブロックと加算することによって、復号ビデオブロックを形成する。加算器80は、この加算演算を実行する1つまたは複数のコンポーネントを表す。所望される場合、プロッキネスアーティファクトを除去するために、復号ブロックをフィルタ処理するためにデブロッキングフィルタも適用され得る。ピクセル遷移を平滑化するか、または場合によってはビデオ品質を改善するために、(コーディンググループ中またはコーディンググループ後のいずれかの)他のループフィルタも使用され得る。所与のフレームまたはピクチャ中の復号ビデオブロックは、次いで、後続の動き補償のために使用される参照ピクチャを記憶する復号ピクチャバッファ82に記憶される。復号ピクチャバッファ82はまた、図1のディスプレイデバイス32などのディスプレイデバイス上での後の表示のための復号ビデオを記憶する。

20

【0340】

[0241] 図12は、本開示の1つの例示的なビデオ処理技法を示すフローチャートである。図12の技法は、ビデオエンコーダ20および/またはビデオプリプロセッサユニット19によって実行され得る。図12の例では、ソースデバイス12は、カメラを使用してビデオデータをキャプチャする(1200)ように構成され得る。ビデオエンコーダ20および/またはビデオプリプロセッサユニット19は、固定小数点算出を使用して、ビデオデータに対してダイナミックレンジ調整プロセスを実行する(1210)ように構成され得る。ビデオエンコーダ20および/またはビデオプリプロセッサユニット19は、固定小数点算出を使用して、ダイナミックレンジ調整プロセスに対して、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を発生する(1220)ようにさらに構成され得る。

30

【0341】

[0242] 一例では、ビデオエンコーダ20および/またはビデオプリプロセッサユニット19は、1つまたは複数の補足エンハンスメント情報(SEI)メッセージ中で1つまたは複数のシンタックス要素を発生することによって、1つまたは複数のシンタックス要素を発生するように構成され得る。一例では、パラメータは、レンジパラメータ、スケールパラメータ、またはオフセットパラメータのうちの1つまたは複数を備える。別の例では、情報は、レンジパラメータを決定するために使用される小数ビットの第1の数、スケールパラメータを決定するために使用される小数ビットの第2の数、およびオフセットパラメータを決定するために使用される小数ビットの第3の数のうちの1つまたは複数を示す。別の例では、情報は、ビデオデータの1つまたは複数の色コンポーネントのための最小値と最大値とを含む。別の例では、情報は、復号ビデオデータのサンプル値の所定のレンジ(predetermined range)に対するインデックスを含む。

40

50

## 【0342】

[0243] 図13は、本開示の別の例的なビデオ処理技法を示すフローチャートである。図13の技法は、ビデオデコーダ30および/またはビデオポストプロセッサユニット31によって実行され得る。本開示の一例では、ビデオデコーダ30および/またはビデオポストプロセッサユニット31は、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を受信し(1300)、復号ビデオデータを受信する(1310)ように構成され得る。

## 【0343】

[0244] ビデオデコーダ30および/またはビデオポストプロセッサユニット31は、受信された情報から逆ダイナミックレンジ調整プロセスのためのパラメータを決定し(1320)、受信された情報と決定されたパラメータとに従って、固定小数点算出を使用して、復号ビデオデータに対して逆ダイナミックレンジ調整プロセスを実行する(1330)ようにさらに構成され得る。宛先デバイス14は、復号ビデオデータに対して逆ダイナミックレンジ調整プロセスを実行した後に復号ビデオデータを表示する(display)(1340)のようにさらに構成され得る。

10

## 【0344】

[0245] 本開示の一例では、ビデオデコーダ30および/またはビデオポストプロセッサユニット31は、1つまたは複数の補足エンハンスメント情報(SEI)メッセージ中で1つまたは複数のシンタックス要素を受信するように構成され得る。一例では、パラメータは、レンジパラメータ、スケールパラメータ、またはオフセットパラメータのうちの1つまたは複数を備える。別の例では、情報は、レンジパラメータを決定するために使用される小数ビットの第1の数、スケールパラメータを決定するために使用される小数ビットの第2の数、およびオフセットパラメータを決定するために使用される小数ビットの第3の数のうちの1つまたは複数を示す。

20

## 【0345】

[0246] 本開示の別の例では、ビデオデコーダ30および/またはビデオポストプロセッサユニット31は、小数ビットの第1の数、小数ビットの第2の数、または小数ビットの第3の数のうちの少なくとも1つが互いに異なる場合、パラメータを決定するために使用される任意の中間計算プロセス(intermediate calculation process)中に任意の小数ビットを累算することによって、パラメータを決定し、所定の小数正確さ(predetermined fractional accuracy)に基づいて、パラメータを決定するための最終結果(final result)をクリッピングするように構成され得る。

30

## 【0346】

[0247] 本開示の別の例では、ビデオデコーダ30および/またはビデオポストプロセッサユニット31は、パラメータを決定するために使用されるすべての中間計算プロセス中に所望の小数正確さを越える小数ビットをトランケートすることによってパラメータを決定するように構成され得る。

## 【0347】

[0248] 別の例では、情報は、復号ビデオデータの1つまたは複数の色コンポーネントのための最小値と最大値とを含み、ビデオデコーダ30および/またはビデオポストプロセッサユニット31は、受信された最小値と受信された最大値とに基づいてパラメータを決定するように構成され得る。

40

## 【0348】

[0249] 別の例では、情報は、復号ビデオデータの1つまたは複数の色コンポーネントのためのサンプル値の所定のレンジに対するインデックスを含み、ビデオデコーダ30および/またはビデオポストプロセッサユニット31は、受信されたインデックスに基づいて復号ビデオデータの1つまたは複数の色コンポーネントのための最小値と最大値とを決定し、決定された最小値と決定された最大値とに基づいてパラメータを決定するように構成され得る。

50

## 【0349】

[0250] 本開示の別の例では、ビデオデコーダ30および/またはビデオポストプロセッサユニット31は、パラメータが符号付きであるのか符号なしであるのかを示すシンタックス要素を受信し、S E Iメッセージ中の情報に対してパースプロセスを実行するように構成され得、ここにおいて、パースプロセスは、シンタックス要素の値にかかわらず同じである。

## 【0350】

[0251] 本開示のいくつかの態様は、説明の目的でH E V C規格の拡張に関して説明された。ただし、本開示で説明された技法は、まだ開発されていない他の規格またはプロプライエタリビデオコーディングプロセスを含む、他のビデオコーディングプロセスのために有用であり得る。

10

## 【0351】

[0252] 本開示で説明されたビデオコーダは、ビデオエンコーダまたはビデオデコーダを指し得る。同様に、ビデオコーディングユニットは、ビデオエンコーダまたはビデオデコーダを指し得る。同様に、ビデオコーディングは、適用可能なとき、ビデオ符号化またはビデオ復号を指し得る。

## 【0352】

[0253] 上記例に応じて、本明細書で説明された技法のいずれかのいくつかの行為またはイベントが、異なるシーケンスで実行され得、全体的に追加、マージ、または除外され得る（たとえば、すべての説明された行為またはイベントが本技法の実施のために必要であるとは限らない）ことを認識されたい。その上、いくつかの例では、行為またはイベントは、連続的にではなく、たとえば、マルチスレッド処理、割込み処理、または複数のプロセッサを通して同時に実行され得る。

20

## 【0353】

[0254] 1つまたは複数の例では、説明された機能は、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合せで実装され得る。ソフトウェアで実装される場合、機能は、1つまたは複数の命令またはコードとして、コンピュータ可読媒体上に記憶されるか、あるいはコンピュータ可読媒体を介して送信され、ハードウェアベース処理ユニットによって実行され得る。コンピュータ可読媒体は、データ記憶媒体などの有形媒体に対応する、コンピュータ可読記憶媒体を含み得るか、または、たとえば、通信プロトコルに従って、ある場所から別の場所へのコンピュータプログラムの転送を可能にする任意の媒体を含む通信媒体を含み得る。このようにして、コンピュータ可読媒体は、概して、（1）非一時的である有形コンピュータ可読記憶媒体、あるいは（2）信号または搬送波などの通信媒体に対応し得る。データ記憶媒体は、本開示で説明された技法の実装のための命令、コードおよび/またはデータ構造を取り出すために、1つまたは複数のコンピュータあるいは1つまたは複数のプロセッサによってアクセスされ得る、任意の利用可能な媒体であり得る。コンピュータプログラム製品はコンピュータ可読媒体を含み得る。

30

## 【0354】

[0255] 限定ではなく例として、そのようなコンピュータ可読記憶媒体は、R A M、R O M、E E P R O M（登録商標）、C D - R O Mまたは他の光ディスクストレージ、磁気ディスクストレージ、または他の磁気ストレージデバイス、フラッシュメモリ、あるいは命令またはデータ構造の形態の所望のプログラムコードを記憶するために使用され得、コンピュータによってアクセスされ得る、任意の他の媒体を備えることができる。また、いかなる接続もコンピュータ可読媒体と適切に呼ばれる。たとえば、命令が、同軸ケーブル、光ファイバーケーブル、ツイストペア、デジタル加入者線（D S L）、または赤外線、無線、およびマイクロ波などのワイヤレス技術を使用してウェブサイト、サーバ、または他のリモートソースから送信される場合、同軸ケーブル、光ファイバーケーブル、ツイストペア、D S L、または赤外線、無線、およびマイクロ波などのワイヤレス技術は媒体の定義に含まれる。ただし、コンピュータ可読記憶媒体およびデータ記憶媒体は、接続、搬送波、信号、または他の一時的媒体を含まないが、代わりに非一時的有形記憶媒体を対象

40

50

とすることを理解されたい。本明細書で使用されるディスク (disk) およびディスク (disc) は、コンパクトディスク (disc) (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 (登録商標) ディスク (disk) および Blu-ray ディスク (disc) を含み、ここで、ディスク (disk) は、通常、データを磁気的に再生し、ディスク (disc) は、データをレーザーで光学的に再生する。上記の組合せもコンピュータ可読媒体の範囲内に含まれるべきである。

【0355】

[0256] 命令は、1つまたは複数のデジタル信号プロセッサ (DSP) などの1つまたは複数のプロセッサ、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)、あるいは他の等価な集積回路またはディスクリート論理回路によって実行され得る。したがって、本明細書で使用される「プロセッサ」という用語は、上記の構造、または本明細書で説明された技法の実装に好適な他の構造のいずれかを指し得る。さらに、いくつかの態様では、本明細書で説明された機能は、符号化および復号のために構成された専用ハードウェアおよび/またはソフトウェアモジュール内に与えられるか、あるいは複合コーデックに組み込まれ得る。また、本技法は、1つまたは複数の回路または論理要素で十分に実装され得る。

【0356】

[0257] 本開示の技法は、ワイヤレスハンドセット、集積回路 (IC) またはICのセット (たとえば、チップセット) を含む、多種多様なデバイスまたは装置で実装され得る。本開示では、開示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、様々なコンポーネント、モジュール、またはユニットが説明されたが、それらのコンポーネント、モジュール、またはユニットは、必ずしも異なるハードウェアユニットによる実現を必要とするとは限らない。むしろ、上記で説明されたように、様々なユニットが、好適なソフトウェアおよび/またはファームウェアとともに、上記で説明された1つまたは複数のプロセッサを含めて、コーデックハードウェアユニットにおいて組み合わせられるか、または相互動作可能なハードウェアユニットの集合によって与えられ得る。

【0357】

[0258] 様々な例が説明された。これらおよび他の例は、以下の特許請求の範囲内に入る。

以下に、本願の出願当初の特許請求の範囲に記載された発明を付記する。

[C1]

ビデオデータを処理する方法であって、前記方法が、

逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を受信することと、  
復号ビデオデータを受信することと、

受信された前記情報に従って、固定小数点算出を使用して、前記復号ビデオデータに対して前記逆ダイナミックレンジ調整プロセスを実行することと  
を備える、方法。

[C2]

前記1つまたは複数のシンタックス要素を受信することが、1つまたは複数の補足エンハンスマント情報 (SEI) メッセージ中で前記1つまたは複数のシンタックス要素を受信することを備え、ここにおいて、前記パラメータが、レンジパラメータ、スケールパラメータ、またはオフセットパラメータのうちの1つまたは複数を備える、C1に記載の方法。

[C3]

前記情報が、前記レンジパラメータを決定するために使用される小数ビットの第1の数、前記スケールパラメータを決定するために使用される小数ビットの第2の数、および前記オフセットパラメータを決定するために使用される小数ビットの第3の数のうちの1つ

10

20

30

40

50

または複数を示す、C 2 に記載の方法。

[ C 4 ]

小数ビットの前記第1の数と、小数ビットビットの前記第2の数と、小数ビットの前記第3の数とを使用して前記パラメータを決定すること  
をさらに備える、C 3 に記載の方法。

[ C 5 ]

前記パラメータを決定することは、

小数ビットの前記第1の数、小数ビットの前記第2の数、または小数ビットの前記第3の数のうちの少なくとも1つが互いに異なる場合、前記パラメータを決定するために使用される任意の中間計算プロセス中に任意の小数ビットを累算することによって、前記パラメータを決定することと、

所定の小数正確さに基づいて、前記パラメータを決定するために最終結果をクリッピングすることと

をさらに備える、C 4 に記載の方法。

[ C 6 ]

前記パラメータを決定することが、

前記パラメータを決定するために使用されるすべての中間計算プロセス中に所望の小数正確さを越える任意の小数ビットをトランケートすることによって前記パラメータを決定すること

をさらに備える、C 4 に記載の方法。

10

[ C 7 ]

前記情報が、前記復号ビデオデータの1つまたは複数の色コンポーネントのための最小値と最大値とを含み、前記方法が、

前記受信された最小値と前記受信された最大値とに基づいて前記パラメータを決定すること

をさらに備える、C 1 に記載の方法。

[ C 8 ]

前記情報が、前記復号ビデオデータの1つまたは複数の色コンポーネントのためのサンプル値の所定のレンジに対するインデックスを含み、前記方法が、

前記受信されたインデックスに基づいて、前記復号ビデオデータの前記1つまたは複数の色コンポーネントのための最小値と最大値とを決定することと、

前記決定された最小値と前記決定された最大値とに基づいて前記パラメータを決定することと

をさらに備える、C 1 に記載の方法。

20

[ C 9 ]

前記パラメータが符号付きであるのか符号なしであるのかを示すシンタックス要素を受信することと、

前記S E Iメッセージ中の前記情報に対してパースプロセスを実行することと、ここにおいて、前記パースプロセスが、前記シンタックス要素の値にかかわらず同じである、をさらに備える、C 1 に記載の方法。

30

[ C 10 ]

前記復号ビデオデータに対して前記逆ダイナミックレンジ調整プロセスを実行した後に前記復号ビデオデータを表示すること

をさらに備える、C 1 に記載の方法。

40

[ C 11 ]

ビデオデータを処理するように構成された装置であって、前記装置が、

復号ビデオデータを記憶するように構成されたメモリと、

1つまたは複数のプロセッサと

を備え、前記1つまたは複数のプロセッサが、

逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定す

50

べきかを指定する情報を含んでいる 1 つまたは複数のシンタックス要素を受信することと

前記復号ビデオデータを受信することと、

受信された前記情報に従って、固定小数点算出を使用して、前記復号ビデオデータに  
対して前記逆ダイナミックレンジ調整プロセスを実行することと  
を行うように構成された、装置。

[ C 1 2 ]

前記 1 つまたは複数のプロセッサが、1 つまたは複数の補足エンハンスメント情報 (S  
E I ) メッセージ中で前記 1 つまたは複数のシンタックス要素を受信することを行うよう  
にさらに構成されたことを受信し、ここにおいて、前記パラメータが、レンジパラメータ  
、スケールパラメータ、またはオフセットパラメータのうちの 1 つまたは複数を備える、  
C 1 1 に記載の装置。

[ C 1 3 ]

前記情報が、前記レンジパラメータを決定するために使用される小数ビットの第 1 の数  
、前記スケールパラメータを決定するために使用される小数ビットの第 2 の数、および前  
記オフセットパラメータを決定するために使用される小数ビットの第 3 の数のうちの 1 つ  
または複数を示す、C 1 2 に記載の装置。

[ C 1 4 ]

前記 1 つまたは複数のプロセッサが、

小数ビットの前記第 1 の数と、部分ビットの前記第 2 の数と、小数ビットの前記第 3 の  
数とを使用して前記パラメータを決定すること  
を行うようにさらに構成された、C 1 3 に記載の装置。

[ C 1 5 ]

前記パラメータを決定するために、前記 1 つまたは複数のプロセッサは、

小数ビットの前記第 1 の数、小数ビットの前記第 2 の数、または小数ビットの前記第 3  
の数のうちの少なくとも 1 つが互いに異なる場合、前記パラメータを決定するために使用  
される任意の中間計算プロセス中に任意の小数ビットを累算することによって、前記パラ  
メータを決定することと、

所定の小数正確さに基づいて、前記パラメータを決定するための最終結果をクリッピング  
することと

を行うようにさらに構成された、C 1 4 に記載の装置。

[ C 1 6 ]

前記パラメータを決定するために、前記 1 つまたは複数のプロセッサが、

前記パラメータを決定するために使用されるすべての中間計算プロセス中に所望の小数  
正確さを越える任意の小数ビットをトランケートすることによって前記パラメータを決定  
すること

を行うようにさらに構成された、C 1 4 に記載の装置。

[ C 1 7 ]

前記情報が、前記復号ビデオデータの 1 つまたは複数の色コンポーネントのための最小  
値と最大値とを含み、ここにおいて、前記 1 つまたは複数のプロセッサが、

前記受信された最小値と前記受信された最大値とに基づいて前記パラメータを決定する  
こと

を行うようにさらに構成された、C 1 1 に記載の装置。

[ C 1 8 ]

前記情報が、前記復号ビデオデータの 1 つまたは複数の色コンポーネントのサンプル値  
の所定のレンジに対するインデックスを含み、ここにおいて、前記 1 つまたは複数のプロ  
セッサが、

前記受信されたインデックスに基づいて、前記復号ビデオデータの前記 1 つまたは複数  
の色コンポーネントのための最小値と最大値とを決定することと、

前記決定された最小値と前記決定された最大値とに基づいて前記パラメータを決定する

10

20

30

40

50

ことと

を行うようにさらに構成された、C 1 1 に記載の装置。

[ C 1 9 ]

前記 1 つまたは複数のプロセッサが、

前記パラメータが符号付きであるのか符号なしであるのかを示すシンタックス要素を受信することと、

前記 S E I メッセージ中の前記情報に対してパースプロセスを実行することと、ここにおいて、前記パースプロセスが、前記シンタックス要素の値にかかわらず同じである、を行うようにさらに構成された、C 1 1 に記載の装置。

[ C 2 0 ]

前記装置は、

前記 1 つまたは複数のプロセッサが、前記復号ビデオデータに対して前記逆ダイナミックレンジ調整プロセスを実行した後に、前記復号ビデオデータを表示するように構成されたディスプレイ

をさらに備える、C 1 1 に記載の装置。

[ C 2 1 ]

前記装置が、カメラ、コンピュータ、モバイルデバイス、ブロードキャスト受信機デバイス、またはセットトップボックスのうちの 1 つまたは複数を備える、C 1 1 に記載の装置。

[ C 2 2 ]

ビデオデータを処理するように構成された装置であって、前記装置が、

逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる 1 つまたは複数のシンタックス要素を受信するための手段と、

復号ビデオデータを受信するための手段と、

受信された前記情報に従って、固定小数点算出を使用して、前記復号ビデオデータに対して前記逆ダイナミックレンジ調整プロセスを実行するための手段と備える、装置。

[ C 2 3 ]

前記 1 つまたは複数のシンタックス要素を受信するための前記手段が、1 つまたは複数の補足エンハンスメント情報 (S E I) メッセージ中で前記 1 つまたは複数のシンタックス要素を受信するための手段を備え、ここにおいて、前記パラメータが、レンジパラメータ、スケールパラメータ、またはオフセットパラメータのうちの 1 つまたは複数を備える、C 2 2 に記載の装置。

[ C 2 4 ]

前記情報が、前記レンジパラメータを決定するために使用される小数ビットの第 1 の数、前記スケールパラメータを決定するために使用される小数ビットの第 2 の数、および前記オフセットパラメータを決定するために使用される小数ビットの第 3 の数のうちの 1 つまたは複数を示す、C 2 3 に記載の装置。

[ C 2 5 ]

小数ビットの前記第 1 の数と、部分ビットの前記第 2 の数と、小数ビットの前記第 3 の数とを使用して前記パラメータを決定するための手段をさらに備える、C 2 4 に記載の装置。

[ C 2 6 ]

前記パラメータを決定するための前記手段は、

小数ビットの前記第 1 の数、小数ビットの前記第 2 の数、または小数ビットの前記第 3 の数のうちの少なくとも 1 つが互いに異なる場合、前記パラメータを決定するために使用される任意の中間計算プロセス中に任意の小数ビットを累算することによって、前記パラメータを決定するための手段と、

所定の小数正確さに基づいて、前記パラメータを決定するための最終結果をクリッピング

10

20

30

40

50

グするための手段と

をさらに備える、C 2 5 に記載の装置。

[ C 2 7 ]

前記パラメータを決定するための前記手段が、

前記パラメータを決定するために使用されるすべての中間計算プロセス中に所望の小数正確さを越える任意の小数ビットをトランケートすることによって前記パラメータを決定するための手段

をさらに備える、C 2 5 に記載の装置。

[ C 2 8 ]

前記情報が、前記復号ビデオデータの1つまたは複数の色コンポーネントのための最小値と最大値とを含み、前記装置が、

前記受信された最小値と前記受信された最大値とに基づいて前記パラメータを決定するための手段

をさらに備える、C 2 2 に記載の装置。

[ C 2 9 ]

前記情報が、前記復号ビデオデータのサンプル値または1つまたは複数の色コンポーネントの所定のレンジに対するインデックスを含み、前記装置が、

前記受信されたインデックスに基づいて、前記復号ビデオデータの前記1つまたは複数の色コンポーネントのための最小値と最大値とを決定するための手段と、

前記決定された最小値と前記決定された最大値とに基づいて前記パラメータを決定するための手段と

をさらに備える、C 2 2 に記載の装置。

[ C 3 0 ]

前記パラメータが符号付きであるのか符号なしであるのかを示すシンタックス要素を受信するための手段と、

前記S E Iメッセージ中の前記情報に対してパースプロセスを実行するための手段と、ここにおいて、前記パースプロセスが、前記シンタックス要素の値にかかわらず同じである、

をさらに備える、C 2 2 に記載の装置。

[ C 3 1 ]

前記復号ビデオデータに対して前記逆ダイナミックレンジ調整プロセスを実行した後に前記復号ビデオデータを表示するための手段

をさらに備える、C 2 2 に記載の装置。

[ C 3 2 ]

実行されたとき、ビデオデータを処理するように構成されたデバイスの1つまたは複数のプロセッサに、

逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を受信することと、

前記復号ビデオデータを受信することと、

受信された前記情報に従って、固定小数点算出を使用して、前記復号ビデオデータに対して前記逆ダイナミックレンジ調整プロセスを実行することと

を行わせる命令を記憶するコンピュータ可読記憶媒体。

[ C 3 3 ]

前記命令が、前記1つまたは複数のプロセッサに、1つまたは複数の補足エンハンスマント情報(S E I)メッセージ中で前記1つまたは複数のシンタックス要素を受信することをさらに行わせ、ここにおいて、前記パラメータが、レンジパラメータ、スケールパラメータ、またはオフセットパラメータのうちの1つまたは複数を備える、C 3 2 に記載のコンピュータ可読記憶媒体。

[ C 3 4 ]

前記情報が、前記レンジパラメータを決定するために使用される小数ビットの第1の数

10

20

30

40

50

、前記スケールパラメータを決定するために使用される小数ビットの第2の数、および前記オフセットパラメータを決定するために使用される小数ビットの第3の数のうちの1つまたは複数を示す、C33に記載のコンピュータ可読記憶媒体。

[C35]

前記命令が、前記1つまたは複数のプロセッサに、  
小数ビットの前記第1の数と、部分ビットの前記第2の数と、小数ビットの前記第3の数とを使用して前記パラメータを決定すること  
をさらに行わせる、C34に記載のコンピュータ可読記憶媒体。

[C36]

前記命令は、前記1つまたは複数のプロセッサに、  
小数ビットの前記第1の数、小数ビットの前記第2の数、または小数ビットの前記第3の数のうちの少なくとも1つが互いに異なる場合、前記パラメータを決定するために使用される任意の中間計算プロセス中に任意の小数ビットを累算することによって、前記パラメータを決定することと、  
所定の小数正確さに基づいて、前記パラメータを決定するための最終結果をクリッピングすることと  
をさらに行わせる、C35に記載のコンピュータ可読記憶媒体。

[C37]

前記命令が、前記1つまたは複数のプロセッサに、  
前記パラメータを決定するために使用されるすべての中間計算プロセス中に所望の小数正確さを越える任意の小数ビットをトランケートすることによって前記パラメータを決定すること  
をさらに行わせる、C35に記載のコンピュータ可読記憶媒体。

[C38]

前記情報が、前記復号ビデオデータの1つまたは複数の色コンポーネントのための最小値と最大値とを含み、ここにおいて、前記命令が、前記1つまたは複数のプロセッサに、  
前記受信された最小値と前記受信された最大値とに基づいて前記パラメータを決定すること  
をさらに行わせる、C32に記載のコンピュータ可読記憶媒体。

[C39]

前記情報が、前記復号ビデオデータの1つまたは複数の色コンポーネントのサンプル値の所定のレンジに対するインデックスを含み、ここにおいて、前記命令が、前記1つまたは複数のプロセッサに、

前記受信されたインデックスに基づいて、前記復号ビデオデータの前記1つまたは複数の色コンポーネントのための最小値と最大値とを決定することと、  
前記決定された最小値と前記決定された最大値とに基づいて前記パラメータを決定すること  
をさらに行わせる、C32に記載のコンピュータ可読記憶媒体。

[C40]

前記命令が、前記1つまたは複数のプロセッサに、  
前記パラメータが符号付きであるのか符号なしであるのかを示すシンタックス要素を受信することと、  
前記S E Iメッセージ中の前記情報に対してパースプロセスを実行することと、ここにおいて、前記パースプロセスが、前記シンタックス要素の値にかかわらず同じである、をさらに行わせる、C32に記載のコンピュータ可読記憶媒体。

[C41]

前記命令が、前記1つまたは複数のプロセッサに、  
前記復号ビデオデータに対して前記逆ダイナミックレンジ調整プロセスを実行した後に前記復号ビデオデータを表示すること  
をさらに行わせる、C32に記載のコンピュータ可読記憶媒体。

10

20

30

40

50

## [ C 4 2 ]

ビデオデータを処理する方法であって、前記方法が、  
固定小数点算出を使用してビデオデータに対してダイナミックレンジ調整プロセスを実行することと、  
固定小数点算出を使用して、前記ダイナミックレンジ調整プロセスに対して、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を発生することと、  
を備える、方法。

## [ C 4 3 ]

前記1つまたは複数のシンタックス要素を発生することが、1つまたは複数の補足エンハンスマント情報（S E I）メッセージ中で前記1つまたは複数のシンタックス要素を発生することを備え、ここにおいて、前記パラメータが、レンジパラメータ、スケールパラメータ、またはオフセットパラメータのうちの1つまたは複数を備える、C 4 2に記載の方法。

10

## [ C 4 4 ]

前記情報が、前記レンジパラメータを決定するために使用される小数ビットの第1の数、前記スケールパラメータを決定するために使用される小数ビットの第2の数、および前記オフセットパラメータを決定するために使用される小数ビットの第3の数のうちの1つまたは複数を示す、C 4 3に記載の方法。

## [ C 4 5 ]

前記情報が、前記ビデオデータの1つまたは複数の色コンポーネントのための最小値と最大値とを含む、C 4 2に記載の方法。

20

## [ C 4 6 ]

前記情報が、前記復号ビデオデータのサンプル値の所定のレンジに対するインデックスを含む、C 4 2に記載の方法。

## [ C 4 7 ]

カメラを用いて前記ビデオデータをキャプチャすることをさらに備える、C 4 2に記載の方法。

## [ C 4 8 ]

ビデオデータを処理するように構成された装置であって、前記装置が、  
ビデオデータを記憶するように構成されたメモリと、  
1つまたは複数のプロセッサと  
を備え、前記1つまたは複数のプロセッサが、  
固定小数点算出を使用して前記ビデオデータに対してダイナミックレンジ調整プロセスを実行することと、  
固定小数点算出を使用して、前記ダイナミックレンジ調整プロセスに対して、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる1つまたは複数のシンタックス要素を発生することと、  
を行うように構成された、装置。

30

## [ C 4 9 ]

前記1つまたは複数のプロセッサが、1つまたは複数の補足エンハンスマント情報（S E I）メッセージ中で前記1つまたは複数のシンタックス要素を発生することを行うように構成され、ここにおいて、前記パラメータが、レンジパラメータ、スケールパラメータ、またはオフセットパラメータのうちの1つまたは複数を備える、C 4 8に記載の装置。

40

## [ C 5 0 ]

前記情報が、前記レンジパラメータを決定するために使用される小数ビットの第1の数、前記スケールパラメータを決定するために使用される小数ビットの第2の数、および前記オフセットパラメータを決定するために使用される小数ビットの第3の数のうちの1つまたは複数を示す、C 4 9に記載の装置。

50

[ C 5 1 ]

前記情報が、前記ビデオデータの 1 つまたは複数の色コンポーネントのための最小値と最大値とを含む、C 4 8 に記載の装置。

[ C 5 2 ]

前記情報が、前記復号ビデオデータのサンプル値の所定のレンジに対するインデックスを含む、C 4 8 に記載の装置。

[ C 5 3 ]

前記装置が、

前記ビデオデータをキャプチャするように構成されたカメラをさらに備える、C 4 8 に記載の装置。

10

[ C 5 4 ]

前記装置が、カメラ、コンピュータ、モバイルデバイス、ブロードキャスト受信機デバイス、またはセットトップボックスのうちの 1 つまたは複数を備える、C 4 8 に記載の装置。

[ C 5 5 ]

ビデオデータを処理するように構成された装置であって、前記装置が、

固定小数点算出を使用してビデオデータに対してダイナミックレンジ調整プロセスを実行するための手段と、

固定小数点算出を使用して、前記ダイナミックレンジ調整プロセスに対して、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる 1 つまたは複数のシンタックス要素を発生するための手段とを備える、装置。

20

[ C 5 6 ]

実行されたとき、ビデオデータを処理するように構成されたデバイスの 1 つまたは複数のプロセッサに、

固定小数点算出を使用して前記ビデオデータに対してダイナミックレンジ調整プロセスを実行することと、

固定小数点算出を使用して、前記ダイナミックレンジ調整プロセスに対して、逆ダイナミックレンジ調整プロセスを実行するためのパラメータをどのように決定すべきかを指定する情報を含んでいる 1 つまたは複数のシンタックス要素を発生することとを行わせる命令を記憶するコンピュータ可読記憶媒体。

30

【図1】

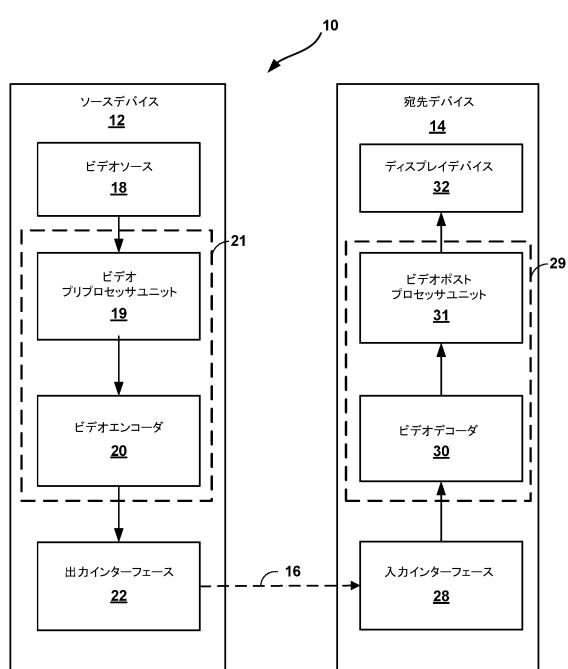


FIG. 1

【図2】

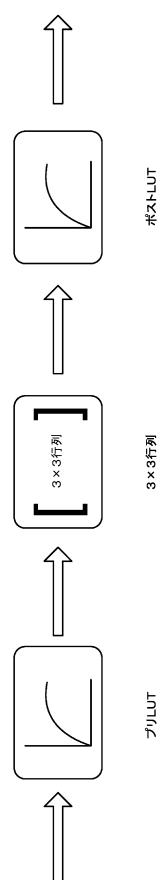


FIG. 2

【図3】

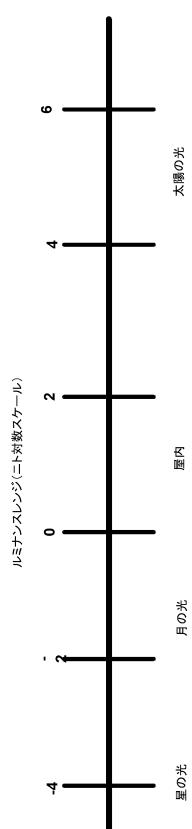


FIG. 3

【図4】

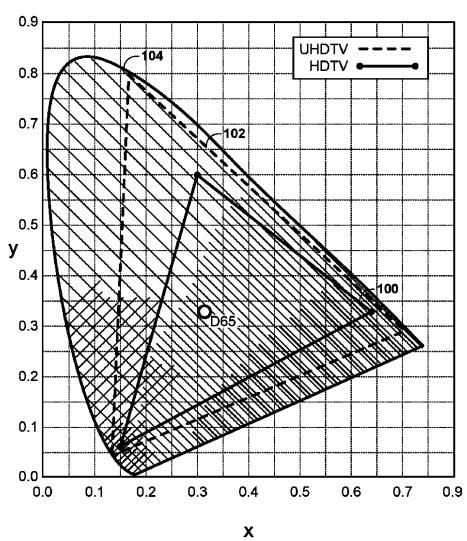
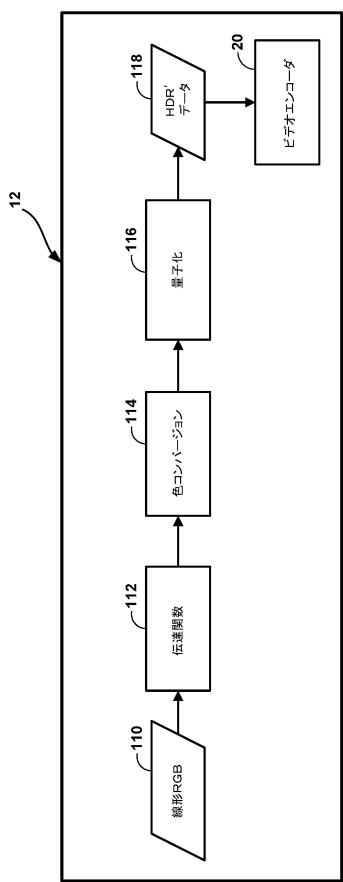


FIG. 4

【図5】



【図6】

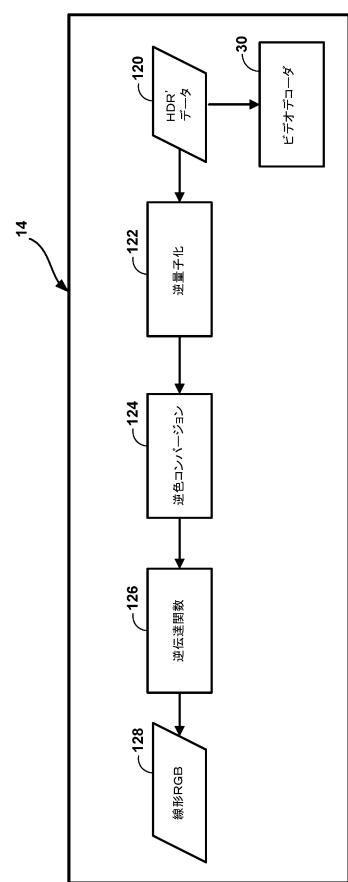


FIG. 5

FIG. 6

【図7】

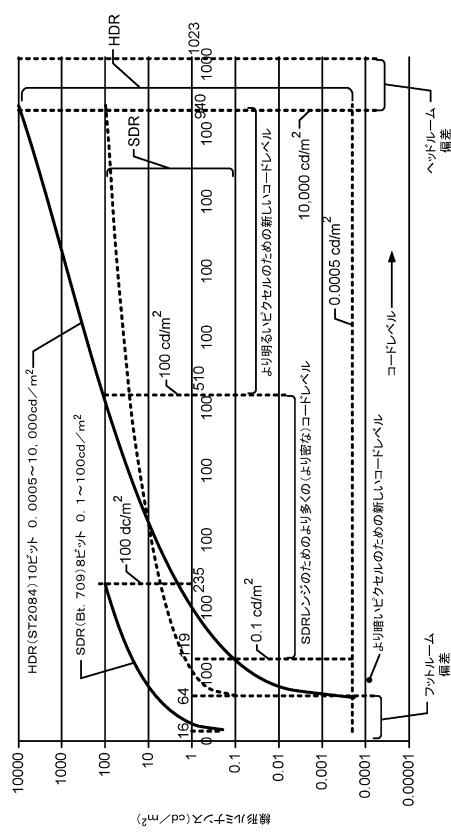


FIG. 7

【図8】

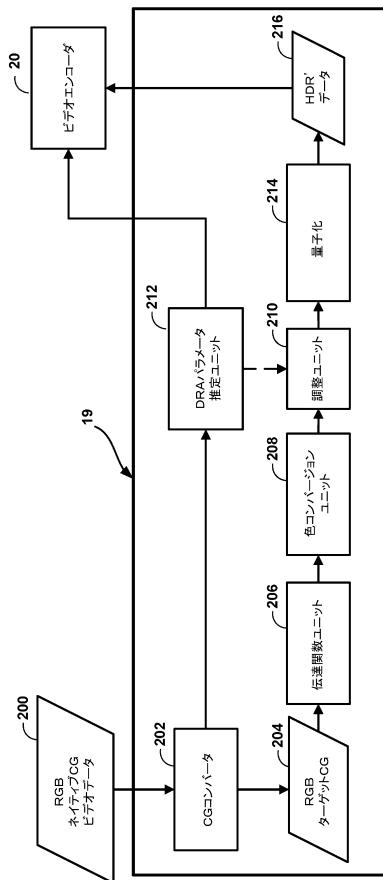


FIG. 8

【 図 9 】

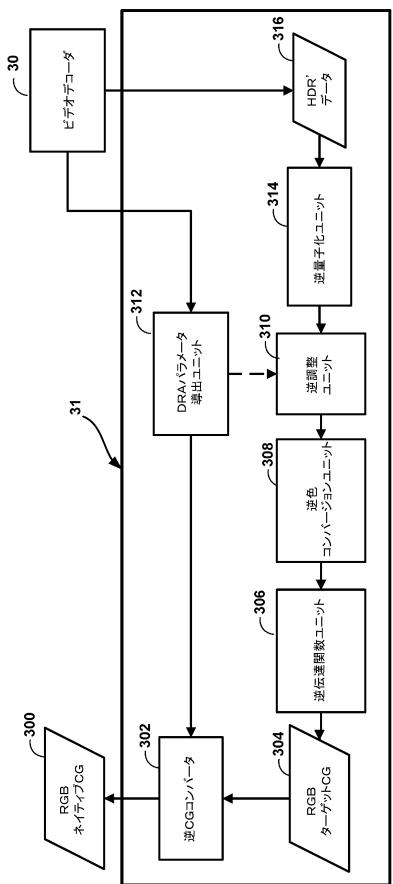


FIG. 9

【図10】

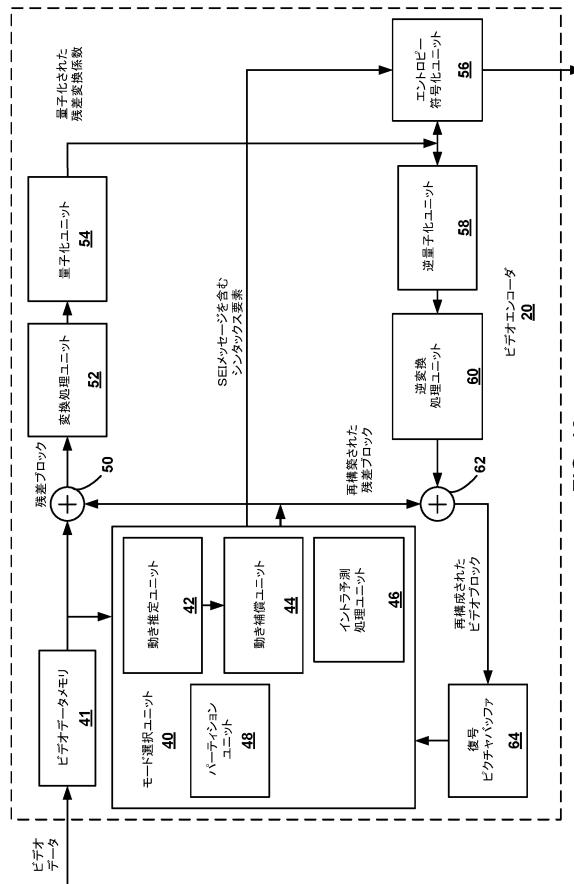


FIG. 10

## 【 図 1 1 】

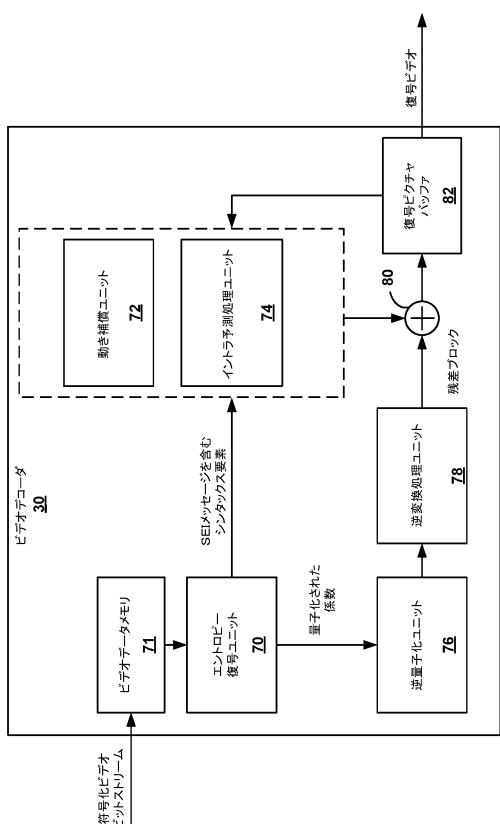


FIG. 11

【図12】

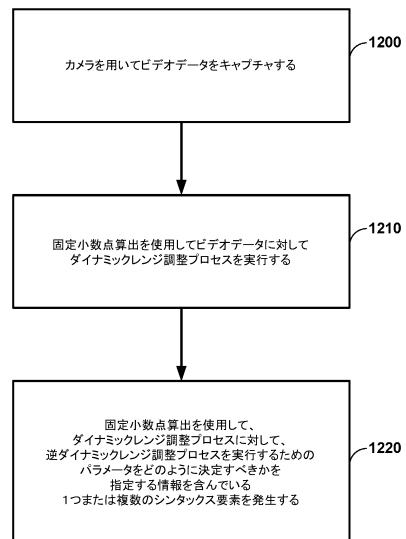


FIG. 12

【図13】

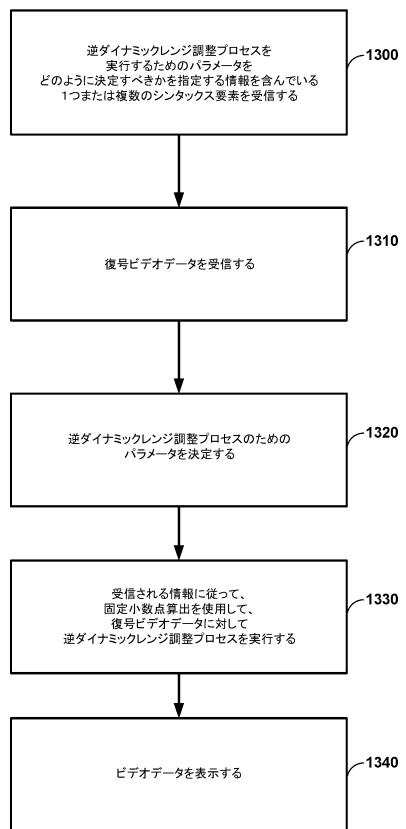


FIG. 13

---

フロントページの続き

(31)優先権主張番号 62/241,063

(32)優先日 平成27年10月13日(2015.10.13)

(33)優先権主張国・地域又は機関

米国(US)

(31)優先権主張番号 15/269,558

(32)優先日 平成28年9月19日(2016.9.19)

(33)優先権主張国・地域又は機関

米国(US)

(74)代理人 100184332

弁理士 中丸 廉洋

(72)発明者 ラマスプラモニアン、アダルシュ・クリシュナン

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

(72)発明者 ルサノフスキー、ドミトロ

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

(72)発明者 ソール・ロジャルス、ジョエル

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

(72)発明者 リ、ソンウォン

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

(72)発明者 ブーダイジュ・シャンスル、ドネ

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

(72)発明者 カルチェビチ、マルタ

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

審査官 岩井 健二

(56)参考文献 國際公開第2015/033809 (WO, A1)

國際公開第2013/067101 (WO, A1)

D. Bugdayci Sansli et al., Dynamic Range Adjustment SEI Message, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 21st Meeting: Warsaw, PL, 2015年 6月, JCTVC-U0098\_r1, pp. 1-6

A. Dsouza et al., Adaptive Gamut Expansion for Chroma Components, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 23rd Meeting: San Diego, USA, 2016年 2月, JCTVC-W0040, pp. 1-9

E. Francois, C. Chevance and Y. Olivier, HDR CE2: CE6-4.6b report on usage of SEI for Test Model, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 23rd Meeting: San Diego, USA, 2016年 2月, JCTVC-W0059\_r1, pp. 1-6

Done Bugdayci Sansli et al., HDR CE6: Test 4.1 Reshaper from m37064, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, 23rd Meeting: San Diego, USA, 2016年 2月, JCTVC-W0103\_r1, pp. 1-11

(58)調査した分野(Int.Cl. , DB名)

H 04N 19/00 - 19/98