

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6334830号
(P6334830)

(45) 発行日 平成30年5月30日(2018.5.30)

(24) 登録日 平成30年5月11日(2018.5.11)

(51) Int. Cl.	F I
HO 4 N 19/119 (2014.01)	HO 4 N 19/119
HO 4 N 19/14 (2014.01)	HO 4 N 19/14
HO 4 N 19/147 (2014.01)	HO 4 N 19/147
HO 4 N 19/176 (2014.01)	HO 4 N 19/176

請求項の数 21 (全 45 頁)

(21) 出願番号	特願2017-546226 (P2017-546226)	(73) 特許権者	507364838
(86) (22) 出願日	平成28年3月3日(2016.3.3)		クアルコム、インコーポレイテッド
(65) 公表番号	特表2018-511231 (P2018-511231A)		アメリカ合衆国 カリフォルニア 921
(43) 公表日	平成30年4月19日(2018.4.19)		21 サン ディエゴ モアハウス ドラ
(86) 国際出願番号	PCT/US2016/020684		イブ 5775
(87) 国際公開番号	W02016/144695	(74) 代理人	100108453
(87) 国際公開日	平成28年9月15日(2016.9.15)		弁理士 村山 靖彦
審査請求日	平成30年3月9日(2018.3.9)	(74) 代理人	100163522
(31) 優先権主張番号	62/129,545		弁理士 黒田 晋平
(32) 優先日	平成27年3月6日(2015.3.6)	(72) 発明者	チェンジエ・ツ
(33) 優先権主張国	米国 (US)		アメリカ合衆国・カリフォルニア・921
(31) 優先権主張番号	14/737,271		21-1714・サン・ディエゴ・モアハ
(32) 優先日	平成27年6月11日(2015.6.11)		ウス・ドライブ・5775
(33) 優先権主張国	米国 (US)		
早期審査対象出願			最終頁に続く

(54) 【発明の名称】 ブロックの区分を用いた高速ビデオ符号化方法

(57) 【特許請求の範囲】

【請求項1】

ビデオデータを符号化する方法であって、

前記ビデオデータのブロックの複雑さが閾値を超える場合における完全レートひずみ(RD)確認方式または前記ブロックの前記複雑さが前記閾値を超えない場合における高速RD確認方式のうちの1つを選択するステップであって、前記ブロックの前記複雑さが、前記ブロックのサブ区分間でのひずみのばらつきに基づき、前記高速RD確認方式が、前記高速RD確認方式の早期終了を含む、ステップと、

前記選択された完全RD確認方式または高速RD確認方式に基づいて、前記ビデオデータの前記ブロックを符号化することと関連付けられるRDコストを決定するステップと、

前記決定されたRDコストに基づいて、前記ブロックの区分方式を決定するステップと、
前記決定されたRDコストに基づいて、前記決定された区分方式を使用して前記ブロックを符号化するステップとを備える、方法。

【請求項2】

前記高速RD確認方式を選択したことに応答して、前記ブロックに対する最大の区分サイズと関連付けられるひずみ値を決定するステップと、

前記ブロックに対する前記最大の区分サイズと関連付けられる前記ひずみ値が閾値のひずみ値を超えると判定したことに応答して、前記ブロックをサブ区分へ区分するかどうかを判定するステップとさらに備える、請求項1に記載の方法。

【請求項3】

10

20

前記ブロックをサブ区分へ区分するかどうかを判定するステップがさらに、
前記サブ区分の各々と関連付けられるひずみ値を第2のひずみ閾値と比較するステップと、

前記サブ区分の少なくとも1つが前記第2のひずみ閾値より大きなひずみ値を有すると判定したことに応答して、前記ブロックをサブ区分へ区分するステップとをさらに備える、請求項2に記載の方法。

【請求項4】

前記第2のひずみ閾値が、前記サブ区分の各々と関連付けられる前記ひずみ値のスケーリングされた平均に基づく、請求項3に記載の方法。

【請求項5】

前記第2のひずみ閾値が、前記サブ区分の各々と関連付けられる前記ひずみ値のばらつきに基づく、請求項3に記載の方法。

【請求項6】

前記ブロックをサブ区分へ区分するかどうかを判定するステップがさらに、
前記サブ区分のうちの1つが、少なくとも1つの他の前記サブ区分と比較して異なる動き情報と関連付けられるかどうかを判定するステップと、

前記サブ区分のうちの前記1つが、前記少なくとも1つの他の前記サブ区分と比較して異なる動き情報と関連付けられると判定したことに応答して、前記サブ区分のうちの前記1つを追加のサブ区分へさらに区分するステップとを備える、請求項3に記載の方法。

【請求項7】

前記完全RD確認方式を選択するステップが、ビデオデータの前記ブロックを含む現在のスライスがiスライスである場合、前記完全RD確認方式を選択するステップを備える、請求項1に記載の方法。

【請求項8】

前記完全RD確認方式または前記高速RD確認方式を選択するステップが、ビデオエンコーダの高速符号化モードが有効にされているかどうかに基づく、請求項1に記載の方法。

【請求項9】

前記完全RD確認方式を選択するステップが、前記ブロックがSKIPモードを使用してコーディングされるときに前記完全RD確認方式を選択するステップを備える、請求項1に記載の方法。

【請求項10】

ビデオデータを記憶するように構成されるメモリと、
少なくとも1つのプロセッサとを備え、前記少なくとも1つのプロセッサが、
前記ビデオデータのブロックの複雑さが閾値を超える場合における完全レートひずみ(RD)確認方式または前記ブロックの前記複雑さが前記閾値を超えない場合における高速RD確認方式のうちの1つを選択することであって、前記ブロックの前記複雑さが、前記ブロックのサブ区分間でのひずみのばらつきに基づき、前記高速RD確認方式が、前記高速RD確認方式の早期終了を含む、選択することをし、

前記選択された完全RD確認方式または高速RD確認方式に基づいて、前記ビデオデータの前記ブロックを符号化することと関連付けられるRDコストを決定し、

前記決定されたRDコストに基づいて、前記ブロックの区分方式を決定し、

前記決定されたRDコストに基づいて、前記決定された区分方式を使用して前記ブロックを符号化するように構成される、ビデオ符号化デバイス。

【請求項11】

前記少なくとも1つのプロセッサが、
前記高速RD確認方式を選択したことに応答して、前記ブロックに対する最大の区分サイズと関連付けられるひずみ値を決定し、

前記ブロックに対する前記最大の区分サイズと関連付けられる前記ひずみ値が閾値のひずみ値を超えると判定したことに応答して、前記ブロックをサブ区分へ区分するかどうかを判定するようにさらに構成される、請求項10に記載のビデオ符号化デバイス。

10

20

30

40

50

【請求項12】

前記ブロックをサブ区分へ区分するかどうかを判定するために、前記少なくとも1つのプロセッサがさらに、

前記サブ区分の各々と関連付けられるひずみ値を第2のひずみ閾値と比較し、

前記サブ区分の少なくとも1つが前記第2のひずみ閾値より大きなひずみ値を有すると判定したことに応答して、前記ブロックをサブ区分へ区分するように構成される、請求項11に記載のデバイス。

【請求項13】

前記ブロックをサブ区分へ区分するかどうかを判定するために、前記少なくとも1つのプロセッサがさらに、

前記サブ区分のうちの1つが、少なくとも1つの他の前記サブ区分と比較して異なる動き情報と関連付けられるかどうかを判定し、

前記サブ区分のうちの前記1つが、前記少なくとも1つの他の前記サブ区分と比較して異なる動き情報と関連付けられると判定したことに応答して、前記サブ区分のうちの前記1つを追加のサブ区分へさらに区分するように構成される、請求項11に記載のデバイス。

【請求項14】

前記完全RD確認方式を選択するために、前記少なくとも1つのプロセッサがさらに、ビデオデータの前記ブロックを含む現在のスライスがiスライスである場合、前記完全RD確認方式を選択するように構成される、請求項10に記載のデバイス。

【請求項15】

前記完全RD確認方式を選択するために、前記少なくとも1つのプロセッサが、前記ブロックがSKIPモードを使用してコーディングされるときに前記完全RD確認方式を選択するように構成される、請求項10に記載のデバイス。

【請求項16】

実行されると、ビデオ符号化デバイスの少なくとも1つのプロセッサに、

ビデオデータのブロックの複雑さが閾値を超える場合における完全レートひずみ(RD)確認方式または前記ブロックの前記複雑さが前記閾値を超えない場合における高速RD確認方式のうちの1つを選択することであって、前記ブロックの前記複雑さが、前記ブロックのサブ区分間でのひずみのばらつきに基づき、前記高速RD確認方式が、前記高速RD確認方式の早期終了を含む、選択することをし、

前記選択された完全RD確認方式または高速RD確認方式に基づいて、前記ビデオデータの
前記ブロックを符号化することと関連付けられるRDコストを決定し、

前記決定されたRDコストに基づいて、前記ブロックの区分方式を決定し、

前記決定されたRDコストに基づいて、前記決定された区分方式を使用して前記ブロックを符号化するようにさせる命令を記憶した、コンピュータ可読記憶媒体。

【請求項17】

実行されると、前記少なくとも1つのプロセッサに、

前記高速RD確認方式を選択したことに応答して、前記ブロックに対する最大の区分サイズと関連付けられるひずみ値を決定し、

前記ブロックに対する前記最大の区分サイズと関連付けられる前記ひずみ値が閾値のひずみ値を超えると判定したことに応答して、前記ブロックをサブ区分へ区分するかどうかを判定するようにさせる命令を記憶した、請求項16に記載のコンピュータ可読記憶媒体。

【請求項18】

前記ブロックをサブ区分へ区分するかどうかを判定するようにさせる命令は、実行されると、前記少なくとも1つのプロセッサに、

前記サブ区分の各々と関連付けられるひずみ値を第2のひずみ閾値と比較し、

前記サブ区分の少なくとも1つが前記第2のひずみ閾値より大きなひずみ値を有すると判定したことに応答して、前記ブロックをサブ区分へ区分するようにさせる、請求項17に記載のコンピュータ可読記憶媒体。

【請求項19】

10

20

30

40

50

前記ブロックをサブ区分へ区分するかどうかを判定するようにさせる命令は、実行されると、前記少なくとも1つのプロセッサに、

前記サブ区分のうちの1つが、少なくとも1つの他の前記サブ区分と比較して異なる動き情報と関連付けられるかどうかを判定し、

前記サブ区分のうちの前記1つが、前記少なくとも1つの他の前記サブ区分と比較して異なる動き情報と関連付けられると判定したことに応答して、前記サブ区分のうちの前記1つを追加のサブ区分へさらに区分するようにさせる、請求項17に記載のコンピュータ可読記憶媒体。

【請求項20】

前記完全RD確認方式を選択するようにさせる命令は、実行されると、前記少なくとも1つのプロセッサに、ビデオデータの前記ブロックを含む現在のスライスがiスライスである場合、前記完全RD確認方式を選択するようにさせる、請求項16に記載のコンピュータ可読記憶媒体。

【請求項21】

前記完全RD確認方式を選択するようにさせる命令は、実行されると、前記少なくとも1つのプロセッサに、前記ブロックがSKIPモードを使用してコーディングされるときに前記完全RD確認方式を選択するようにさせる、請求項16に記載のコンピュータ可読記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本出願は、その内容全体が参照により本明細書に組み込まれる、2015年3月6日に出願された米国特許出願第62/129,545号の優先権を主張する。

【0002】

本開示は、ビデオ符号化に関する。

【背景技術】

【0003】

デジタルビデオ機能は、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、広範囲のデバイスに組み込まれ得る。デジタルビデオデバイスは、MPEG-2、MPEG-4、ITU-T H.263、ITU-T H.264/MPEG-4、Part 10、アドバンストビデオコーディング(AVC)、高効率ビデオコーディング(HEVC)規格によって定義された規格、ならびに、スケーラブルビデオコーディング(SVC)マルチビュービデオコーディング(MVC)、スケーラブルHEVC(SHVC)、マルチビューHEVC(MV-HEVC)、3D-HEVC、およびHEVC範囲拡張(Range Extensions)などの、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のビデオコーディング技法を実装する。HEVC規格は、「ITU-T Recommendation H.265」とも呼ばれる。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し、受信し、符号化し、復号し、かつ/または記憶することができる。

【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に、空間(ピクチャ内)予測および/または時間(ピクチャ間)予測を含む。ブロックベースのビデオコーディングのために、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)がビデオブロックに区分されることがあり、ビデオブロックは、ツリーブロック、符号化ツリーユニット(CTU)、符号化ユニット(CU)、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。ビデオブロックは、ルーマブロックとクロマブロックとを含み得る。ピクチャのイントラコーディングされた(I)スライスの中のビデオブロックは、同じピクチャ中の隣接ブロック中の参照サンプルに対する空間予測を使用し

10

20

30

40

50

て符号化される。ピクチャのインターコーディングされた(PまたはB)スライスの中のビデオブロックは、同じピクチャにおける隣接するブロックの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の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。

【0005】

空間予測または時間予測が、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もたらす。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。インターコーディングされたブロックは、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に従って符号化され、残差データは、コーディングされたブロックと予測ブロックとの差分を示す。イントラコーディングされたブロックは、イントラコーディングモードおよび残差データ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、画素領域から変換領域に変換され、残差変換係数をもたらすことがあり、その残差変換係数は、次いで量子化され得る。なお一層の圧縮が達成するために、量子化された変換係数がエントロピーコーディングされ得る。

10

【0006】

1つまたは複数の例の詳細が、添付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、説明および図面から、ならびに特許請求の範囲から明らかになる。

【発明の概要】**【課題を解決するための手段】**

20

【0007】

本開示の技法は全般に、ビデオデータを符号化することに関する。ビデオエンコーダは、ビデオ符号化の間にピクチャ(フレームとも呼ばれる)を符号化する。ビデオエンコーダは、ビデオデータのブロックへとピクチャを区分する。ビデオエンコーダは、ブロックの区分のサイズ、ならびに任意の区分がさらに区分されるべき回数を決定することができる。ブロックが区分される方法は、区分方式と呼ばれる。ビデオエンコーダは一般に、選択された区分方式が最低のレートひずみ(RD)コストを生み出すように、各区分方式と関連付けられるRDコストに基づいてブロックを区分することができる。本開示の技法は全般に、ビデオエンコーダが、ビデオデータのブロックの区分方式を決定するために高速RD確認を使用するか完全RD確認を使用するかを決定することを、可能にすることに関する。

30

【0008】

ビデオエンコーダはまた、ビデオデータのブロックの区分モードを選択する。たとえば、ブロックはイントラ予測またはインター予測され得る。一般に、ビデオエンコーダは、ブロックの候補予測モードを確認することを決定するために、モード確認順序を使用する。ビデオエンコーダは、候補予測モードと関連付けられるRDコストに基づいて、ブロックを予測するためにどのビデオコーディングモードを使用するかを決定することができる。本開示の技法は、ビデオエンコーダが、適応的なモード確認順序を使用して、ビデオデータのブロックの予測モードを選択することを可能にし得る。

【0009】

本開示の一例では、ビデオデータを符号化する方法は、完全レートひずみ(RD)確認方式または高速RD確認方式のうちの1つを選択するステップと、選択された完全RD確認方式または高速RD確認方式に基づいてビデオデータのブロックを符号化することと関連付けられるRDコストを決定するステップと、決定されたRDコストに基づいてブロックの区分方式を決定するステップと、決定されたRDコストに基づいて決定された区分方式を使用してブロックを符号化するステップとを備える。

40

【0010】

別の例では、ビデオ符号化デバイスは、ビデオデータを記憶するように構成されるメモリと、少なくとも1つのプロセッサとを備え、少なくとも1つのプロセッサが、完全レートひずみ(RD)確認方式または高速RD確認方式のうちの1つを選択し、選択された完全RD確認方式または高速RD確認方式に基づいてビデオデータのブロックを符号化することと関連付

50

けられるRDコストを決定し、決定されたRDコストに基づいてブロックの区分方式を決定し、決定されたRDコストに基づいて決定された区分方式を使用してブロックを符号化するように構成される。

【0011】

別の例では、ビデオを符号化するための手段を備える装置はさらに、選択された完全RD確認方式または高速RD確認方式に基づいてビデオデータのブロックを符号化することと関連付けられるRDコストを決定するための手段と、決定されたRDコストに基づいてブロックの区分方式を決定するための手段と、決定されたRDコストに基づいて決定された区分方式を使用してブロックを符号化するための手段とを備える。

【0012】

別の例では、非一時的コンピュータ可読記憶媒体は、実行されると、少なくとも1つのプロセッサに、完全レートひずみ(RD)確認方式または高速RD確認方式のうちの1つを選択させ、選択された完全RD確認方式または高速RD確認方式に基づいてビデオデータのブロックを符号化することと関連付けられるRDコストを決定させ、決定されたRDコストに基づいてブロックの区分方式を決定させ、決定されたRDコストに基づいて決定された区分方式を使用してブロックを符号化させる、記憶された命令を含んでいる。

【図面の簡単な説明】

【0013】

【図1】本開示の1つまたは複数の態様による、適応的な色成分間残差予測のための技法を実行することができる例示的なビデオ符号化および復号システムを示すブロック図である。

【図2】リーフCUへと区分されるCUを示す概念図である。

【図3A】様々な区分方式に従って区分される例示的なCUを示す概念図である。

【図3B】様々な区分方式に従って区分される例示的なCUを示す概念図である。

【図4】予測ユニット(PU)の例示的なサイズおよび分割を示す概念図である。

【図5】様々な例示的なTU区分方式を示す概念図である。

【図6】HEVC規格に従った、ビデオエンコーダがレートひずみ(RD)コスト比較に基づいてコーディングモードを選択することができる例示的なプロセスを示すフローチャートである。

【図7A】HEVC規格の様々な態様による、例示的なモード選択技法を示すフローチャートである。

【図7B】HEVC規格の様々な態様による、例示的なモード選択技法を示すフローチャートである。

【図8】本開示の1つまたは複数の態様による技法を実行することができるビデオエンコーダの例を示すブロック図である。

【図9】本開示の1つまたは複数の態様による、ビデオエンコーダがモード確認順序を適応的に選択するために実行することができる例示的なプロセスを示すフローチャートである。

【図10】本開示の1つまたは複数の態様による、ビデオエンコーダがCUサイズを確認するための2つの可能な順序の中から選択するために実行することができる例示的なプロセスを示すフローチャートである。

【図11】本開示の技法による、ビデオエンコーダが予測モード選択の適応的な順序付けのために実行することができる別の例示的なプロセスを示すフローチャートである。

【図12】本開示に態様による、時間レイヤのサポートのための例示的な予測構造を示す概念図である。

【図13A】本開示の1つまたは複数の態様による、ビデオエンコーダがPU区分方式を適応的に決定するために使用することができる、対応する8×8のCUの誤差分布を示す概念図である。

【図13B】本開示の1つまたは複数の態様による、ビデオエンコーダがPU区分方式を適応的に決定するために使用することができる、対応する8×8のCUの誤差分布を示す概念図

10

20

30

40

50

である。

【図14】本開示の態様による、ビデオエンコーダがTU区分方式を適応的に決定するために実施することができる例示的なプロセスを示すフローチャートである。

【図15A】本開示の1つまたは複数の態様による、ビデオエンコーダがTU区分方式を適応的に決定するために使用することができる、対応する8×8のCUの誤差分布を示す概念図である。

【図15B】本開示の1つまたは複数の態様による、ビデオエンコーダがTU区分方式を適応的に決定するために使用することができる、対応する8×8のCUの誤差分布を示す概念図である。

【図16】本開示の態様による、ビデオエンコーダがPU区分方式を適応的に決定するために実施することができる例示的なプロセスを示すフローチャートである。

【図17A】本開示による、ビデオエンコーダがブロック区分の決定を用いた高速符号化の決定を実行するために実施することができる例示的なプロセスを示すフローチャートを集合的に示す流れ図である。

【図17B】本開示による、ビデオエンコーダがブロック区分の決定を用いた高速符号化の決定を実行するために実施することができる例示的なプロセスを示すフローチャートを集合的に示す流れ図である。

【図17C】本開示による、ビデオエンコーダがブロック区分の決定を用いた高速符号化の決定を実行するために実施することができる例示的なプロセスを示すフローチャートを集合的に示す流れ図である。

【図18】符号化利得に関して、ブロック区分の決定を用いた高速符号化の実行の例を示すグラフを示す概念図である。

【図19】制御パラメータに従って符号化される画像と本開示の高速符号化および区分の決定に従って符号化される画像とのピクチャ品質の比較を示す概念図である。

【図20】本開示の1つまたは複数の態様による技法を実行することができるビデオデコーダの例を示すブロック図である。

【図21】本開示の技法による、モード確認順序を適応的に決定するためのプロセスを示すフローチャートである。

【図22】本開示の技法による、レートひずみコストに基づいてブロックの区分方式を決定するためのプロセスを示すフローチャートである。

【発明を実施するための形態】

【0014】

全般に、本開示の技法は、ビデオデータのブロックのビデオコーディングモードおよび区分サイズに関する決断をビデオエンコーダが行うことを可能にするための技法に関する。本明細書において説明される様々な例は、ビデオエンコーダがブロックに対して決定した統計に基づいて、ビデオエンコーダがブロックの予測モードをより迅速に決定することを可能にし得る。本明細書において説明される例はまた、ビデオエンコーダがビデオデータのブロックの区分方式をより迅速に決定することを可能にし得る。区分の決定は、候補予測モードと関連付けられるレートひずみなどの統計に基づき得る。

【0015】

図1は、本開示の技法を利用し得る例示的なビデオコーディングシステム10を示すブロック図である。本明細書において使用される「ビデオエンコーダ」という用語は、総称的に、ビデオエンコーダとビデオデコーダの両方を指す。本開示では、「ビデオコーディング」または「コーディング」という用語は、ビデオ符号化またはビデオ復号を総称的に指すことがある。図1に示されるように、ビデオコーディングシステム10は、ソースデバイス12および宛先デバイス14を含む。ソースデバイス12は、符号化されたビデオデータを生成する。したがって、ソースデバイス12は、ビデオ符号化デバイスまたはビデオ符号化装置と呼ばれ得る。宛先デバイス14は、ソースデバイス12によって生成された符号化されたビデオデータを復号し得る。したがって、宛先デバイス14は、ビデオ復号デバイスまたはビデオ復号装置と呼ばれ得る。ソースデバイス12および宛先デバイス14は、ビデオコーディン

10

20

30

40

50

グデバイスまたはビデオコーディング装置の例であり得る。

【0016】

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、モバイルコンピューティングデバイス、ノートブック(たとえば、ラップトップ)コンピュータ、タブレットコンピュータ、セットトップボックス、いわゆる「スマート」フォンなどの電話ハンドセット、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、車載コンピュータなどを含む、広範囲のデバイスを備え得る。

【0017】

宛先デバイス14は、符号化されたビデオデータをソースデバイス12からチャンネル16を介して受信し得る。チャンネル16は、符号化されたビデオデータをソースデバイス12から宛先デバイス14に移動することが可能な1つまたは複数の媒体またはデバイスを備え得る。一例では、チャンネル16は、ソースデバイス12が符号化されたビデオデータをリアルタイムで宛先デバイス14に直接送信することを可能にする、1つまたは複数の通信媒体を備え得る。この例では、ソースデバイス12は、ワイヤレス通信プロトコルなどの通信規格に従って符号化されたビデオデータを変調することができ、変調されたビデオデータを宛先デバイス14へ送信することができる。1つまたは複数の通信媒体は、高周波(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の、ワイヤレスおよび/または有線の通信媒体を含み得る。1つまたは複数の通信媒体は、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、またはグローバルネットワーク(たとえば、インターネット)などの、パケットベースネットワークの一部を形成し得る。1つまたは複数の通信媒体は、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への通信を容易にする他の機器を含み得る。

【0018】

別の例では、チャンネル16は、ソースデバイス12によって生成された符号化されたビデオデータを記憶する記憶媒体を含み得る。この例では、宛先デバイス14は、たとえば、ディスクアクセスまたはカードアクセスを介して記憶媒体にアクセスし得る。記憶媒体は、Blu-ray(登録商標)ディスク、DVD、CD-ROM、フラッシュメモリ、または符号化されたビデオデータを記憶するための他の適切なデジタル記憶媒体などの、ローカルにアクセスされる様々なデータ記憶媒体を含み得る。

【0019】

さらなる例では、チャンネル16は、ソースデバイス12によって生成された符号化されたビデオデータを記憶するファイルサーバまたは別の中間記憶デバイスを含み得る。この例では、宛先デバイス14は、ファイルサーバまたは他の中間記憶デバイスにおいて記憶された符号化されたビデオデータに、ストリーミングまたはダウンロードを介してアクセスし得る。ファイルサーバは、符号化されたビデオデータを記憶するとともに符号化されたビデオデータを宛先デバイス14へ送信することが可能なタイプのサーバであり得る。例示的なファイルサーバは、ウェブサーバ(たとえば、ウェブサイト用の)、ファイル転送プロトコル(FTP)サーバ、ネットワーク接続ストレージ(NAS)デバイス、およびローカルディスクドライブを含む。

【0020】

宛先デバイス14は、インターネット接続などの標準的なデータ接続を通じて符号化されたビデオデータにアクセスし得る。例示的なタイプのデータ接続は、ファイルサーバに記憶されている符号化されたビデオデータにアクセスするために適切である、ワイヤレスチャンネル(たとえば、Wi-Fi接続)、有線接続(たとえば、DSL、ケーブルモデムなど)、または両方の組合せを含み得る。ファイルサーバからの符号化されたビデオデータの送信は、ストリーミング送信、ダウンロード送信、または両方の組合せであってよい。

【0021】

本開示の技法は、ワイヤレスの用途または設定に限定されない。技法は、オーバーエ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、たとえば、インターネットを介したストリーミングビデオ送信、データ記憶媒体に記憶するためのビデオ

10

20

30

40

50

データの符号化、データ記憶媒体に記憶されたビデオデータの復号、または他の用途などの様々なマルチメディア用途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、ビデオコーディングシステム10は、ビデオストリーミング、ビデオ再生、ビデオブロードキャスト、および/またはビデオ電話のような用途をサポートするために、単方向または双方向のビデオ送信をサポートするように構成され得る。

【0022】

図1に示されるビデオコーディングシステム10は例にすぎず、本開示の技法は、符号化デバイスと復号デバイスとの間のデータ通信を必ずしも含まないビデオコーディングの設定(たとえば、ビデオ符号化またはビデオ復号)に適用され得る。他の例では、データは、ローカルメモリからの取出し、ネットワークを介したストリーミングなどが行われる。ビデオ符号化デバイスはデータを符号化してメモリに記憶することができ、かつ/または、ビデオ復号デバイスはメモリからデータを取り出して復号することができる。多くの例では、互いに通信しないが、単にデータをメモリへと符号化し、かつ/またはメモリからデータを取り出して復号するデバイスによって、符号化および復号が実行される。ビデオエンコーダ20およびビデオデコーダ30は、ビデオデータを記憶するように構成されるメモリを備え得る。ビデオエンコーダ20は、メモリに記憶されたビデオデータを符号化し得る。ビデオデコーダ30は、符号化されたビデオデータを復号し、得られたビデオデータをメモリに記憶し得る。

【0023】

図1の例では、ソースデバイス12は、ビデオソース18、ビデオエンコーダ20、および出力インターフェース22を含む。いくつかの例では、出力インターフェース22は、変調器/復調器(モデム)および/または送信機を含み得る。ビデオソース18は、ビデオキャプチャデバイス、たとえば、ビデオカメラ、以前キャプチャされたビデオデータを含むビデオアーカイブ、ビデオデータをビデオコンテンツプロバイダから受信するためのビデオフィードインターフェース、および/もしくはビデオデータを生成するためのコンピュータグラフィックスシステム、またはビデオデータのそのようなソースの組合せを含み得る。

【0024】

ビデオエンコーダ20は、ビデオソース18からのビデオデータを符号化し得る。いくつかの例では、ソースデバイス12は、符号化されたビデオデータを宛先デバイス14へ出力インターフェース22を介して直接送信する。他の例では、復号および/または再生のために宛先デバイス14が後でアクセスするために、符号化されたビデオデータはまた、記憶媒体またはファイルサーバへ記憶され得る。

【0025】

図1の例では、宛先デバイス14は、入力インターフェース28、ビデオデコーダ30、およびディスプレイデバイス32を含む。いくつかの例では、入力インターフェース28は、受信機および/またはモデムを含む。入力インターフェース28は、符号化されたビデオデータをチャンネル16を通じて受信し得る。ディスプレイデバイス32は、宛先デバイス14と一体であることがあり、または宛先デバイス14の外部にあることがある。一般に、ディスプレイデバイス32は復号されたビデオデータを表示する。ディスプレイデバイス32は、液晶ディスプレイ(LCD)、プラズマディスプレイ、有機発光ダイオード(OLED)ディスプレイ、または別のタイプのディスプレイデバイスなどの様々なディスプレイデバイスを備え得る。

【0026】

ビデオエンコーダ20およびビデオデコーダ30は各々、1つまたは複数のマイクロプロセッサ、デジタル信号プロセッサ(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回路、ハードウェア、またはそれらの任意の組合せなどの、様々な適切な回路のいずれかとして実装され得る。技法が部分的にソフトウェアで実装される場合、本開示の技法を実行するために、デバイスは、ソフトウェアのための命令を適切な非一時的コンピュータ可読記憶媒体に記憶することができ、1つまたは複数のプロセッサを使用するハードウェアにおいて命令を実行することができる。前述のもの(ハードウェア、ソフトウェア、ハードウェアとソフトウェアの組合せなど

10

20

30

40

50

を含む)のいずれもが、1つまたは複数のプロセッサであると見なされ得る。ビデオエンコーダ20およびビデオデコーダ30の各々は、1つまたは複数のエンコーダまたはデコーダに含まれてよく、それらのいずれもが、それぞれのデバイスの中で複合エンコーダ/デコーダ(コーデック)の一部として一体化されてよい。

【0027】

本開示は全般に、ある情報をビデオデコーダ30などの別のデバイスへ「シグナリング」または「送信」するビデオエンコーダ20に言及し得る。「シグナリングすること」または「送信すること」という用語は、全般に、シンタックス要素、および/または圧縮されたビデオデータを復号するために使用される他のデータの通信を指し得る。そのような通信は、リアルタイムで、またはほぼリアルタイムで発生し得る。代替的に、そのような通信は、ある時間の範囲にわたって、たとえば、符号化の時点において符号化されたビットストリームの中のシンタックス要素をコンピュータ可読記憶媒体に記憶するときが発生することがあり、次いで、シンタックス要素が、この媒体に記憶された後の任意の時点において復号デバイスによって取り出され得る。

10

【0028】

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、上述のHEVC規格などの、文書ITU-T H.265、SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS、Infrastructure of Audiovisual Services-Coding of Moving Video、「High Efficiency Video Coding」、2014年10月に記載されている、HEVCバージョン1において記載されているビデオ圧縮規格に従って動作する。HEVCバージョン1に加えて、HEVCのためのスケーラブルビデオコーディング拡張、マルチビュービデオコーディング拡張、範囲拡張、スクリーンコンテンツコーディング(SCC)拡張、および3Dコーディング拡張を制作するための取り組みが進行中である。

20

【0029】

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通常、一連のピクチャを含む。ピクチャは、「フレーム」と呼ばれることもある。ピクチャは、 S_L 、 S_{Cb} 、および S_{Cr} と表記される3成分のサンプルアレイを含み得る。 S_L 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_{Cb} は、Cbクロミナンスサンプルの2次元アレイである。 S_{Cr} 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「クロマ」サンプルと呼ばれることもある。この色空間は、「YUV」または「YCbCr」と呼ばれることがある。他の事例では、ピクチャはモノクロであることがあり、ルーマサンプルのアレイだけを含むことがある。HEVC範囲拡張によれば、ピクチャは、YcCo、YcCo-R、およびこれらの組合せなどの、色空間とクロマサブサンプリングとを備え得る。

30

【0030】

ピクチャの符号化された表現を生成するために、ビデオエンコーダ20は、符号化ツリーユニット(CTU)のセットを生成し得る。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、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、および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であり得る。コーディングツリーブロックは、サンプルの $N \times N$ ブロックであり得る。CTUは、「ツリーブロック」または「最大符号化ユニット」(LCU)と呼ばれることもある。HEVCのCTUは、H.264/AVCなどの他の規格のマクロブロックと概ね類似していることがある。しかしながら、CTUは、必ずしも特定のサイズに限定されず、1つまたは複数の符号化ユニット(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。

40

【0031】

コーディングされたCTUを生成するために、ビデオエンコーダ20がCTUのコーディングツリーブロック上で4分木区分を再帰的に実行して、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ことができ、したがって、「符号化ツリーユニット」という名前である。コーディングブロックは、サンプルの $N \times N$ ブロックである。CUは、ルーマサン

50

ブルアレイ、Cbサンプルアレイ、およびCrサンプルアレイ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およびクロマサンプルの2つの対応するコーディングブロック、ならびに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であり得る。モノクロピクチャ、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、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。

【0032】

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、長方形(すなわち、正方形または非正方形)の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、クロマサンプルの2つの対応する予測ブロック、および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であり得る。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ブロックを生成し得る。モノクロピクチャ、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、予測ブロック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。

10

【0033】

ビデオエンコーダ20は、PUの予測ブロックを生成するために、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の予測ブロックを生成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と関連付けられるピクチャの復号されたサンプルに基づいて、PUの予測ブロックを生成し得る。

20

【0034】

ビデオエンコーダ20がPUの予測ブロックを生成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と関連付けられるピクチャ以外の1つまたは複数のピクチャの復号されるサンプルに基づいて、PUの予測ブロックを生成し得る。ビデオエンコーダ20は、PUの予測ブロックを生成するために、単予測または双予測を使用し得る。ビデオエンコーダ20がPUの予測ブロックを生成するために単予測を使用するとき、PUは単一の動きベクトル(MV)を有し得る。ビデオエンコーダ20がPUの予測ブロックを生成するために双予測を使用するとき、PUは2つのMVを有し得る。

30

【0035】

ビデオエンコーダ20がCUの1つまたは複数のPUに対して予測ブロック(たとえば、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)を生成した後、ビデオエンコーダ20は、CUの残差ブロックを生成し得る。CUの残差ブロック中の各サンプルは、CUのPUの予測ブロック中のサンプルとCUの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。たとえば、ビデオエンコーダ20は、CUのルーマ残差ブロックを生成し得る。CUのルーマ残差ブロック中の各サンプルは、CUのPUの予測ルーマブロック中のルーマサンプルとCU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。加えて、ビデオエンコーダ20は、CUのCb残差ブロックを生成し得る。CUのCb残差ブロック中の各サンプルは、CUのPUの予測Cbブロック中のCbサンプルとCU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す。ビデオエンコーダ20はまた、CUのCr残差ブロックを生成し得る。CUのCr残差ブロック中の各サンプルは、CUのPUの予測Crブロック中のCrサンプルとCU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。

40

【0036】

さらに、ビデオエンコーダ20は、4分木区画を使用して、CUの残差ブロック(たとえば、ルーマ残差ブロック、Cb残差ブロック、およびCr残差ブロック)を1つまたは複数の変換ブロック(たとえば、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)に区分し得る。変換ブロックは、同じ変換が適用されるサンプルの、長方形の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、クロマサンプルの2つの対応する変換ブロック、および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であり得る。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

50

、およびCr変換ブロックと関連付けられ得る。TUと関連付けられ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。モノクロピクチャ、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、変換ブロックの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。

【0037】

ビデオエンコーダ20は、1つまたは複数の変換をTUの変換ブロックに適用して、TUの係数ブロックを生成し得る。たとえば、ビデオエンコーダ20は、1つまたは複数の変換をTUのルーマ変換ブロックに適用して、TUのルーマ係数ブロックを生成し得る。ビデオエンコーダ20は、1つまたは複数の変換をTUのCb変換ブロックに適用して、TUのCb係数ブロックを生成し得る。ビデオエンコーダ20は、1つまたは複数の変換をTUのCr変換ブロックに適用して、TUのCr係数ブロックを生成し得る。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。本開示は、全体で「ブロック」という用語を使用する。本明細書において定義されるように、ブロックは、CTU、CU、PU、TU、または成分のサンプルのグループ、たとえばルーマもしくはクロマサンプルのグループを指し得る。

【0038】

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)を生成した後、ビデオエンコーダ20は、係数ブロックを量子化し得る。量子化は一般に、変換係数が量子化されて、場合によっては、変換係数を表すために使用されるデータの量を低減し、さらなる圧縮をもたらすプロセスを指す。ビデオエンコーダ20が係数ブロックを量子化した後、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(CABAC)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されたシンタックス要素をビットストリームの中に出し得る。ビットストリームは、ビデオデータの符号化された表現を備え得る。

【0039】

ビデオエンコーダ20は、エントロピー符号化されたシンタックス要素を含むビットストリームを出し得る。ビットストリームは、コーディングされたピクチャの表現および関連するデータを形成するビットのシーケンスを含み得る。ビットストリームは、ネットワーク抽象化レイヤ(NAL)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)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規定される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む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個のビットを含む。

【0040】

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)のためのRBSPをカプセル化することがあり、第2のタイプのNALユニットはコーディングされたスライスのためのRBSPをカプセル化することがあり、第3のタイプのNALユニットは補足強化情報(SEI)のためのRBSPをカプセル化することがあり、以下同様である。(パラメータセットおよびSEIメッセージのためのRBSPではなく)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、ビデオ符号化レイヤ(VCL)NALユニットと呼ばれ得る。

【0041】

図1の例では、ビデオデコーダ30は、ビデオエンコーダ20によって生成されるビットストリームを受信する。加えて、ビデオデコーダ30は、ビットストリームを構文解析して、ビットストリームからシンタックス要素を取得し得る。ビデオデコーダ30は、ビットスト

10

20

30

40

50

リーフから取得されたシンタックス要素に少なくとも一部基づいて、ビデオデータのピクチャを再構築し得る。ビデオデータを再構築するためのプロセスは全般に、ビデオエンコーダ20によって実行されるプロセスの逆であり得る。たとえば、ビデオデコーダ30は、イントラ予測またはインター予測を使用して、現在のCUのPUの予測ブロックを決定し得る。加えて、ビデオデコーダ30は、現在のCUのTUの係数ブロックを逆量子化し得る。ビデオデコーダ30は、係数ブロックに対して逆変換を実行して、現在のCUのTUの変換ブロックを再構築し得る。ビデオデコーダ30は、現在のCUのPUの予測ブロックのサンプルを、現在のCUのTU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、現在のCUのコーディングブロックを再構築し得る。ピクチャのCUごとにコーディングブロックを再構築することによって、ビデオデコーダ30はピクチャを再構築し得る。

10

【 0 0 4 2 】

上で説明されたように、HEVC規格の様々な態様によれば、ビデオエンコーダ20は、ピクチャ(本明細書の様々な事例において「フレーム」とも呼ばれる)をCTUに分割し得る。特定のCTUは、CUからなる4分木データ構造のルートに相当する。そして、各CUは、ビデオエンコーダ20が様々なコーディングツールを適用することができる基本単位に相当する。ビデオエンコーダはさらに、CUをサブCUに分割(すなわち、区分)することができる。たとえばCUおよびサブCUへの、CTUの区分の様々な例が、ここでより詳細に説明される。

【 0 0 4 3 】

図2は、リーフCUへと区分されるCUを示す概念図である。図2の例では、例示的なCU100は $2N \times 2N$ の次元を有し、 N はサンプルの数である。図2に示されるように、ビデオエンコーダ20は、CU100を最大で4つのリーフCU102A~102Dに分割し得るので、リーフCU102A~102Dの各々が $N \times N$ の次元を有する。図2に示される対照的なCUの構造および分割(区分)に加えて、多くの他のCUの構造も、HEVC規格に従って可能である。

20

【 0 0 4 4 】

図3Aは、区分されたCTUを示す概念図である。図3AはCTU110を含む。図3Aの例では、例示的なCTU110は4つのCUに区分され、これらはCTU110内の最大の正方形として示されている。4つの最大のCUの左上および右下はさらに、4つのサブCUへと分割される。左上のCUの右下のサブCU内で、サブCUはさらに4つのサブCUへと区分され、これらのサブCUの右下は再び4つのリーフCUへと区分される。右下のCUの右下のサブCUは、同様の区分方式を使用して区分される。

30

【 0 0 4 5 】

図3Aの目的は、ビデオエンコーダ20が様々な次元および深度のリーフCUへとCUを分割できることを示すことである。言い換えると、図3Aに示されるように、ビデオエンコーダ20は、4分木構造の異なる段階において分割プロセスを中止することによって、CTU110をリーフCUへと分割することができる。本開示の様々な事例において、図3Aに示されるようなCTU110の構造は、サブCUまたはリーフCUへと「多重分割」されるCUを含むCTUとして言及される。別のCUの分割により(直接または間接的に)生じる各CUは、各CUの分割元のCUのサブCUとして言及される。サブCUへとさらに分割されないCUは、リーフCUと呼ばれる。

【 0 0 4 6 】

図3Bは、CTUの区分およびCTUの対応する4分木表現を示す概念図である。図3Bは、例示的なCTU111および対応する4分木構造112を示す。この例では、CTU111は 32×32 の次元を有し得る。図3Bにおいて、ビデオエンコーダ20は、 8×8 と 16×16 の間で次元が変化するサブCUへとCTU111を分割することを決定した。ビデオエンコーダ20およびビデオデコーダ30は、対応する4分木データ構造112を使用して、CTU111を表現し得る。4分木112において、4分木のルートノードは4つのノードに分割される。この例では、4つのノードの各々は、 16×16 のサブCUへの 32×32 のCTUの分割に対応する。CTUの右上のCUに対応する 16×16 ノードのうちの1つが、4つの 8×8 のサブCUへとさらに分割される。

40

【 0 0 4 7 】

CTU111に関して示される区分構造は、コーディング効率を改善し得る。たとえば、ビデオエンコーダ20は、複雑な領域に対してはより小さいブロックサイズ(たとえば、 8×8 の

50

サブCU)を選択し得るが、動きがほとんどもしくはまったくないエリア、または滑らかなテクスチャのエリアに対してはより大きなブロックサイズ(たとえば、16×16のサブCU)を選択し得る。HEVC参照ソフトウェアによれば、ビデオエンコーダ20は、レートひずみ(RD)最適化、すなわち要求されるコーディングビットの数およびビデオ画像コンテンツのひずみの量の最適化を実行するブロックサイズを決定し得る。一般に、帯域幅の効率性のためには多数のコーディングビットが望ましくないことがあるが、画像品質および視聴体験のためには大量のひずみが望ましくないことがある。RD最適化によれば、ビデオエンコーダ20は、可能な区分方式のRDコストを互いに比較することができ、最低の(最小の)可能なコストを伴う区分方式を選択することができる。

【0048】

HEVCに従ってRD最適化を実行することはコーディング効率に関して最適な区分の選択を実現できるが、すべての可能な区分を試験することはコーディング時間を大幅に増やすことがある。リアルタイム符号化のための現実的なシステムでは、すべての可能な区分事例(すなわち、次元および/または深度)を確認することは、リソースを大量に消費し時間がかかることがあり、いくつかの状況では可能でならないことがある。

【0049】

代替的に、ビデオエンコーダ20は、CTUを区分しないことを選ぶことができる。しかしながら、ビデオエンコーダ20がCTUを区分することをスキップし、大きなブロックサイズを使用してある特定の領域をコーディングすると、視覚的なアーティファクトが発生し得る。視覚的なアーティファクトは、大きなブロックの中の複数のエリアまたは複数の物体の間で異なる動きが示されるときに発生する可能性が特に高いことがある。一方、小さな区分を使用することは、特に大きなブロックのエリアが比較的均一な動きを示す事例において、すべての区分に対して動きおよび予測モードの情報をコーディングすることによって、コーディング効率を低下させ得る。したがって、様々な状況において、可能な区分方式の、特に主観的な品質の劣化の少ない区分方式のRDコストを完全に(たとえば、網羅的に)確認することなくブロックを区分することが、有益であり得る。

【0050】

図4は、予測ユニット(PU)の例示的なサイズおよび分割を示す概念図である。CUの分割により生じるサブCUに加えて、CUはさらにPUを含み得る。上で説明されたように、PUは予測情報を示し得る。各PUは、1つまたは複数のブロックへと分割(区分)され得る。PUの例示的なサイズおよび分割が、PUの区分114A~114Dの例として図4に示されている。

【0051】

図4の例では、PU114Aはサイズ2N×2Nの単一のブロックを備え、PU114Bはサイズ2N×Nの2つのブロックへと水平方向に区分され、PU114CはサイズN×2Nの2つのブロックへと垂直方向に区分され、PU114DはサイズN×Nの4つのブロックへと区分される。ビデオエンコーダ20は、予測モードを決定し、決定された予測モードを1つまたは複数のPUに適用することができる。HEVC規格に従った予測モードの例には、イントラ(Intra)モード、スキップ(Skip)モード、マージダイレクト(Merge Direct)モード、または高度動きベクトル予測(AMVP:Advanced Motion Vector Prediction)モードがある。

【0052】

ビデオエンコーダ20は、特定の予測モードの使用のために、いくつかのパラメータを選択し、導出し、または別様に決定することができる。たとえば、ビデオエンコーダ20は、イントラモードコーディングされたPUの予測方向を決定することができ、またはスキップモード、マージモード、およびダイレクトモードでコーディングされたPUに対して統合候補を決定することができ、または、AMVPモードでコーディングされたPUに対して基準フレームおよび/もしくは動きベクトルを決定することができる、などである。予測の後で、ビデオエンコーダ20は残差係数を決定することができる。ビデオエンコーダ20は、CUの残差係数をTUへと区分し得るので、ビデオエンコーダ20は、変換および/または量子化を独立に各TUへ適用することができる。

【0053】

図5は、様々な例示的なTU区分方式を示す概念図である。符号化ユニットは変換木を含み得る。図5に示される変換木116A～116Cは、そのような変換木の例である。変換木は、1つまたは複数のTUを備え得る。TUの各々は、離散コサイン変換または別の変換から生じる残差変換係数データ、ならびにデルタ量子化パラメータ(QP)データを含み得る。図5の例では、ビデオエンコーダ20が各TUのサイズを決定した。変換木116Aにおいて、TUサイズは $2N \times 2N$ のサンプルであり得る。変換木116Bは、サイズ $N \times N$ のサンプルの4つのTUを備えることがあり、変換木116Cは、サイズ $N/2 \times N/2$ のサンプルの16個のTUを備えることがある。

【0054】

ビデオエンコーダ20は、CTUに対する最適な区分方式、コーディングモード、および/またはコーディングパラメータを選択し、もしくは別様に決定するために、既存の技法を実施し得る。たとえば、ビデオエンコーダ20は、1つまたは複数の計算上の複雑さの制約のもとでコーディング効率を改善する(または場合によっては最大化する)区分方式およびコーディングモードを決定することができる。区分方式を決定するために、ビデオエンコーダ20は、本明細書において「m1」および「m2」と呼ばれる2つの符号化モードの間で一連の比較を実行することができる。モードm1およびm2は、たとえば、2つの異なるCUサイズ、2つの異なるPU区分方式、2つの異なるTU区分方式、PUに対する2つの異なる予測モード、ある予測モードに対する2つの異なる設定、またはこれらの任意の組合せを、CTUが有することに基き得る。

【0055】

図6は、モード選択プロセスの一部としての、網羅的なモード確認プロセスを示す流れ図である。一般に、ビデオエンコーダ20は、2つのモードのそれぞれの設定と関連付けられるレートひずみ(RD)コストを決定し、または見積もることができる。2つのモードは本明細書では「m1」および「m2」と呼ばれる。モードm1およびm2のRDコストは、「RDCost(m1)」および「RDCost(m2)」と本明細書では表記される。図6に示されるように、ビデオエンコーダ20は、コストRDCost(m1)を決定し(122)、RDCost(m2)を決定する(124)。ビデオエンコーダ20は次いで、関連するコストRDCost(m1)とRDCost(m2)を比較する(126)。

【0056】

ビデオエンコーダ20は、より小さいRDコストと関連付けられる符号化モードを選択することができる。RDCost(M1)がRDCost(M2)より大きくない場合、ビデオエンコーダ20はモードM1を選ぶ(128)。それ以外の場合、ビデオエンコーダ20はモードM2を選ぶ(130)。本明細書において説明されるように、プロセス120は網羅的なモード確認プロセスを代表する。プロセス120などの網羅的なモード確認技法によれば、ビデオエンコーダ20は、モードm1およびm2を無条件で、または言い換えると、すでに利用可能な符号化情報、統計、またはヒューリスティックデータを考慮することなく確認する。

【0057】

図7Aおよび図7Bは、本開示の技法による、例示的なモード確認プロセス140および160をそれぞれ示す概念図である。プロセス140および160は、HEVC規格の様々な態様による、モード選択技法を代表する。ビデオエンコーダ20は、モード確認および選択に関する「早期終了」を実施するために、プロセス140と160のいずれかを実行し得る。たとえば、ビデオエンコーダ20は、特定の条件が満たされていることに基づいて、プロセス140または160を実行して、実現性について特定のモードを確認するのを条件的にスキップすることができる。プロセス160の例では、ビデオエンコーダ20はまず、モードm1と関連付けられるRDコストを確認する(142)。

【0058】

モードm1の確認された特性(この場合は関連するRDコスト)がある条件(本明細書では「C1」と呼ばれる)を満たす、または満足する場合(ブロック144の「Yes」の分岐)、ビデオエンコーダ20は、関連するブロックの符号化のためにモードm1を選ぶ(145)。したがって、現在のブロックに対するモードm1と関連付けられるRDコストが条件C1を満たすとビデオエンコーダ20が判定する場合、ビデオエンコーダ20は、現在のブロックの符号化のための、モードm2の実現性についての確認をスキップまたはバイパスすることができる。しかしな

10

20

30

40

50

がら、現在のブロックに対するモードm1のRDコストが条件C1を満たさないとビデオエンコーダ20が判定する場合(144の「No」ブロック)、ビデオエンコーダ20はm2のRDコストを確認する(146)。この状況では、ビデオエンコーダ20は、モードm1と関連付けられるRDコストをモードm2と関連付けられるRDコストと比較することができる(148)。そして、ビデオエンコーダ20は、現在のブロックの符号化のために、より小さいRDコストと関連付けられるモードを選択することができる(150または152)。

【 0 0 5 9 】

図7Bのプロセス160は、図7Aに示されるプロセス140の逆を表す。たとえば、ビデオエンコーダ20は、プロセス160を実行して、モードm2と関連付けられるRDコストをまず確認することができる(162)。モードm2の確認された特性(この場合は関連するRDコスト)がある条件(本明細書では「C2」と呼ばれる)を満たす、または満足する場合(164の「Yes」の分岐)、ビデオエンコーダ20は、関連するブロックの符号化のためにモードm2を選ぶ(166)。したがって、現在のブロックに対するモードm2と関連付けられるRDコストが条件C2を満たすとビデオエンコーダ20が判定する場合(164の「Yes」の分岐)、ビデオエンコーダ20は、現在のブロックの符号化のための、モードm1の実現性についての確認をスキップまたはバイパスして、モードm2を選ぶことができる(166)。しかしながら、現在のブロックに対するモードm2のRDコストが条件C2を満たさないとビデオエンコーダ20が判定する場合(ブロック164の「No」の分岐)、ビデオエンコーダ20はm1のRDコストを確認する(168)。この状況では、ビデオエンコーダ20は、モードm2と関連付けられるRDコストをモードm1と関連付けられるRDコストと比較することができる(170)。そして、ビデオエンコーダ20は、現在のブロックの符号化のために、より小さいRDコストと関連付けられるモードを選択することができる(172または174)。

【 0 0 6 0 】

図7Aおよび図7Bに示されるようなモード選択技法は、モード確認のための固定された順序を代表する。しかしながら、図7Aおよび図7Bに示されるような固定順序のモード確認には、いくつかの潜在的な問題があり得る。1つのそのような潜在的な問題は、固定順序のモード確認がすべての事例において最良の複雑さ/品質のトレードオフを提供するとは限らないことである。図7Aのプロセス140の状況では、たとえば、複雑さまたは品質に関してモードm1がm2より良い可能性はるかに高い場合、ビデオエンコーダ20がモードm2のRDコストに基づいて好ましいモード(この場合、モードm1)の確認をスキップする可能性は小さく、またはわずかである。言い換えると、ビデオエンコーダ20がプロセス140を実施する事例においてモードm1がm2より好ましい場合、プロセス140は、不利なモードを評価するためにリソースを費やすことなく有利なモードを選択することを優先して、モード確認プロセスを歪曲し、または偏向させることに成功する。したがって、モードm1がモードm2より有利である特定の状況では、プロセス140はプロセス160よりも良好な複雑さ/品質のトレードオフを提供する。

【 0 0 6 1 】

しかしながら、複雑さまたは品質に関してモードm2がm1より良好である可能性が高い場合、プロセス140は、有利なモードの評価を開始する前であっても、それぞれの事例において不利なモードを評価することを優先して、モード選択プロセスを歪曲する/偏向させる。逆に、現在のブロックを符号化するのにモードm2がm1より良好である可能性が高い状況では、プロセス160は良好な複雑さ/品質のトレードオフを提供する。たとえば、現在のブロックを符号化することに関してモードm2がモードm1より小さなRDコストをもたらす可能性が高い場合、プロセス160はプロセス140より良好な複雑さ/品質のトレードオフを提供する。

【 0 0 6 2 】

したがって、HEVC規格に従った既存のモード確認順序は、すべての事例において最良の利用可能な複雑さ/品質のトレードオフを提供しないことがある。本開示の技法は、これらのリスクを軽減すること、およびより多数の状況において複雑さと品質のトレードオフを潜在的に改善することを対象とする。上で説明された問題に対処し、複雑さ/品質のト

10

20

30

40

50

リードオフを改善するために、ビデオエンコーダ20は、本明細書において説明されるような1つまたは複数の適応的なモード確認順序を実装し得る。

【0063】

図8は、本開示の1つまたは複数の態様による技法を実行することができるビデオエンコーダの例を示すブロック図である。図8の例では、ビデオエンコーダ20は、ビデオデータメモリ41と、モード選択ユニット40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、区分ユニット48とを含む。ビデオブロック再構築のために、ビデオエンコーダ20はまた、逆量子化ユニット58と、逆変換ユニット60と、加算器62とを含む。ブロック境界をフィルタリングしてブロックネスアーティファクトを再構築されたビデオから除去するために、デブロッキングフィルタ(図8に示さず)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の出力をフィルタリングする。追加のフィルタ(ループ内またはループ後)が、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の出力をフィルタリングし得る。いくつかの例では、デブロッキングフィルタが、上で説明されたように、色成分間予測のためのルーマサンプルのフィルタリングされたブロックを生成するために使用され得る。

10

【0064】

ビデオデータメモリ41は、ビデオエンコーダ20の構成要素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コーディングモードにおいて、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ため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、シンクロナスDRAM(SDRAM)を含むダイナミックランダムアクセスメモリ(DRAM)、磁気抵抗RAM(MRAM)、抵抗変化型RAM(RRAM(登録商標))、または他のタイプのメモリデバイスなどの、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提供され得る。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の他の構成要素とともにオンチップであることがあり、または、これらの構成要素に対してオフチップであることがある。

20

30

【0065】

符号化プロセスの間に、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。このようにして、ビデオエンコーダ20は、符号化されるべきビデオフレーム内の現在のビデオブロックを受信し得る。

【0066】

動き推定ユニット42および動き補償ユニット44は、時間予測を行うために、1つまたは複数の参照フレームの中の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。代替的に、イントラ予測ユニット46は、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の中の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、ビデオデータの各ブロックに対する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。

40

【0067】

加算器50は、コーディングされている現在のビデオブロックのピクセル値と予測ブロックのピクセル値との差を決定することによって、残差ビデオブロックを形成し得る。いくつかの例では、加算器50は残差ブロックを符号化しないことを決定し得る。

50

【 0 0 6 8 】

区分ユニット48は、以前のコーディングパスにおける以前の区分方式の評価に基づいて、ビデオデータのブロックをサブブロックに区分し得る。たとえば、区分ユニット48は、最初にフレームまたはスライスをLCUに区分し、レートひずみ分析(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。区分ユニット48は、本明細書において説明される高速RD確認モードまたは完全RD確認モードのいずれか、またはこれらの組合せを実行するように構成され得る。モード選択ユニット40は、LCUをサブCUに区分することを示す4分木データ構造をさらに作成し得る。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。モード選択ユニット40は、本明細書において説明される適応的なモード選択技法のいずれかまたはそれらの組合せを実行するよう

10

【 0 0 6 9 】

モード選択ユニット40は、たとえば誤差の結果に基づいて、複数のコーディングモード、すなわちイントラコーディングモードとインターコーディングモードのうちの1つを選択することができ、得られたイントラコーディングまたはインターコーディングされたブロックを加算器50に提供することができる。加算器50は残差ブロックデータを生成し得る。たとえば、加算器50は、残差ブロックデータの各サンプルが、現在のCUのコーディングブロックの中のサンプルと現在のCUのPUの予測ブロックの対応するサンプルとの差分に等しいように、現在のCUの残差ブロックデータを生成し得る。加算器62は、参照フレームとして使用するために、符号化されたブロック(すなわち、コーディングブロック)を再構築

20

【 0 0 7 0 】

HEVC規格のモード確認順序に関して上で説明された様々な問題に対処するために、モード選択ユニット40は、本開示の様々な適応的に順序付けられたモード確認技法を実施し得る。現在のブロックの特性により合うようにモード確認順序を適応的に変更することによって、モード選択ユニット40は、固定順序のモード確認に関して上で説明された問題に対処するために、場合によってはより良い複雑さ/品質のトレードオフを達成するために、本開示の態様を実施し得る。本開示の技法は、ビデオエンコーダ20のモード選択ユニット

30

【 0 0 7 1 】

モード選択ユニット40は、RDコストまたは周波数のメトリックなどの、メトリックを決定し得る。モード選択ユニットは、このメトリックが条件を満たすかどうかを判定し得る。この条件は、閾値のRDコストまたは閾値の周波数などの、閾値を備え得る。閾値の周波数および/またはメトリックは、いくつかの例では、特定のモード確認順序が発生する回数または比率を備え得る。様々な例において、メトリックは、RDの統計に、および/また

40

【 0 0 7 2 】

本開示の技法によれば、モード選択ユニット40は、メトリックが条件を満たすかどうかを判定し、メトリックが条件を満たすかどうかを判定することが統計に基づき、統計が第1のビデオ符号化モード確認順序および第2のビデオ符号化モード確認順序と関連付けられる、ステップと、メトリックが条件を満たすと判定したことに応答して、ビデオデータの第1のブロックを符号化するために第1の符号化モード確認順序を選択し、条件が満たされないと判定したことに応答して、ビデオデータの第1のブロックを符号化するために第2の符号化モード確認順序を選択し、選択された第1の符号化モード確認順序に基づいて、ま

50

たは第2の選択された符号化モード確認順序に基づいて統計を更新し、更新された統計に基づいて第2のブロックを符号化し、第1の符号化モード確認順序または第2の符号化モード確認順序を使用するように構成され得る。

【0073】

動き推定ユニット42および動き補償ユニット44は、高度に統合され得るが、概念的な目的のために別々に示されている。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生成するプロセスである。動きベクトルは、たとえば、現在のフレーム(またはコーディングされた他のユニット)内でコーディングされている現在のブロックに対する、参照フレーム(またはコーディングされた他のユニット)内の予測ブロックに対する、現在のビデオフレームまたはピクチャ内のビデオブロックのPUの変位を示し得る。言い換えると、動きベクトルは、PUの予測ブロックと参照ピクチャの中の対応する予測ブロックとの間の変位を示し得る。予測ブロックは、絶対差分和(SAD)、二乗差分和(SSD)、または他の差分のメトリックによって決定される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(すなわち、予測ブロック)と厳密に一致することが判明したブロックである。

10

【0074】

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。言い換えると、ビデオエンコーダ20は、1つまたは複数の補間フィルタを1つまたは複数の参照ピクチャのサンプルに適用して、PUの予測ブロックの中のサンプルを生成し得る。いくつかの例では、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、フルピクセル位置および分数ピクセル位置に対する動き探索を実行し、分数ピクセル精度で動きベクトルを出力し得る。

20

【0075】

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、インターコーディングされたスライス中のビデオブロックのPUの動きベクトルを計算し得る。参照ピクチャは、そ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、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)から選択され得る。動き推定ユニット42が動きベクトルを計算してある場合、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および動き補償ユニット44に送信し得る。

30

【0076】

動き補償ユニット44は動き補償を実行し得る。動き補償は、動き推定ユニット42によってPUのために決定される1つまたは複数の動きベクトルに基づいて、PUの1つまたは複数の予測ブロックをフェッチまたは生成することを伴い得る。やはり、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、機能的に統合されてもよい。現在のビデオブロックのPUの動きベクトルを受信すると、動き補償ユニット44は、動きベクトルに基づいて、参照ピクチャリストのうちの1つのピクチャから予測ブロックを位置特定し得る。一般に、動き推定ユニット42は、ルーマ成分に対する動き推定を実行し、動き補償ユニット44は、クロマ成分とルーマ成分の両方に対してルーマ成分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によって使用するための、ビデオブロックおよびビデオスライスと関連付けられたシンタックス要素を生成し得る。

40

【0077】

イントラ予測ユニット46は、上で説明されたように、動き推定ユニット42および動き補償ユニット44によって実行されるインター予測の代替として、現在のブロックをイントラ予測し得る。具体的には、イントラ予測ユニット46は、現在のブロックを符号化するために使用するイントラ予測モードを決定し得る。いくつかの例では、イントラ予測ユニット46は、たとえば、別々の符号化パスの間、様々なイントラ予測モードを使用して現在のブロックを符号化することができ、イントラ予測ユニット46(または、いくつかの例ではモ

50

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適切なイントラ予測モードを選択することができる。

【0078】

たとえば、イントラ予測ユニット46は、様々なテストされたイントラ予測モードに対してレートひずみ分析を使用してレートひずみ値を計算することができ、テストされたイントラ予測モードの中から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することができる。レートひずみ分析は、一般に、符号化されたブロックと、符号化されたブロックを作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(または誤差)の量、ならびに、符号化されたブロックを作成するために使用されるビットレート(すなわち、ビット数)を決定する。イントラ予測ユニット46は、どのイントラ予測モードがブロックに対して最良のレートひずみ値を示すかを決定するために、様々な符号化されたブロックに対するひずみおよびレートから比を計算することができる。

10

【0079】

ブロックのためのイントラ予測モードを選択した後、イントラ予測ユニット46は、ブロックのための選択されたイントラ予測モードを示す情報を、エントロピー符号化ユニット56に提供し得る。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)を含み得る、送信されるビットストリーム構成データに、様々なブロックのための符号化コンテキストの定義と、最もあり得るイントラ予測モードの指示と、イントラ予測モードインデックステーブルと、コンテキストの各々のために使用すべき変更されたイントラ予測モードインデックステーブルとを含め得る。

20

【0080】

ビデオエンコーダ20は、モード選択ユニット40からの予測データ(たとえば、予測ブロック)と、コーディングされている元のビデオブロック(たとえば、コーディングブロック)からのデータとの差分を決定することによって、残差ビデオブロックを形成し得る。加算器50は、この差分演算を実行する1つまたは複数の構成要素を表す。変換処理ユニット52は、変換を残差ブロックに適用して、残差変換係数値を備えるビデオブロック(すなわち、変換係数ブロック)を作成し得る。たとえば、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を適用して、残差係数値を作成し得る。

30

【0081】

変換処理ユニット52は、DCTと概念的に同様の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。いずれの場合にも、変換処理ユニット52は、残差ブロックに変換を適用し、残差変換係数のブロックを作成する。変換は、残差情報をピクセル(またはサンプル)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。

【0082】

さらに、変換ユニット52は、ACT(適応色変換)変換および/またはCCP(成分間予測)変換を残差データに適用し得る。さらに、本開示の1つまたは複数の技法によれば、変換ユニット52は、サンプル領域から変換領域への変換、ACT変換、および/またはCCP変換により生じる残差データのビット深度を低減するために、切り取り動作を残差データに適用し得る。

40

【0083】

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、係数の一部またはすべてと関連付けられ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。代替的には、エントロピー符号化ユニット56が走査を実行し得る。

【0084】

50

ビデオエンコーダ20は、コーディングされたビデオビットストリームにおいて様々なパラメータセットを符号化し得る。そのようなパラメータセットは、1つまたは複数のピクチャに共通のシンタックス要素を含み得るピクチャパラメータセット(PPS)と、ピクチャの1つまたは複数のシーケンスに共通のシンタックス要素を含み得るシーケンスパラメータセットとを含み得る。

【0085】

量子化に続いて、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。言い換えると、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表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、エントロピー符号化ユニット56は、コンテキスト適応型バイナリ算術コーディング(CABAC)、コンテキスト適応型可変長コーディング(CAVLC)、シンタックスベースコンテキスト適応型バイナリ算術コーディング(SBAC)、確率間隔区分エントロピー(PIPE)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のエントロピーコーディングの場合には、コンテキストは、隣接ブロックに基づき得る。エントロピー符号化ユニット56によるエントロピーコーディングに続いて、ビデオデコーダ30は、符号化されたビットストリームを、別のデバイス(たとえば、ビデオデコーダ30)へ送信し、または後の送信もしくは取出しのためにアーカイブし得る。

10

【0086】

エントロピーコーディング量子化された変換係数に加えて、エントロピー符号化ユニット56はスケール係数をエントロピー符号化するように構成され得る。様々な例において、エントロピー符号化ユニット56は、CABACコードのバイパスモードにおいて、固定長コードとしてスケール係数(アルファ)値をシグナリングするように構成され得る。

20

【0087】

逆量子化ユニット58および逆変換ユニット60は、たとえば、参照ブロックとして後で使用するための、ピクセル領域における残差ブロックを再構築するために、それぞれ、逆量子化および逆変換を適用する。たとえば、逆量子化ユニット58は、変換係数ブロックを逆量子化し得る。逆変換ユニット60は、逆変換を逆量子化された変換係数ブロックに適用することによって、TUのための変換ブロックを再構築し得る。

【0088】

加算器62は、復号ピクチャバッファ64へ記憶する再構築されたビデオブロックを作成するために、動き補償ユニット44によって作成された動き補償された予測ブロックに再構築された残差ブロックを加算する。動き推定ユニット42および動き補償ユニット44は、再構築されたビデオブロックを参照ブロックとして使用して、後続のビデオフレームの中のブロックをインターコーディング(すなわち、インター予測)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、再構築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。

30

【0089】

動き推定ユニット42は、ビデオエンコーダ20がインター予測される1つまたは複数のPUのピクセル値を予測するために使用し得る、1つまたは複数の参照ピクチャを決定することができる。動き推定ユニット42は、各参照ピクチャをLTRPまたは短期参照ピクチャとしてシグナリングし得る。動き推定ユニット42は、ピクチャが参照のために使用されないものとしてマークされるまで、復号ピクチャバッファ64に参照ピクチャを記憶し得る。ビデオエンコーダ20のモード選択ユニット40は、1つまたは複数の参照ピクチャの識別情報を含む、様々なシンタックス要素を符号化し得る。

40

【0090】

図9は、本開示の1つまたは複数の態様による、ビデオエンコーダがモード確認順序を適応的に選択するために実行することができる例示的なプロセスを示すフローチャートである。図9は、本開示の様々な態様による、モード選択ユニット40が実施し得るプロセス180を示す。モード選択ユニット40は、適応的な順序のモード確認のためのプロセス180を実施し得る。たとえば、モード選択ユニット40は、ある条件(図9において「C」と表記され

50

る)に基づいてビデオモード確認の順序を決定し得る。より具体的には、モード選択ユニットは、レートひずみメトリックなどのメトリックが条件Cを満足するかどうかを判定し得る。条件Cは、いくつかの例ではレートひずみ値であり得る。「順序1」は、プロセス140に関して図7Aにおいて示されるモード確認順序に対応し、「順序2」は、プロセス160に関して図7Bにおいて示されるモード確認順序に対応する。条件Cは、プロセス180のモード選択が、複雑さ/品質がより良好であることが予想される分岐をモード選択ユニット40が選択することを優先して歪曲または偏向されるような方法で、設計され得る。ビデオエンコーダ20のモード選択ユニットまたは他の構成要素はまた、利用可能な情報に基づいて、条件Cを適応的に調整することができる。モード選択ユニット40が条件Cを適応的に調整するために使用し得る利用可能な情報の例は、以前のピクチャの統計、現在のピクチャの分析前情報、および近隣のブロックからの情報または以前に確認されたモードの情報を含む。

10

【0091】

加えて、モード選択ユニット40は、それぞれ図7Aおよび図7Bに関して上で説明された条件C1(144)およびC2(164)と共同で、条件Cを較正および/または調節することができる。図7Aまたは図7Bのいずれかにおいて示される固定順序のモード確認技法の1つを実施することと比較して、図9のプロセス180のモード確認のための適応的な順序の選択は、モード選択ユニット40の動作に関する複雑さの低減をもたらし得る。いくつかの状況では、図9に示されるプロセス180は、プロセス140または160のいずれかと比較して、約二十パーセント(20%)の複雑さの低減をもたらし得る。モード選択ユニット40は、様々なタイプのモード確認の状況に加えて、またはその前に、プロセス180を実施し得る。

20

【0092】

図10は、例示的なプロセス200を示すフローチャートであり、このプロセスにおいて、モード選択ユニット40は、CUサイズを確認するための2つの可能な順序の中から選択する。より具体的には、モード選択ユニット40は、CUサイズの確認を適応的に順序付けるためにプロセス200を実施し得る。プロセス200の左(「YES」)の分岐は、図9のプロセス180の「順序1」の特定の事例を表す。同様に、プロセス200の右(「NO」)の分岐は、図9のプロセス180の「順序2」の特定の事例を表す。

【0093】

プロセス200の例では、モード選択ユニット40は、様々なCUの異なるサイズの確認を適応的に順序付けることができる。より具体的には、モード選択ユニット40は、 $2N \times 2N$ のCUサイズが最初に確認されるべきか(順序1)(204)、または $N \times N$ のCUが最初に確認されるべきか(順序2、206)を判定するために、プロセス200を実施し得る。図6のプロセス120に関連して、モードm1は $2N \times 2N$ のCUを最初に確認することを表すことがあり(204)、一方、モードm2は4つの $N \times N$ のサブCUを最初に確認することを表すことがある(206)。条件Cを決定するために、ビデオエンコーダ20は、 $2N \times 2N$ のCUの低コスト動き推定を実行することができ、これによって対応する予測誤差を得る。動き推定から得られる予測誤差の誤差エネルギーが下側の閾値(「 t_1 」)より小さい場合、ビデオエンコーダ20は、現在のブロックがバックグラウンドブロックである、または全体が動いている(たとえば、「一緒に動いている」)ブロックである可能性が高いと判定することができ、条件C1が満たされる。この場合、ビデオエンコーダ20は、CUにわたる低い誤差エネルギーによって示される共有される動き情報に基づき、 $2N \times 2N$ のCUを分割する可能性は低い。これに基づいて、プロセス200は、モード選択ユニット40が最初に $2N \times 2N$ のCUを確認し(204)、 $N \times N$ のサブCUを条件的に確認する(順序1)(208)ことを優先して、確認の順序を歪曲し、または偏向させることができる。

30

40

【0094】

しかしながら、誤差エネルギーが上側の閾値(「 t_2 」)を超えるとビデオエンコーダ20が判定する場合、ブロックは複雑であることがあるので、ビデオエンコーダ20は、符号化の際により小さいブロックサイズを使用する可能性が高い。この状況では、プロセス200は、モード選択ユニット40が4つの $N \times N$ のサブCUを最初に確認し(206)、 $2N \times 2N$ のCUを全体と

50

して条件的に確認する(210)ことを優先して、順序の選択を歪曲する/偏向させる。

【0095】

RDコストなどのメトリックがC1またはC2を満たすとビデオエンコーダ20が判定する場合、ビデオエンコーダ20はそれぞれ、確認順序1または確認順序2を終了することができる。ある例として、 $2N \times 2N$ のCUサイズと関連付けられるRDコストが閾値のRDコスト(たとえば、C1)より小さい場合、ビデオエンコーダ20は、 $2N \times 2N$ のCUの区分を選択することができ、 $N \times N$ のCUの区分サイズを確認しない。

【0096】

図11は、本開示の技法による、ビデオエンコーダが予測モード選択の適応的な順序付けのために実行することができる別の例示的なプロセスを示すフローチャートである。図11において、プロセス220によれば、モード選択ユニット40は、PUのために使用されるべき予測モードを適応的に確認することができ、図6のモードm1およびm2が2つの異なる予測モードに対応する。いくつかの例では、プロセス220は、ビデオエンコーダ20が時間レイヤのサポートのための予測構造に関して実施し得るプロセスを表す。

10

【0097】

図11の具体的な例において、(上で説明される)モードm1およびm2はそれぞれ、AMVPモードおよびSKIPモードに対応する。AMVPモードおよびSKIPモードは非限定的な例として図11に示されており、本開示の適応的な順序付けの技法は、予測モードのいくつかの可能な組合せおよび置換をサポートするために適用され得ることが理解されるだろう。プロセス220に従ってPUの予測モード確認を適応的に順序付けることによって、ビデオエンコーダ20

20

【0098】

ビデオエンコーダ20は、ピクチャを符号化するための3つの時間レイヤをサポートし得る。いくつかのピクチャは単方向予測されることだけが可能であり、1つの参照ピクチャが、現在のピクチャから比較的離れて位置している(すなわち、大きく異なる)ことがある。結果として、ビデオエンコーダ20は、そのようなピクチャを符号化するためにSKIPモードを選択する可能性がより低いことがある。そのような事例では、本開示の技法は、モード選択ユニットがAMVPモードを最初に評価し、SKIPモードを条件的に評価するように、モード確認順序を歪曲し、または偏向させることができる。しかしながら、双方向予測されるフレームは、双方向予測されるフレームに比較的(視覚的に)近い2つの参照ピクチャから予測され得る。双方向予測されるピクチャに対しては、SKIPモードは主要な選択であり得る。したがって、双方向予測されるピクチャの場合、モード選択ユニット40は、SKIPモードを最初に確認することを優先して、順序確認プロセスを偏向させることができる。

30

【0099】

さらにいくつかの他のフレームに対しては、モード選択ユニット40は、動きの小さいシーケンスに対しては最初にSKIPモードを確認することによって、および動きの大きいシーケンスに対しては最初にAMVPを確認することによって、より良好な複雑さ/品質のトレードオフを生み出すことができる。モード選択ユニット40がSKIPモードとAMVPモードの中から選択する状況では、モード選択ユニットは、現在のピクチャの時間インスタンスと異なる時間インスタンスT1と関連付けられる以前にコーディングされたピクチャにおけるSKIPモードとAMVPモードの発生の回数に基づくように、条件Cを編成することができる(222)。言い換えると、SKIPモードとAMVPモードの中から選択するために、モード選択ユニット40は、SKIPモードコーディングの発生が時間T1のピクチャのブロックにおいて優勢である、またはより頻繁である場合、SKIPモードを最初に確認することができる(224)。逆に、モード選択ユニット40は、AMVPモードコーディングの発生が時間T1のピクチャのブロックにおいてより頻繁である、または優勢である場合、AMVPモードを最初に確認する(226)ように、本開示の技法を実施することができる。

40

【0100】

順序1においてCUのためにスキップモードを確認した後で(224)、条件C1が満たされない場合、モード選択ユニット40はAMVPモードを確認することができる(228)。逆に、順序2に

50

において、モード選択ユニット40がAMVPモードを確認した後で(226)、条件C2が満たされない場合、モード選択ユニット40はCUのためにSKIPモードを確認する(230)。このようにして、モード選択ユニット40は、本開示の適応的に順序付けられたモード確認技法を実施して、すでに符号化されているピクチャから入手可能なヒューリスティックデータを利用することができる。

【0101】

順序1においてSKIPモードを確認した後で(224)、または順序2においてAMVPモードを確認した後で(226)、ビデオエンコーダ20は、確認されたモードのRDコスト(メトリック)を決定することができる。RDコストのメトリックが閾値のコスト(たとえば、C1またはC2)より小さい場合、これは条件C1またはC2の「Yes」の分岐に相当することがあり、ビデオエンコーダ20は、条件(たとえば、C1またはC2)を満足するビデオコーディングモードを選択し、他のビデオコーディングモードのRDコストを確認するのを止めることができる。

10

【0102】

図12は、本開示に態様による、時間レイヤのサポートのための例示的な予測構造を示す概念図である。図11のプロセス220の例を適用すると、モード選択ユニット40は、現在のブロックを符号化するためにSKIPモードとAMVPモードの中から選択することができる。図11に示されるように、ビデオエンコーダ20は、T0、T1、およびT2と表記される、ビデオデータの3つの時間レイヤをサポートし得る。数1~12の各々は、その時間レイヤと関連付けられるピクチャに対応し得る。

【0103】

20

図12に示される予測構造は、時間レイヤT0の中のピクチャに対して、0、4、8、...、などに等しい数Nを有するピクチャは値N-4を有する別のピクチャを予測することだけが可能であるというものとして、説明され得る。時間レイヤT1のピクチャに対して、2、6、10、...、などに等しいNを有するフレームは、N1-2および(N2+2)に等しい値を有するフレームを予測するために使用されることが可能である。1、3、5、7、...、などに等しい値Nを有する時間レイヤT2のピクチャは、値N-1およびN+1を有するピクチャを予測するために使用され得る。

【0104】

したがって、レイヤT0のピクチャは、単方向予測されることのみが可能である。レイヤT0のピクチャの参照ピクチャは、レイヤT1およびT2のピクチャと比べて、相対的に遠く離れている。この予測方式の結果として、レイヤT1は他のピクチャに基づいて予測される可能性がより高いので、SKIPモードを使用して最適に符号化される可能性がより低い。したがって、モード選択ユニット40は、時間レイヤT1の中の単方向予測されるフレームに対してSKIPモードを選択する可能性がより低いことがある。したがって、モード選択ユニット40は、レイヤT1のピクチャに対してAMVPモードが最初に(無条件で)確認されることを優先して選択プロセスを歪曲するように、本開示の適応的な確認順序を実施することができる。

30

【0105】

しかしながら、時間レイヤT2のピクチャは双方向予測され得るので、2つの参照フレームが非常に近い。よって、時間レイヤT2のフレームは、双方向予測を使用して相対的に正確に予測されることが可能であり、SKIPモードはそのようなフレームの主要なコーディングモードであり得る。時間レイヤT2の中の双方向予測されるフレームの場合、有利なモード確認プロセスは、SKIPモードを最初に(無条件で)確認することを含む。したがって、モード選択ユニット40は、SKIPモードを最初に(無条件で)確認することを優先してモード確認順序を歪曲するように、本開示の適応的な順序付け技法を実施することができる。

40

【0106】

時間レイヤT1の中のフレームの場合、モード選択ユニット40は、動きの小さいシーケンスに対しては最初にSKIPモードを確認し、動きの大きいシーケンスに対しては最初にAMVPを確認することを優先して、モード確認順序を偏向させるように構成され得る。モード選択ユニット40は、時間レイヤT1の以前のフレームにおけるSKIPモードおよびAMVPモードの

50

発生に基づいて、図11のプロセス220において示されるように条件Cを編成することができる。たとえば、モード選択ユニット40は、SKIPモード符号化の発生がAMVPモード符号化の発生よりも以前のフレームにおいて頻繁である場合、SKIPモードを最初に確認するために、時間レイヤT1の表示順序にて以前のフレームからのヒューリスティックデータを利用することができる。

【0107】

図13Aおよび図13Bは、本開示の1つまたは複数の態様による、ビデオエンコーダがPU区分方式を適応的に決定するために使用することができる、対応する 8×8 のCUの誤差分布を示す概念図である。図13Aおよび図13Bはそれぞれ、対応する 8×8 のCUに対する、誤差予測分布240Aおよび240Bを示す。本開示の技法の例によれば、ビデオエンコーダ20は、誤差分布に基づいてPU区分方式を適応的に決定することができる。たとえば、ビデオエンコーダ20は、この技法を実施して、ブロックの係数エネルギーに基づいてPU区分サイズを決定することができる。

10

【0108】

1つの例示的な使用事例において、ビデオエンコーダ20は 8×8 のPUをすでに確認しており、ビデオエンコーダ20は、 8×4 のPUを最初に確認することと、 4×8 のPUを最初に確認することの中から選択しなければならない。 8×8 のPUの予測誤差エネルギーが図13Aに示されるように 8×8 のCUの下半分に集中している場合(8×8 のPUが横方向に分かれるとき)、エネルギー分布は通常、 8×8 のPUの下半分が上半分に対して異なるように動くことを示す。誤差分布240Aのエネルギー分布のこれらの傾向に基づいて、モード選択ユニット40は、ビデオエンコーダ20が 8×8 のPUを2つの 8×4 ($2N \times N$)のPUに区分する可能性がより高いと判定し得る。そして、モード選択ユニット40は、 $2N \times N$ の区分方式を最初に、すなわち他の区分方式の前に確認することを優先して、本開示の適応的に順序付けられるモード確認技法を歪曲する/偏向させることができる。

20

【0109】

図13Bの例では、誤差分布240B、予測誤差エネルギーが、図13Bに示されるように(8×8 のPUが縦方向に分かれるとき) 8×8 のPUの右半分に集中している。誤差分布240Bのエネルギー分布は通常、 8×8 のPUの右半分が左半分に対して異なるように動くことを示す。誤差分布240Bのエネルギー分布のこれらの傾向に基づいて、モード選択ユニット40は、ビデオエンコーダ20が 8×8 のPUを2つの 4×8 ($N \times 2N$)のPUに区分する可能性がより高いと判定し得る。したがって、モード選択ユニット40は、 $N \times 2N$ の区分方式を最初に確認することを優先して、本開示の適応的に順序付けられるモード確認技法を歪曲する/偏向させることができる。

30

【0110】

図14は、TUの区分方式を適応的に決定するためにビデオエンコーダ20が本開示の態様に従って実施し得る、例示的なプロセス260を示すフローチャートである。ビデオエンコーダ20は、条件Cに基づいてTUの区分方式を決定し得る。ビデオエンコーダ20は、RDコストのメトリックが条件Cを備える閾値より小さいか大きいかを判定し得る。条件Cが満たされるとビデオエンコーダ20が判定する場合、ビデオエンコーダ20は、 8×8 のTUサイズを最初に確認し(262)、 8×8 のTUサイズに対するRDコストを決定する。別の条件C1が満たされない場合、ビデオエンコーダ20は次いで、 4×4 のTUサイズを確認する(264)。Cが満たされない場合、ビデオエンコーダ20は、 4×4 のTUサイズを最初に確認し(266)、 4×4 のTUサイズと関連付けられるRDコストを決定する。条件C2が満たされない場合、ビデオエンコーダ20は、 8×8 のTUサイズを確認する(268)。

40

【0111】

8×8 のTUサイズ(262)と関連付けられるRDコスト、および 4×4 のTUサイズ(266)と関連付けられるRDコストがそれぞれ、条件C1またはC2を満足するとビデオエンコーダ20が判定する場合、ビデオエンコーダ20は、ビデオ符号化モード確認プロセスを終了する。すなわち、ビデオエンコーダ20は、関連するRDコストが閾値コストC1より小さい場合、ブロックに対して 8×8 のTUサイズを選択し得る。同様に、ビデオエンコーダ20は、関連するRDコスト

50

が閾値コストC2より小さい場合、 4×4 のTUサイズを選択し得る。C1またはC2のいずれかが選択される場合、ビデオエンコーダ20は他のTU区分サイズを確認するのを止める。

【0112】

ビデオエンコーダ20は、ブロックの係数エネルギーに基づいてTU区分方式を適応的に決定するために、プロセス260を実施し得る。より具体的には、ビデオエンコーダ20は、どのTU区分がCUに対して最初に確認されるべきかを決定するためにプロセス260を実施することができ、ここでモードm1およびm2は、図5に示されるTU区分方式116A～116Cのうちの2つの異なる区分方式と関連付けられる。図14のプロセス260の例では、評価されるべき2つのTUサイズは 8×8 および 4×4 である。

【0113】

図15Aおよび図15Bは、予測誤差分布を示す概念図である。ビデオエンコーダ20はまた、どのTUサイズが上で説明されたアルゴリズムにおいて最初に確認されるべきかを決定するために、予測誤差分布を使用し得る。図15Aおよび図15Bはそれぞれ、誤差分布280Aおよび280Bを示す。図15Aは、(誤差分布280Aによって) 8×8 のブロックに対する予測誤差を示す。誤差分布280Aは比較的均一であるので、 8×8 の変換は通常、さらなる区分よりも符号化の観点からは良好である。したがって、誤差分布280Aに関して、最初に 8×8 のTUを確認することは、より有益な選択肢を代表する。したがって、モード選択ユニット40は、 8×8 のTUを最初に確認することを優先して順序付けプロセスを歪曲し、または偏向させるように、本明細書において説明される適応的な順序付け技法を実施し得る。

【0114】

しかしながら、1つの 4×4 のブロックの予測誤差がより大きな 8×8 のブロックにおいて支配的である場合(これは図15Bの誤差分布280Bにおいて当てはまる)、 8×8 の変換を使用することは、エネルギーが主に位置する係数領域へと誤差エネルギーを局限するのではなく、 8×8 のブロック全体にわたって誤差エネルギーを拡散することがある。したがって、 8×8 の変換サイズは、誤差分布280Bの場合、より多くの0ではない係数のコーディングをもたらすことがあり、不鮮明にするアーティファクトを作り出すことがある。誤差分布280Bの場合、モード選択ユニット40は、 4×4 の変換サイズを最初に確認することを優先して選択を偏向させるように、TUサイズ確認プロセスを適応的に順序付けることができる。

【0115】

図16は、ビデオエンコーダ20がPUの予測モードの確認を適応的に順序付けるために実施し得る、例示的なプロセス300を示すフローチャートである。プロセス300の例では、モード選択ユニット40は、PU区分サイズ $2N \times N$ (302)および $N \times 2N$ (304)にそれぞれ対応するモードm1とm2の中から選択する。ビデオエンコーダ20がプロセス300において評価するPU区分サイズの例が、図13Aおよび図13Bに示されている。たとえば、プロセス300の文脈において、図13Aの誤差分布240Aは、順序1が有益である分布を表し得るが、図13Bの誤差分布240Bは、順序2が有益である分布を表し得る。

【0116】

順序1では、モード選択ユニット40が $2N \times N$ の区分を確認し、関連するRDコストを決定した後で、モード選択ユニット40は、RDコストがC1と表記される条件を満たすかどうかを判定する(306)。C1が満たされないとモード選択ユニット40が判定する場合、モード選択ユニット40は $N \times 2N$ のPU区分を確認する(308)。順序2では、モード選択ユニット40が $N \times 2N$ の区分(304)を確認し、 $N \times 2N$ の区分と関連付けられるRDコストを決定した後で、モード選択ユニット40は条件C2を確認する(310)。C2が満たされないとモード選択ユニット40が判定する場合、モード選択ユニット40はPUの $2N \times N$ の区分を確認する(312)。条件C1またはC2が満たされる、たとえば、 $2N \times N$ のPUサイズ(302)と関連付けられるRDコスト、または $N \times 2N$ のPUサイズ(304)と関連付けられるRDコストが閾値のRDコストより小さいとビデオエンコーダ20が判定する場合、ビデオエンコーダ20はPU区分サイズ確認順序を終了することができる。

【0117】

このようにして、ビデオエンコーダ20は、ビデオデータのブロックのための第1のビデオ

10

20

30

40

50

オ符号化パラメータを第1のビデオ符号化パラメータと関連付けられる統計に基づいて適応的に決定し、第1のビデオ符号化パラメータと関連付けられるメトリックがビデオ符号化パラメータと関連付けられる条件を満たすかどうかを判定し、第1のビデオ符号化パラメータが条件を満たす場合、ブロックを符号化するために第1のビデオ符号化パラメータと関連付けられる第1の符号化モード確認順序を使用し、第1のビデオコーディングパラメータが条件を満たさない場合、ブロックを符号化するために第2のビデオ符号化パラメータと関連付けられる第2の符号化モード確認順序を使用し、第1の符号化パラメータと関連付けられる統計および第2の符号化パラメータと関連付けられる統計を更新し、第1のビデオ符号化パラメータおよび第2のビデオ符号化パラメータと関連付けられる更新された統計を使用して第2のブロックを適応的に符号化するように構成される、デバイスの例を表し得る。

10

【0118】

図17A～図17Cは、本開示による、ビデオエンコーダ20がブロック区分の決定を用いて高速な符号化の決定(たとえば、高速RD確認と完全RD確認の中から選択すること)を実行するために実施し得る、例示的なプロセス320を示す流れ図である。プロセス320を実施することによって、ビデオエンコーダ20は、視覚的な品質を下げることなく高速な符号化を達成することができる。

【0119】

本明細書において説明されるように、ビデオエンコーダ20は、符号化モード、区分方式、およびこれらの組合せなどの、様々な符号化パラメータを決定するために、RDコストを決定する。本明細書においてより詳細に説明されるように、ビデオエンコーダ20は、ビデオデータのより高速な符号化を、他の高速ビデオ符号化モードと比べて視覚的な品質の低下を少なくしながら実行するために、RD決定の早期終了を利用することができる。図17A～図17Cに示されるようなプロセス320は、そのような早期終了プロセスの例である。

20

【0120】

プロセス320において、ビデオエンコーダ20は、最大のCUに対してRD最適化されたモード選択を実行して、関連するRDコストを決定し得る(322)。図17Aの例では、CUは 32×32 というサイズを有し得る。そして、選択されたコーディングモードおよび最大のCUにおけるRDコストの量に基づいて、ビデオエンコーダ20は、完全RD確認を実行するか、高速RD確認を実行するかを判定し得る。

30

【0121】

モード選択ユニット40は、現在のスライスがIスライスである事例、または高速符号化モード機能がビデオエンコーダ20上で無効にされている(または「オフにされている」)(324の「Yes」の分岐)事例において、完全RD確認を選択することができる。モード選択ユニット40は、ピクチャのエリアが複雑である(たとえば、大きな変動を有する)場合、完全RD確認(たとえば、判断木の完全RD確認の分岐)を選択することができる(326)。現在のスライスがIスライスではなく、高速符号化モード機能がビデオエンコーダ20上で無効にされていない(または「オフにされていない」)(324の「No」の分岐)事例では、モード選択ユニット40は、「スキップ」が 32×32 のブロックに対して最良のモードであるかどうかを判定することができる(328)。ビデオエンコーダ20はまた、最大のCUのRDコストが閾値より大きい事例において、完全RD確認の選択肢を選択することができ、または、最大のCUに対する選択されたモードはSKIPモードではない(判断ブロック328の「No」の分岐)。それ以外の場合、ビデオエンコーダ20は、円Aに対応する高速RDの選択肢(たとえば、判断木の高速RD分岐)を選択することができる。高速RD分岐は図17Bに示されている。

40

【0122】

図17Bに対応する、高速RDの分岐が選択される事例では、最大のCU(この例では、 32×32 のCUのRD、 L_{32})のひずみが閾値より小さい場合(330の「No」の分岐)、ビデオエンコーダ20は、それぞれ最後のCUおよびコーディングモードとして上で選択された、最大のCUおよびコーディングモードを選択することができる(336)。図17Bの例では、最大の区分は 32×32 の区分である。そして、ビデオエンコーダ20は、CU区分プロセスを終了し得る(図17Aの

50

「終了」ブロックを参照する、円「C」)。

【0123】

しかしながら、最大のCUサイズのひずみが閾値より小さくない場合(判断ブロック330の「Yes」の分岐)、ビデオエンコーダ20は、最大のCUの各サブ区分ブロックのひずみを計算することができる(332)。4つのブロックの各々のひずみはそれぞれ、 D_1 、 D_2 、 D_3 、および D_4 と表記される。平均のひずみは μ と表記される。

【0124】

サブブロックのうちのある1つのひずみが、その大きさが μ と表記されるある程度/量/値だけ、平均のひずみの閾値に等しい閾値を超える場合(判断ブロック334の「Yes」の分岐)、ビデオエンコーダ20は、現在のサブブロックがピクチャの他のブロックと異なる動きを有し、図17Cにおいてより詳細に説明される円「B」に進むと、決定することができる。この場合、ビデオエンコーダ20が最大のCUを使用する場合、視覚的なアーティファクトが発生し得る。したがって、図17Cに対応する円Bにおいて説明されるように、特定のサブブロックのひずみがある量だけ閾値を超える場合、最大のCUを区分することが望ましいことがある。CUをさらに区分するプロセスは、図17Cにおいてより詳細に説明される。それ以外の場合、ビデオエンコーダ20は、それぞれ最後のCUおよびコーディングモードとして上で選択された、最大のCUおよびコーディングモードを選択することができる(336)。

【0125】

図17Cでは、ビデオエンコーダ20は、CUをサブCUにさらに区分するように構成され得る。CUをサブCUに区分するかどうかを判定するために、ビデオエンコーダ20は、その区分のひずみ D_i が閾値のひずみより大きいかどうか、および、その区分がCU内の他の区分と比べて異なる動き情報を有するかどうかを判定することができる(338)。

【0126】

区分が異なる動き情報を有し、第2の閾値のひずみコスト Th_2 を超える場合(判断ブロック338の「Yes」の分岐)、ビデオエンコーダ20は、 16×16 の区分を 8×8 の区分に区分し(340)、 8×8 の区分のRDコストを 16×16 の区分サイズおよび 32×32 の区分サイズのRDコストと比較することができる。ビデオエンコーダ20は、RDコストを最適にする、たとえば最低のRDコストを生み出す区分を選択することができる(342)。そうではなく、区分が異なる動き情報を有さず、第2の閾値のひずみコスト Th_2 を超えない場合(判断ブロック338の「No」の分岐)、ビデオエンコーダ20は、どちらの区分方式がRDコストを最適にするかに応じて、 32×32 または 16×16 の区分方式を使用することができる(344)。

【0127】

したがって、本開示の技法によれば、ビデオエンコーダ20は、ビデオデータのブロックを符号化することと関連付けられるレートひずみ(RD)コストを決定し、RDコストを決定することがRDコストを決定するために完全RD確認方式を使用するか高速RD確認方式を使用するかを決定することを備え、決定されたRDコストに基づいてブロックの区分方式を決定し、決定されたRDコストに基づいて決定された区分方式を使用してブロックを符号化するように構成され得る。

【0128】

ビデオエンコーダ20が各サブブロックのひずみを評価するために使用し得る技法は、ひずみを平均のひずみと比較することである。たとえば、ビデオエンコーダ20は、4つのサブブロックのひずみを計算し、4つの値の平均のひずみを計算することができる。ビデオエンコーダ20は、二乗誤差の平均(合計)または絶対誤差の平均(誤差)などの、様々なひずみのメトリックの1つまたは複数を使用し得る。様々な例において、ビデオエンコーダ20は、ルーマ成分のひずみだけを、またはルーマ成分とクロマ成分の両方のひずみを使用し得る。いくつかの例では、ビデオエンコーダ20は、異なる加重係数を使用してルーマひずみとクロマひずみを加算し得る。加えて、ビデオエンコーダ20は、あらかじめ定められた係数によって平均をスケールリングすることによって、閾値を設定することができる。1つまたは複数のブロックが閾値より大きなひずみを有する場合、ビデオエンコーダ20は、そのようなブロックを区分し得る。それ以外の場合、ビデオエンコーダ20は、それぞれ最後

10

20

30

40

50

のCUおよびコーディングモードとして、最大のCUおよび対応するコーディングモードを選択することができる。そして、ビデオエンコーダ20は、CU区分プロセスを終了することができる。スケーリング係数は、ビデオエンコーダ20がブロックに適用する区分の量を制御することができる。スケーリング係数が小さい場合、ビデオエンコーダ20は、より大きな量の区分を実行することがあり、場合によっては、より良好なコーディング効率および視覚的な品質(ピクチャ品質)をもたらす。しかしながら、スケーリング係数が大きい場合、ビデオエンコーダ20は、より少量の区分を実行することがあり、場合によっては、より高速な符号化速度をもたらす。

【0129】

別の例示的な実装形態によれば、ビデオエンコーダ20は、サブブロックの間でのひずみのばらつきの測定値を使用し得る。たとえば、4つのサブブロックのひずみのばらつきまたは標準偏差が大きい(たとえば、所定の閾値を超える)場合、区分がブロックの符号化に有益である可能性がより高いことがある。しかしながら、ばらつきまたは標準偏差は、サンプル値(4つのサブブロックのひずみ)の大きさが増大するにつれて、より大きくなる傾向にある。いくつかの事例では、ひずみの量は、局所的な特性に基づいて、ビデオフレームにおいて大きく変動し得る。したがって、ばらつきまたは標準偏差の代わりに、ビデオエンコーダ20は正規化された標準偏差を使用することがあり、またはばらつきの係数が使用され得る。正規化された標準偏差またはばらつきの係数は、以下の式に従って定義され得る。

$$c = \frac{\sigma}{\mu}$$

【0130】

上の式において、 μ および σ はそれぞれ、4つのサブブロックのひずみの平均および標準偏差を示す。加えて、ビデオエンコーダ20は、現在のCUのばらつきの係数「c」を閾値と比較し得る。cが閾値を超える場合、ビデオエンコーダ20は、現在のブロックを区分し得る。一方、cが閾値より小さい場合、ビデオエンコーダ20は、最後のCUおよびコーディングモードとして選択された最大のCUおよび対応するコーディングモードを選択することができ、CU区分プロセスを終了することができる。cを計算するための演算の数を減らすために、ビデオエンコーダ20は、様々な例においては σ の代わりに平均の絶対偏差を使用し得る。また、除算演算を避けるために、ビデオエンコーダ20は、 μ を閾値と乗算することができ、 $\mu \cdot c$ または平均の絶対偏差を、 μ と閾値の積と比較することができる。

【0131】

符号化時間をさらに減らすために、ビデオエンコーダ20は、各サブブロックに対してRDコスト確認を部分的に実行することができる。たとえば、ビデオエンコーダ20は、イントラコーディングをスキップすることができる。加えて、ビデオエンコーダ20は、SKIPモードのRDコストを最初に確認することができる。SKIPモードからのひずみが閾値より小さい場合、ビデオエンコーダ20は、他のモードの確認をスキップすることができる。

【0132】

CU区分方式選択およびモード選択に関して上で説明された技法は、ビデオ符号化デバイス20が、最大のCUを区分するかどうかを判定し、4分木の0番目(たとえば、ルート)および1番目の深度に対応するサブブロックのコーディングモードを選択することを可能にし得る。ビデオエンコーダ20は、4分木区分方式に関して、より大きな深度に対して同様のまたは同じ手順を実施することができる。代替的に、ビデオエンコーダ20は、より大きな深度に対して簡略化された深度を適用することができる。たとえば、第1の深度のブロックのひずみが閾値より大きい場合、ビデオエンコーダ20はさらなる区分を実行することができる。加えて、ビデオエンコーダ20は、1番目の深度レベルにあるブロックが0番目の深度レベルにあるブロックと同じ動き情報を共有するかどうかを確認することができる。1番目の深度レベルにあるブロックが0番目の深度レベルにあるブロックと同じ動き情報を共有しない場合、ビデオエンコーダ20はさらなる区分を実行することができる。

【0133】

図18は、符号化利得に関して、ブロック区分の決定を用いた高速符号化の実行の例を示

10

20

30

40

50

すグラフを示す概念図である。図18のグラフ340において、ひずみレートがx軸にプロットされ、符号化速度がy軸にプロットされる。「Test1」は、SKIPモードが32×32のCUに対して選択されるときにビデオエンコーダ20が常に区分をスキップし、それによってより高速な符号化速度をもたらす使用事例である。Test1は、他の使用事例の場合の速度および符号化利得を測定する際に比較の対象とする、アンカーまたは基準として使用され得る。「Test2」は、ビデオエンコーダ20がサブブロックのひずみを比較することなく区分を実行する使用事例である。グラフ340に示されるように、Test2はコーディング速度が相対的により遅い使用事例を代表する。

【0134】

Test1およびTest2の使用事例と比較して、本開示の高速符号化決定技法(たとえば、高速RD確認および完全RD確認からの選択)と区分決定技法(1および2としてプロットされる)は、7~12%の符号化速度の低下とともに、コーディング効率の改善を示している。2は、1が基づくスケール係数より小さいスケール係数に基づく。加えて、本開示の高速符号化技法および区分技法と関連付けられる1および2は、図19に示されるTest1の使用事例と比較して、視覚品質の改善をもたらす。

【0135】

図19は、制御パラメータに従って符号化される画像と本開示の高速符号化および区分の決定に従って符号化される画像とのピクチャ品質の比較を示す概念図である。図19は、破線によって区別された2つの画像を並べて示す。図19では、左側に位置する画像は、2つの円、すなわち左の円360および右の円362を含む。左側の画像は、図18のグラフ340にプロットされたTest1の使用事例のパラメータ(たとえば、区分サイズおよび符号化速度)を使用して符号化することに対応する。左の画像の左の円360の中のエリアはブロッキングアーティファクトを示しており、右の円362の中のエリアは物体の切断を示している。

【0136】

図19の右に位置する画像は、本開示の高速符号化技法および区分決定技法に従った符号化に対応する。1つの特定の例では、図19の右の画像は、図18のグラフ340にプロットされた1使用事例のパラメータに従った符号化により生じる。左の画像のブロッキングアーティファクトおよび物体の切断は、図19の右に位置する画像において示されるように、1パラメータを使用して視覚的に改善される(たとえば、軽減され、または除去すらされる)。Test2の使用事例により生じる画像と比較すると、視覚的品質は同様/同等であるが、符号化速度ははるかに速い。このようにして、ビデオエンコーダ20は、符号化速度および視覚的品質の両方を改善するために(たとえば、RDを下げることによって)、本開示の区分および符号化速度の決定を実施することができる。

【0137】

このようにして、ビデオエンコーダ20は、本開示の例によれば、ビデオデータのブロックを符号化することと関連付けられるレートひずみ(RD)コストを決定し、RDコストを決定することがRDコストを決定するために完全RD確認方式を使用するか高速RD確認方式を使用するかを決定することを備え、決定されたRDコストに基づいてブロックの区分方式を決定し、決定されたRDコストに基づいて決定された区分方式を使用してブロックを符号化するように構成される、デバイスを代表し得る。

【0138】

図20は、本開示の1つまたは複数の態様による技法を実行することができるビデオデコーダの例を示すブロック図である。図20の例では、ビデオデコーダ30は、ビデオデータメモリ69と、エントロピー復号ユニット70と、動き補償ユニット72と、イントラ予測ユニット74と、逆量子化ユニット76と、逆変換ユニット78と、復号ピクチャバッファ82と、加算器80とを含む。いくつかの例では、ビデオデコーダ30は、ビデオエンコーダ20(図8)に関して説明された符号化パスとは全般に逆の復号パスを実行し得る。

【0139】

ビデオデータメモリ69は、ビデオデコーダ30の構成要素によって復号されるべき、符号化された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。ビデオデータメモリ

10

20

30

40

50

69に記憶されるビデオデータは、たとえば、通信チャネル16から、たとえば、カメラなどのローカルビデオソースから、ビデオデータの有線ネットワーク通信もしくはワイヤレスネットワーク通信を介して、また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69は、符号化されたビデオビットストリームからの符号化されたビデオデータを記憶するコーディングピクチャバッファ(CPB)を形成し得る。復号ピクチャバッファ82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において、ビデオデコーダ30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69および復号ピクチャバッファ82は、シンクロナスDRAM(SDRAM)を含むダイナミックランダムアクセスメモリ(DRAM)、磁気抵抗RAM(MRAM)、抵抗変化型RAM(RRAM(登録商標))、または他のタイプのメモリデバイスなどの、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69および復号ピクチャバッファ82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提供され得る。様々な例では、ビデオデータメモリ69は、ビデオデコーダ30の他の構成要素とともにオンチップであることがあり、または、それらの構成要素に対してオフチップであることがある。

【0140】

復号プロセスの間、ビデオデコーダ30は、ビデオエンコーダ20から、符号化されたビデオスライスのビデオブロックと関連するシンタックス要素および/またはシンタックスデータとを表す符号化されたビデオビットストリームを受信する。ビデオデコーダ30のエントロピー復号ユニット70は、ビットストリームをエントロピー復号して、量子化された係数と、動きベクトルまたはイントラ予測モードインジケータと、他のシンタックス要素とを生成する。エントロピー復号ユニット70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72に転送し得る。

【0141】

エントロピー復号ユニット70は、様々なパラメータセットの中の追加のシンタックス要素を復号して構文解析し得る。そのようなパラメータセットは、1つまたは複数のピクチャに共通のシンタックス要素を含み得るPPSと、ピクチャの1つまたは複数のシーケンスに共通のシンタックス要素を含み得るSPSとを含み得る。

【0142】

ビデオデコーダ30は、復号ピクチャバッファ82に記憶された参照ピクチャに基づいて、(たとえば、デフォルトの構築技法を使用して)参照ピクチャリスト、すなわちリスト0およびリスト1を構築し得る。ビデオスライスがイントラコーディングされる(I)スライスとしてコーディングされる時、イントラ予測ユニット74は、現在のビデオスライスのビデオブロックの予測データを生成し得る。イントラ予測ユニット74は、現在のフレームまたはピクチャの以前に復号されたブロックからのシグナリングされたイントラ予測モードおよびデータに基づいて、予測データを生成し得る。ビデオデコーダ30がビデオフレームのスライスをインターコーディングされる(すなわち、BまたはP)スライスとしてコーディングするとき、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と他のシンタックス要素とに基づいて、現在のビデオスライスのビデオブロックの予測ブロックを作成し得る。動き補償ユニット72は、参照ピクチャリストのうちの1つの中の参照ピクチャのうちの1つから予測ブロックを作成し得る。

【0143】

動き補償ユニット72は、動きベクトルおよび/またはシンタックス要素を使用して、現在のビデオスライスのビデオブロックの予測情報を決定し得る。いくつかの例では、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、予測情報を生成し得る。動き補償ユニット72は、予測情報を使用して、復号されている現在のビデオブロックの予測ブロックを作成し得る。たとえば、動き補償ユニット72は、現在の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

10

20

30

40

50

複数のための構築情報と、現在のビデオスライスの各々のインター符号化されたビデオブロックの動きベクトルと、スライスの各々のインターコーディングされたビデオブロックのインター予測ステータスと、現在のビデオスライスの中のビデオブロックを復号するための他の情報とを決定するために、受信されたシンタックス要素のいくつかを使用する。

【0144】

PUの動きベクトルがサブピクセル精度を有するとき、動き補償ユニット72は、1つまたは複数の補間フィルタを参照ピクチャのサンプルに適用して、PUの予測ブロックを生成し得る。言い換えると、動き補償ユニット72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72は、ビデオブロックの符号化の間にビデオエンコーダ20が使用したのと同じ補間フィルタを使用して、参照ブロックのサブ整数ピクセルに対する補間された値を計算し得る。したがって、いくつかの例では、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、補間フィルタを使用して、予測ブロックを作成し得る。

10

【0145】

逆量子化ユニット76は、ビットストリームにおいて提供されエントロピー復号ユニット70によって復号される量子化された変換係数を逆量子化する(inverse quantize)、すなわち逆量子化する(de-quantize)。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に、量子化パラメータ QP_Y の使用を含み得る。ビデオデコーダ30は、ビデオスライスの中の各ビデオブロックに対して量子化パラメータ QP_Y を計算し得る。

20

【0146】

逆変換ユニット78は、逆量子化された変換係数ブロックを受信し得る。現在のブロックに対して変換がスキップされる場合、逆変換ユニット78は逆量子化された残差ブロックを受信し得る。逆変換ユニット78は、逆変換を使用して受信されたブロックを変換し得る。いくつかの例では、ピクセル領域における残差ブロック(たとえば、変換ブロック)を作成するための、変換係数への逆変換(たとえば、逆DCT、逆整数変換、または概念的に類似の逆変換プロセス)。逆変換ユニット78は、「再構築された残差信号」と呼ばれる信号を出力し得る。

【0147】

さらに、逆変換ユニット78は、逆ACT変換または逆CCP変換を残差データに適用し得る。さらに、本開示の1つまたは複数の技法によれば、逆変換ユニット78は、変換領域からサンプル領域への変換、逆ACT変換、および/または逆CCP変換により生じる残差データのビット深度を低減するために、切り取り動作を残差データに適用し得る。

30

【0148】

ビデオデコーダ30はまた、シンタックス要素または他の情報に基づいて、現在のブロックがイントラ予測されると決定し得る。現在のビデオブロックがイントラ予測される場合、イントラ予測ユニット74は、現在のブロックを復号し得る。イントラ予測ユニット74は、現在のブロックと同じピクチャから近隣予測ブロックを決定し得る。イントラ予測ユニット74は、予測ブロックに基づいて、変換係数ブロックおよび/または残差ブロックを生成し得る。

40

【0149】

動き補償ユニット72またはイントラ予測ユニット74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のビデオブロックの変換係数ブロックおよび/または残差ブロックを生成した後、ビデオデコーダ30は、逆変換ユニット78からの残差ブロックを、動き補償ユニット72によって生成された対応する予測ブロックと合成することによって、復号された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つまたは複数の構成要素を表す。所望される場合、ブロックネスアーティファクトを除去するために、復号されたブロックをフィルタリングするように、デブロッキングフィルタも適用され得る。(コーディンググループ内またはコーディンググループ後のいずれかの)他のループフィルタも、ピクセル遷移を平滑化するために、または別様にビデオ品質を改善するために使用さ

50

れ得る。復号ピクチャバッファ82は、ビデオデコーダ30が後続の動き補償のために使用し得る所与のフレームまたはピクチャに、復号されたビデオブロックを記憶する。復号ピクチャバッファ82はまた、後で図1のディスプレイデバイス32などのディスプレイデバイス上に提示するために、復号されたビデオを記憶し得る。

【0150】

動き補償ユニット72および/またはイントラ予測ユニット74は、ルーマ残差ブロック、スケーリング係数、および予測されるクロマ残差サンプルに基づいて、クロマ残差サンプルのブロックを決定し得る。クロマ残差サンプルのブロックにより、加算器80は、クロマ残差サンプルおよびルーマ残差サンプルを、予測ブロックのそれぞれのクロマサンプルおよびルーマサンプルと加算して、現在のブロックを復号する(たとえば、現在のブロックを再構築する)ことができる。ビデオデコーダ30が再構築されたビデオを生成すると、ビデオデコーダ30は、いくつかの例では、(たとえば、表示または記憶のために)復号されたビデオとして再構築されたビデオブロックを出力し得る。

10

【0151】

上で説明されたように、インター予測の間、動き補償ユニット72は、ビデオデコーダ30が復号されている現在のブロックの予測ビデオブロックを形成するために使用し得る、1つまたは複数の参照ピクチャを決定し得る。動き補償ユニット72は、コーディングされたビデオビットストリームのシンタックス要素に基づいて、参照ピクチャが長期参照ピクチャであるか短期参照ピクチャであるかを判定することができ、このシンタックス要素は、参照ピクチャが長期参照としてマークされるか短期参照としてマークされるかを示す。動き補償ユニット72は、参照ピクチャが参照のために使用されないものとしてマークされるまで、復号ピクチャバッファ82に参照ピクチャを記憶し得る。

20

【0152】

図21は、本開示の技法による、モード確認順序を適応的に決定するためのプロセスを示すフローチャートである。図21の技法は、本開示において説明されるようなビデオエンコーダ、たとえばビデオエンコーダ20によって実行され得ることを理解されたい。図21のプロセスでは、ビデオエンコーダ20は、メトリックが条件を満たすかどうかを判定するように構成され得る(400)。ビデオエンコーダ20は、統計に基づいてメトリックが条件を満たすかどうかを判定することができ、この統計は、第1のビデオ符号化モード確認順序および第2のビデオ符号化モード確認順序と関連付けられる。

30

【0153】

メトリックが条件を満たすと判定したことに応答して、ビデオエンコーダ20は、ビデオデータの第1のブロックを符号化するために、第1の符号化モード確認順序を選択し得る(402)。条件を満たされないと判定したことに応答して、ビデオエンコーダ20は、ビデオデータの第1のブロックを符号化するために、第1の符号化モード確認順序と異なる第2の符号化モード確認順序を選択し得る(404)。モード選択ユニット40は、選択された第1または第2の符号化モード確認順序に基づいて、統計を更新し得る(406)。統計の例は、特定の符号化モード確認順序が選択される頻度、モード確認順序と関連付けられる平均のRDコスト、またはこれらの組合せを含み得る。モード選択ユニット40およびエントロピー符号化ユニット56は、更新された統計に基づいて、かつ第1または第2のモード確認順序を使用して、ビデオデータの第2のブロックを符号化し得る(408)。

40

【0154】

いくつかの例では、統計は、ビデオデータの以前に符号化されたブロックから決定された統計を備える。第1のビデオ符号化モード確認順序は第1の予測ユニットモード確認順序を備え、第2のビデオ符号化モード確認順序は第2の予測ユニットモード確認順序を備え、第1のビデオ予測ユニットモード確認順序は第2の予測ユニットモード確認順序と異なる。

【0155】

いくつかの例では、第1の予測ユニットモード確認順序は、モード選択ユニット40によって、第1のブロックを符号化するために高度動きベクトル予測(AMVP)を使用することについてのレートひずみ(RD)コストを決定する前に、第1のブロックを符号化するためにス

50

キップモードを使用することについてのRDコストを決定することを備え得る。第2の予測ユニットモード確認順序は、モード選択ユニット40によって第1のブロックを符号化するためにスキップモードを使用することについてのRDコストを条件的に決定する前に、第1のブロックを符号化するためにAMVPを使用することについてのRDコストを決定することを備え得る。

【0156】

いくつかの例では、この条件はピクチャの時間レイヤを備え、時間レイヤはビデオデータの最初のブロックと関連付けられる。第1の符号化モード確認順序は、第1の予測ユニット(PU)区分サイズ確認順序を備え得る。第2のビデオ符号化モード確認順序は第2のPU区分サイズ確認順序を備え、第1のPU区分サイズ確認順序は第2のPU区分サイズ確認順序と異なる。

10

【0157】

いくつかの例では、第1のPU区分サイズ確認順序は、モード選択ユニット40によって、第1のブロックを符号化するために $N \times 2N$ のPUサイズを使用することについてのレートひずみ(RD)コストを条件的に決定する前に、第1のブロックを符号化するために $2N \times N$ のPUサイズを使用することについてのRDコストを決定することを備え、 N は整数のサンプル数である。第2のPU区分サイズ確認順序は、モード選択ユニット40によって、第1のブロックを符号化するために $2N \times N$ のPUサイズを使用することについてのRDコストを条件的に決定する前に、第1のブロックを符号化するために $N \times 2N$ のPUサイズを使用することについてのRDコストを決定することを備える。

20

【0158】

いくつかの例では、この条件は、第1のブロック内への予測誤差の集中を備える。第1の符号化モード確認順序は第1の変換ユニット(TU)区分サイズ確認順序を備えることがあり、第2のビデオ符号化モード確認順序は第2のTU区分サイズ確認順序を備え、第1のTU区分サイズ確認順序は第2のTU区分サイズ確認順序と異なる。

【0159】

第1のTU区分サイズ確認順序は、モード選択ユニット40によって、第1のブロックを符号化するために 4×4 のTUサイズを使用することについてのレートひずみ(RD)コストを条件的に決定する前に、第1のブロックを符号化するために 8×8 のTUサイズを使用することについてのRDコストを決定することを備えることがあり、 N は整数のサンプル数である。第2のTU区分サイズ確認順序は、第1のブロックを符号化するために 8×8 のTUサイズを使用することについてのRDコストを条件的に決定する前に、第1のブロックを符号化するために 4×4 のTUサイズを使用することについてのRDコストを決定することを備え得る。

30

【0160】

条件が第1のブロック内への予測誤差の集中を備えるいくつかの例では、第1の符号化モード確認順序は第1の符号化ユニット(CU)区分サイズ確認順序を備えることがあり、第2のビデオ符号化モード確認順序は第2のCU区分サイズ確認順序を備えることがあり、第1のCU区分サイズ確認順序は第2のCU区分サイズ確認順序と異なることがある。

【0161】

様々な例において、CU区分サイズ確認順序は、第1のブロックを符号化するために $N \times N$ のCUサイズを使用することについてのレートひずみ(RD)コストを条件的に決定する前に、第1のブロックを符号化するために $2N \times 2N$ のCUサイズを使用することについてのRDコストを決定することを備えることがあり、 N は整数のサンプル数であり、第2のCU区分サイズ確認順序は、第1のブロックを符号化するために $2N \times 2N$ のCUサイズを使用することについてのRDコストを条件的に決定する前に、第1のブロックを符号化するために $N \times N$ のCUサイズを使用することについてのRDコストを決定することを備えることがある。様々な例において、この条件は1つまたは複数の閾値に基づき得る。

40

【0162】

図22は、本開示の技法による、レートひずみコストに基づいてブロックの区分方式を決定するためのプロセスを示すフローチャートである。図22の例では、区分ユニット48は、

50

完全RD確認方式または高速RD確認方式のうちの1つを選択し得る(420)。区分ユニット48は次いで、選択された完全RD確認方式または高速RD確認方式に基づいて、ビデオデータのブロックを符号化することと関連付けられるRDコストを決定し得る(422)。区分ユニット48は次いで、決定されたRDコストに基づいてブロックの区分方式を決定し(424)、決定されたRDコストに基づいて決定された区分方式を使用してブロックを符号化し得る(426)。

【0163】

いくつかの例では、完全RD確認方式または高速RD確認方式を選択することは、ブロックの複雑さに基づき得る。完全RD確認方式または高速RD確認方式を選択するために、モード選択ユニット40は、ブロックが閾値を超える複雑さを有する場合、完全RD確認方式を選択し得る。完全RD確認方式または高速RD確認方式を選択するために、モード選択ユニット40は、ブロックが閾値以内である複雑さを有する場合、高速RD確認方式を選択し得る。

10

【0164】

いくつかの例では、高速RD確認方式を選択したことに応答して、モード選択ユニット40は、ブロックの最大の区分サイズと関連付けられるひずみ値を決定することができ、ブロックの最大の区分サイズと関連付けられるひずみ値が閾値のひずみ値を超えると判定したことに応答して、モード選択ユニット40は、ブロックをサブ区分へと区分するかどうかを判定することができる。

【0165】

ブロックをサブ区分へと区分するかどうかを判定するために、モード選択ユニット40はさらに、サブ区分の各々と関連付けられるひずみ値を第2のひずみ閾値と比較し、サブ区分の少なくとも1つが第2のひずみ閾値より大きなひずみ値を有すると判定したことに応答して、ブロックをサブ区分へと区分するように構成され得る。第2のひずみ閾値は、いくつかの例ではサブ区分の各々と関連付けられるひずみ値のスケーリングされた平均に基づくことがあり、かつ/または、サブ区分の各々と関連付けられるひずみ値のばらつきに基づくことがある。

20

【0166】

ブロックをサブ区分へと区分するかどうかを判定するために、モード選択ユニット40はさらに、サブ区分の1つが他のサブ区分の少なくとも1つと比べて異なる動き情報と関連付けられるかどうかを判定するように構成されることがあり、サブ区分の1つが他のサブ区分の少なくとも1つと比べて異なる動き情報と関連付けられると判定したことに応答して、モード選択ユニット40はさらに、サブ区分の1つを追加のサブ区分へと区分するように構成されることがある。

30

【0167】

モード選択ユニット40は、ビデオデータのブロックを含む現在のスライスがiスライスである場合、完全RD確認を選択し得る。いくつかの例では、モード選択ユニット40は、ビデオエンコーダの高速符号化モードが有効にされているかどうかに基づいて、完全RD確認または高速RD確認を選択し得る。

【0168】

いくつかの例では、完全RD確認を選択することは、ブロックがSKIPモードを使用してコーディングされるときに完全RD確認を選択することを備える。

40

【0169】

例に応じて、本明細書で説明される技法のいずれかのいくつかの行為またはイベントは異なる順序で実行されてもよく、一緒に追加され、統合され、または省略されてもよい(たとえば、説明される行為またはイベントのすべてが技法の実践のために必要とは限らない)ことを認識されたい。さらに、いくつかの例では、行為またはイベントは、順次的ではなく、たとえばマルチスレッド処理、割り込み処理またはマルチプロセッサを通じて同時に実行されることがある。

【0170】

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、

50

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、またはコンピュータ可読媒体を介して送信されることがあり、かつハードウェアに基づく処理ユニットによって実行されることがある。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、または、たとえば、通信プロトコルに従って、ある場所から別の場所へのコンピュータプログラムの転送を可能にする任意の媒体を含む通信媒体を含み得る。このようにして、コンピュータ可読媒体は一般に、(1)非一時的である有形コンピュータ可読記憶媒体、または(2)信号もしくは搬送波などの通信媒体に相当し得る。データ記憶媒体は、本開示で説明する技法の実装のための命令、コードおよび/またはデータ構造を取り出すために1つもしくは複数のコンピュータまたは1つもしく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。

10

【0171】

限定ではなく例として、そのようなコンピュータ可読記憶媒体は、RAM、ROM、EEPROM、CD-ROMもしくは他の光ディスクストレージ、磁気ディスクストレージもしくは他の磁気記憶デバイス、フラッシュメモリ、または、命令もしくはデータ構造の形式の所望のプログラムコードを記憶するために使用され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含み得る。また、あらゆる接続が、コンピュータ可読媒体と適切に呼ばれる。たとえば、同軸ケーブル、光ファイバケーブル、ツイストペア、デジタル加入者回線(DSL)、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命令が送信される場合、同軸ケーブル、光ファイバケーブル、ツイストペア、DSL、または赤外線、無線、およびマイクロ波などのワイヤレス技術は、媒体の定義に含まれる。しかしながら、コンピュータ可読記憶媒体およびデータ記憶媒体は、接続、搬送波、信号、または他の一時的な媒体を含まず、代わりに非一時的な有形記憶媒体を指すことを理解されたい。ディスク(disk)およびディスク(disc)は、本明細書では、コンパクトディスク(disc)(CD)、レーザーディスク(登録商標)(disc)、光ディスク(disc)、デジタル多用途ディスク(disc)(DVD)、フロッピーディスク(disk)、およびBlu-ray(登録商標)ディスク(disc)を含み、ディスク(disk)は通常、データを磁氣的に再生し、ディスク(disc)は、レーザーを用いてデータを光学的に再生する。上記の組合せも、コンピュータ可読媒体の範囲内に含まれるものとする。

20

【0172】

命令は、1つまたは複数のデジタル信号プロセッサ(DSP)、汎用マイクロプロセッサ、特定用途向け集積回路(ASIC)、フィールドプログラマブル論理アレイ(FPGA)、または他の等価の集積論理回路もしくは個別論理回路などの、1つまたは複数のプロセッサによって実行され得る。したがって、本明細書では、「プロセッサ」という用語は、前述の構造、または本明細書で説明する技法の実装に適した任意の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、本明細書において説明される機能は、符号化および復号のために構成された専用のハードウェアモジュールおよび/もしくはソフトウェアモジュール内で与えられてよく、または複合コーデックに組み込まれてよい。また、技法は、1つまたは複数の回路または論理要素において完全に実施され得る。

30

【0173】

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ICのセット(たとえば、チップセット)を含む、様々なデバイスまたは装置において実施され得る。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側面を強調するために、様々な構成要素、モジュール、またはユニットが説明されているが、それらは、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、上で説明されたように、様々なユニットは、コーデックハードウェアユニットにおいて結合されることがあり、または適切なソフトウェアおよび/もしくはファームウェアとともに、上で説明されたような1つもしくは複数のプロセッサを含む相互動作可能なハードウェアユニットの集合によって提供されることがある。

40

【0174】

50

様々な例が説明された。これらの例および他の例は以下の特許請求の範囲内に入る。

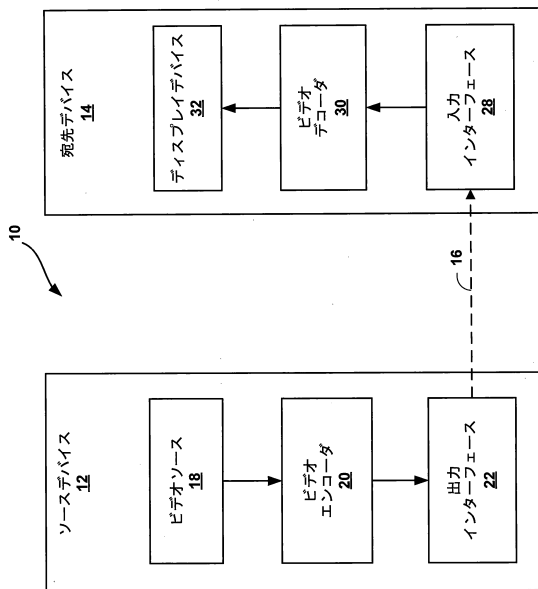
【符号の説明】

【 0 1 7 5 】

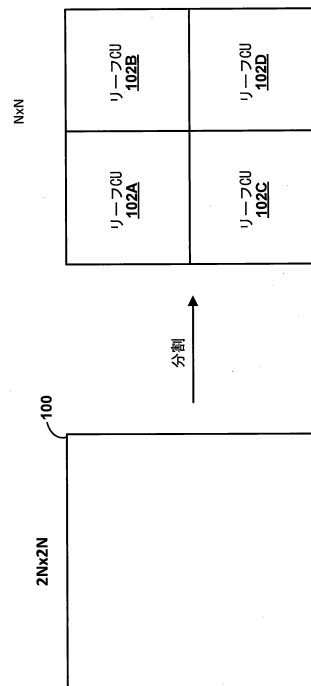
10	ビデオコーディングシステム	
12	ソースデバイス	
14	宛先デバイス	
16	チャンネル	
18	ビデオソース	
20	ビデオエンコーダ	
22	出力インターフェース	10
28	入力インターフェース	
30	ビデオデコーダ	
32	ディスプレイデバイス	
40	モード選択ユニット	
41	ビデオデータメモリ	
42	動き推定ユニット	
44	動き補償ユニット	
46	イントラ予測ユニット	
48	区分ユニット	
50	加算器	20
52	変換処理ユニット	
54	量子化ユニット	
56	エントロピー符号化ユニット	
58	逆量子化ユニット	
60	逆変換ユニット	
62	加算器	
64	復号ピクチャバッファ	
69	ビデオデータメモリ	
70	エントロピー復号ユニット	
72	動き補償ユニット	30
74	イントラ予測ユニット	
76	逆量子化ユニット	
78	逆変換ユニット	
80	加算器	
82	復号ピクチャバッファ	
100	CU	
102	リーフCU	
110	CTU	
111	CTU	
112	4分木構造	40
114	区分	
116	変換木	
120	プロセス	
140	モード確認プロセス	
160	モード確認プロセス	
180	プロセス	
200	プロセス	
220	プロセス	
240	誤差予測分布	
260	プロセス	50

- 280 誤差分布
- 300 プロセス
- 320 プロセス
- 340 グラフ
- 360 円
- 362 円

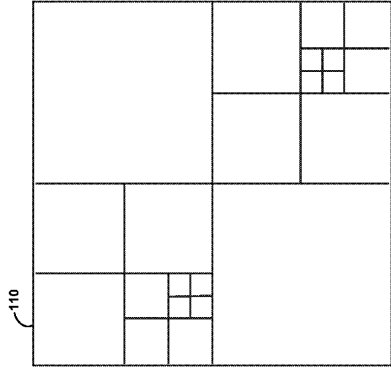
【図1】



【図2】



【 3 A 】



【 3 B 】

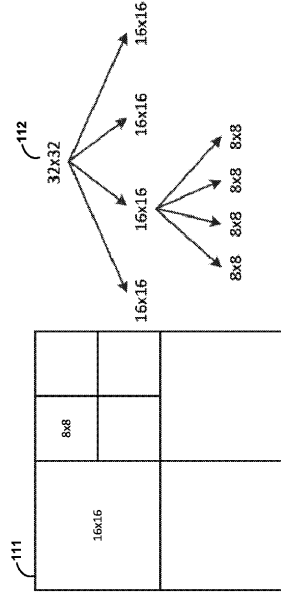


FIG. 3A

FIG. 3B

【 4 】

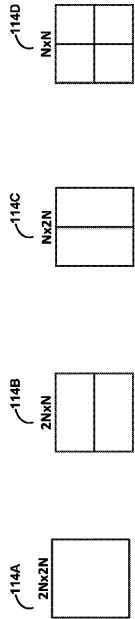


FIG. 4

【 5 】

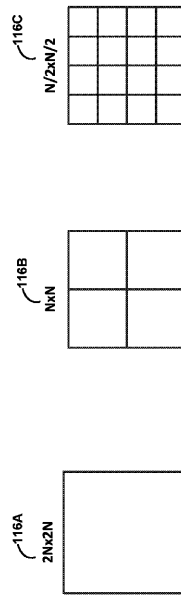
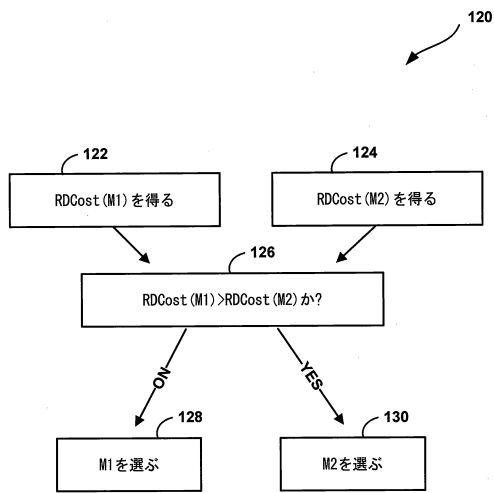
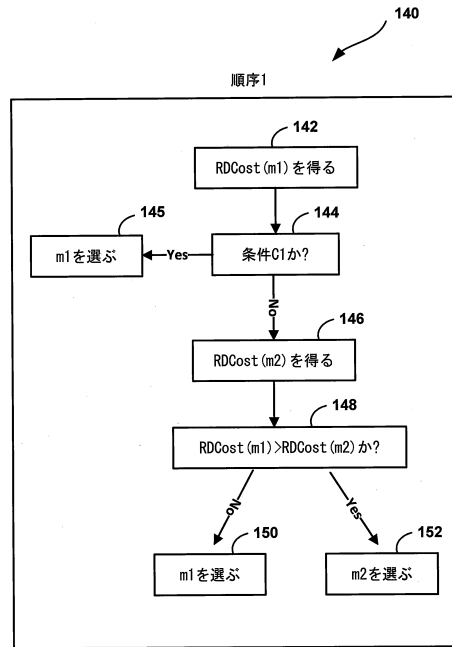


FIG. 5

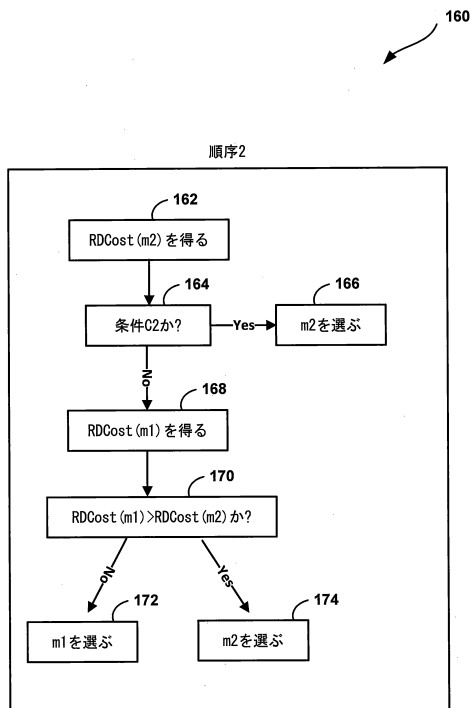
【図6】



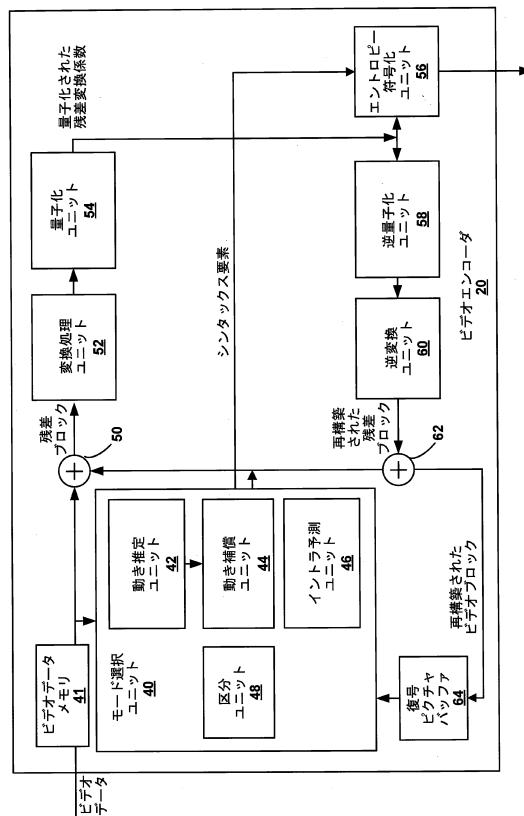
【図7A】



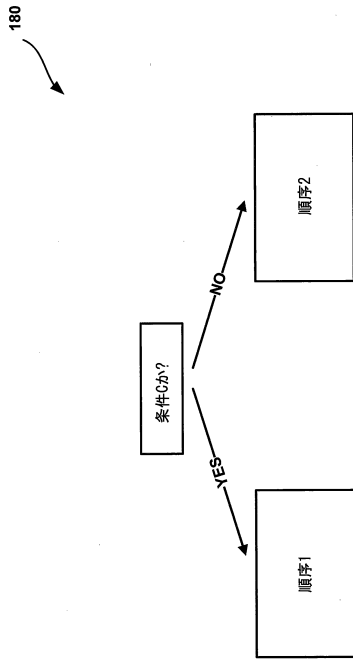
【図7B】



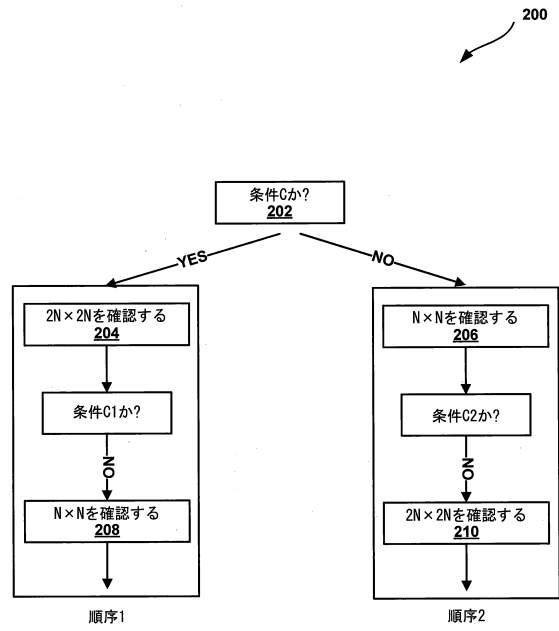
【図8】



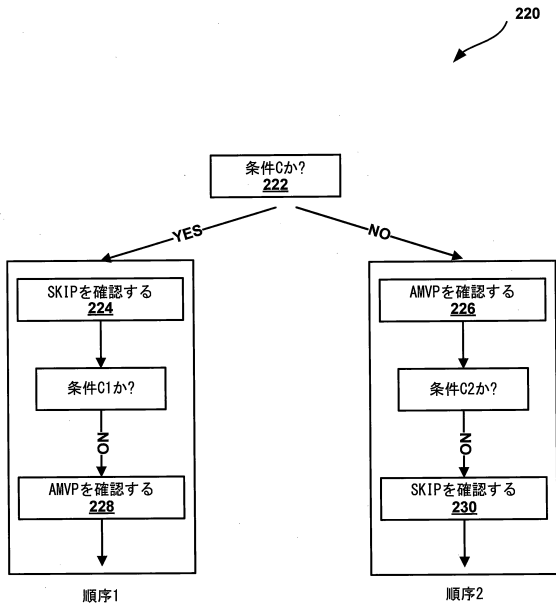
【図9】



【図10】



【図11】



【図12】

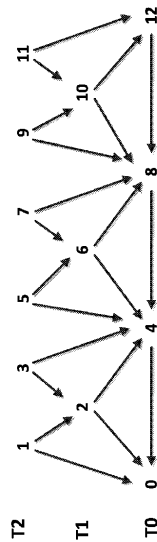


FIG. 12

【図13A】

-2	4	-7	3	6	-5	1	-7
5	7	-2	6	6	-4	-2	-4
4	1	-4	-5	0	-5	-7	-2
3	-7	-6	-5	1	0	0	-7
4	5	-6	-1	4	-4	-4	1
-32	16	-12	-24	-8	-4	8	-32
-24	12	-16	-20	4	8	8	-16
16	-24	-8	-12	-20	-16	-28	20

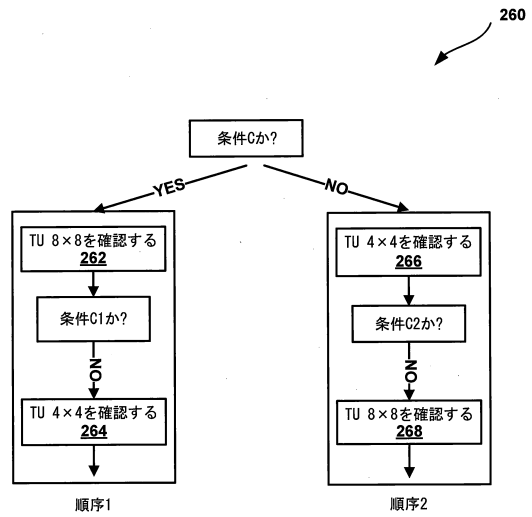
FIG. 13A

【図13B】

-2	4	-7	3	6	-20	4	-28
5	7	-2	6	6	-16	-8	-16
4	1	-4	-5	0	-20	-28	-8
3	-7	-6	-5	1	0	0	-28
4	5	-6	-1	4	-16	-16	4
-8	4	-3	-6	-2	-4	8	-32
-6	3	-4	-5	1	8	8	-16
4	-6	-2	-3	-5	-16	-28	20

FIG. 13B

【図14】



【図15A】

-2	4	-7	3	6	-5	1	-7
5	7	-2	6	6	-4	-2	-4
4	1	-4	-5	0	-5	-7	-2
3	-7	-6	-5	1	0	0	-7
4	5	-6	-1	4	-4	-4	1
-8	4	-3	-6	-2	-1	2	-8
-6	3	-4	-5	1	2	2	-4
4	-6	-2	-3	-5	-4	-7	5

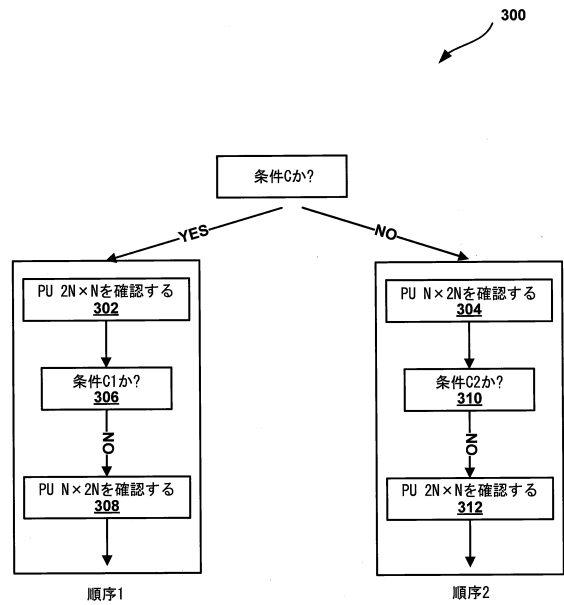
FIG. 15A

【図15B】

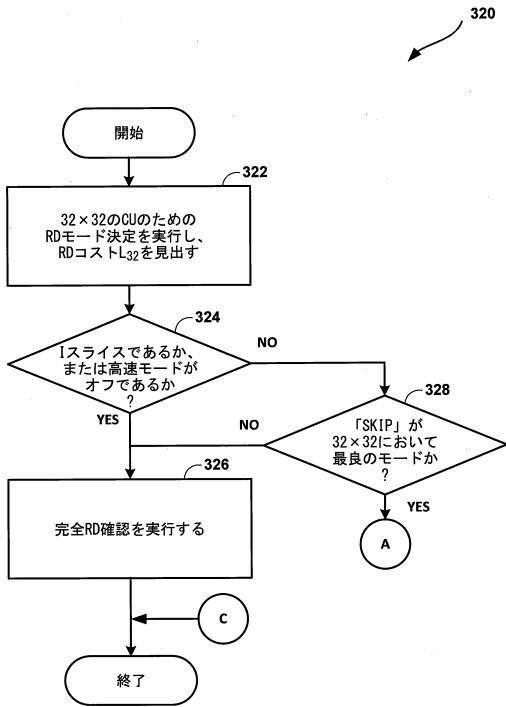
-2	4	-7	3	6	-5	1	-7
5	7	-2	6	6	-4	-2	-4
4	1	-4	-5	0	-5	-7	-2
3	-7	-6	-5	1	0	0	-7
4	5	-6	-1	4	-4	-4	1
-8	4	-3	-6	-2	-20	32	-8
-6	3	-4	-5	1	2	25	-4
4	-6	-2	-3	-5	-4	-22	-34

FIG. 15B

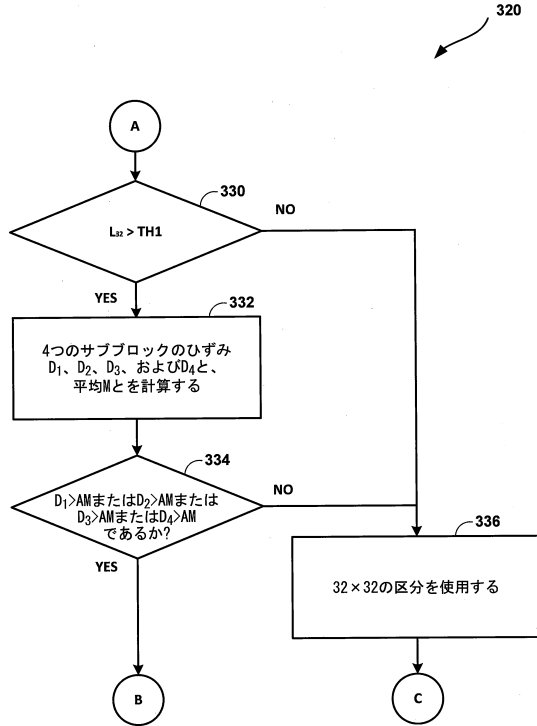
【図16】



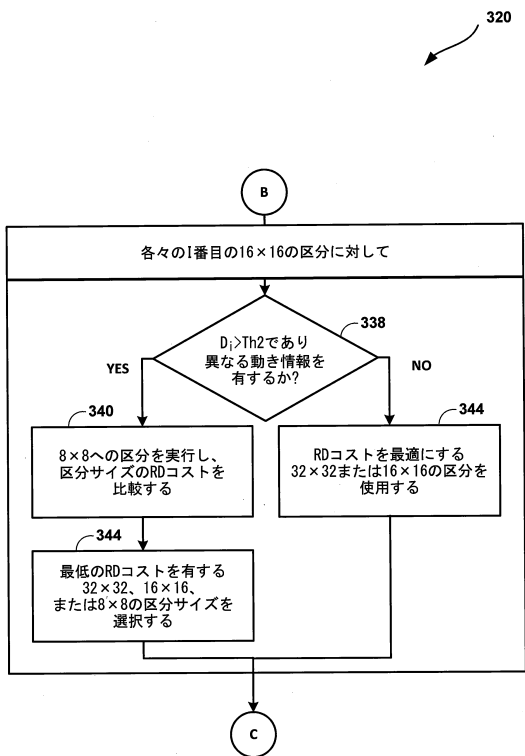
【図17A】



【図17B】



【図17C】



【図18】

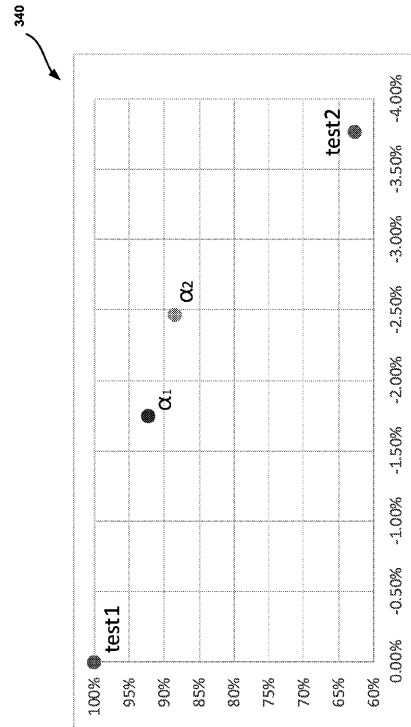


FIG. 18

【図 19】

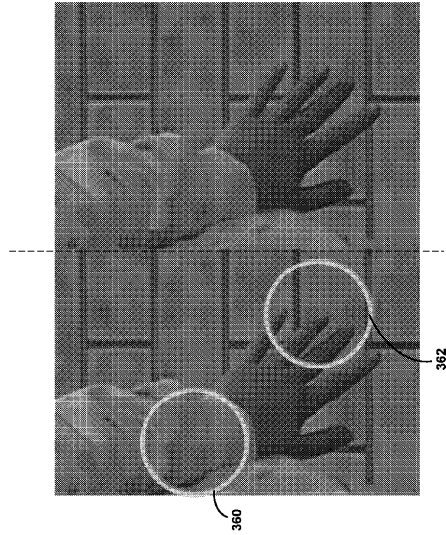
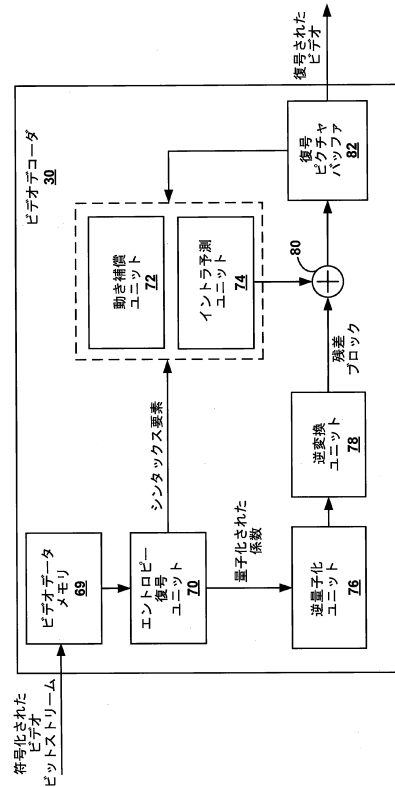
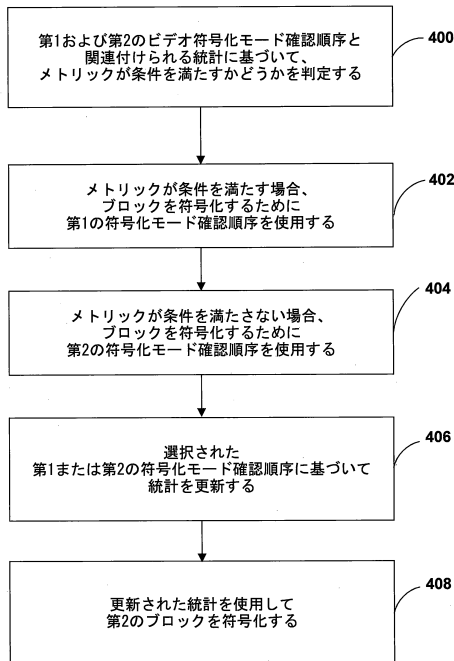


FIG. 19

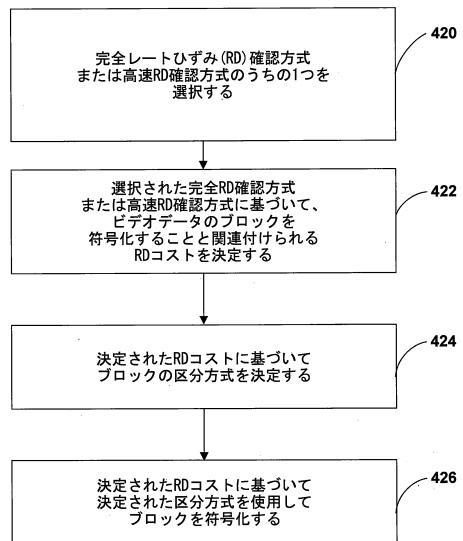
【図 20】



【図 21】



【図 22】



フロントページの続き

- (72)発明者 ウェイ - ジュン・チエン
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライブ・5 7 7 5
- (72)発明者 シアンリン・ワン
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライブ・5 7 7 5
- (72)発明者 ジェホン・チョン
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライブ・5 7 7 5
- (72)発明者 イン・スク・チョン
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライブ・5 7 7 5
- (72)発明者 マルタ・カルチェヴィッチ
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライブ・5 7 7 5
- (72)発明者 ウ - シク・キム
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライブ・5 7 7 5
- (72)発明者 シン・ジャオ
アメリカ合衆国・カリフォルニア・9 2 1 2 1 - 1 7 1 4・サン・ディエゴ・モアハウス・ドライブ・5 7 7 5

審査官 岩井 健二

- (56)参考文献 特表2008-519503(JP,A)
特開2008-283402(JP,A)
特開2007-166608(JP,A)
特表2006-513636(JP,A)
米国特許出願公開第2014/0241436(US,A1)
米国特許出願公開第2013/0195190(US,A1)
米国特許出願公開第2009/0110066(US,A1)
Jaeho Lee et al., A Fast CU Size Decision Algorithm for HEVC, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE, 2015年 3月 1日, VOL.25, NO.3, pp.411-421
Jiangpeng Hou et al., Fast CU Size Decision Based on Texture Complexity for HEVC Intra Coding, 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC), IEEE, 2013年12月, pp.1096-1099
Taiki Nishikori et al., A Fast CU Decision Using Image Variance in HEVC Intra Coding, 2013 IEEE Symposium on Industrial Electronics & Applications (ISIEA2013), IEEE, 2013年 9月, pp.52-56

(58)調査した分野(Int.Cl., DB名)

H04N 19/00 - 19/98