

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号
特表2018-535609
(P2018-535609A)

(43) 公表日 平成30年11月29日(2018.11.29)

(51) Int.Cl.
H04N 19/60 (2014.01)

F I
H04N 19/60 200

テーマコード (参考)
5C159

審査請求 未請求 予備審査請求 未請求 (全 119 頁)

(21) 出願番号	特願2018-526698 (P2018-526698)	(71) 出願人	507364838 クアルコム、インコーポレイテッド アメリカ合衆国 カリフォルニア 921 21 サン ディエゴ モアハウス ドラ イブ 5775
(86) (22) 出願日	平成28年11月23日 (2016.11.23)	(74) 代理人	100108453 弁理士 村山 靖彦
(85) 翻訳文提出日	平成30年5月22日 (2018.5.22)	(74) 代理人	100163522 弁理士 黒田 晋平
(86) 国際出願番号	PCT/US2016/063672	(72) 発明者	リ・ジャン アメリカ合衆国・カリフォルニア・921 21-1714・サン・ディエゴ・モアハ ウス・ドライブ・5775
(87) 国際公開番号	W02017/091770		
(87) 国際公開日	平成29年6月1日 (2017.6.1)		
(31) 優先権主張番号	62/260,103		
(32) 優先日	平成27年11月25日 (2015.11.25)		
(33) 優先権主張国	米国 (US)		
(31) 優先権主張番号	62/310,271		
(32) 優先日	平成28年3月18日 (2016.3.18)		
(33) 優先権主張国	米国 (US)		
(31) 優先権主張番号	15/359,568		
(32) 優先日	平成28年11月22日 (2016.11.22)		
(33) 優先権主張国	米国 (US)		

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非正方形変換単位のための変換係数の修正

(57) 【要約】

ビデオデータを符号化するプロセスの一部として、ビデオエンコーダは、変換を非正方形残差ブロックに適用して変換係数のブロックを生成する。さらに、ビデオエンコーダは、変換係数のブロックの各々のそれぞれの変換係数が、

$\sqrt{2}$

という係数または

$\sqrt{2}$

の近似と乗じられたそれぞれの逆量子化された変換係数に基づくように変換係数を修正する。ビデオデータを復号するプロセスの一部として、ビデオデコーダは、ビデオデコーダによって、逆量子化プロセスを非正方形変換ブロックの変換係数に適用する。ビデオデコーダは、逆量子化された変換係数のブロックの各々のそれぞれの逆量子化された変換係数が、

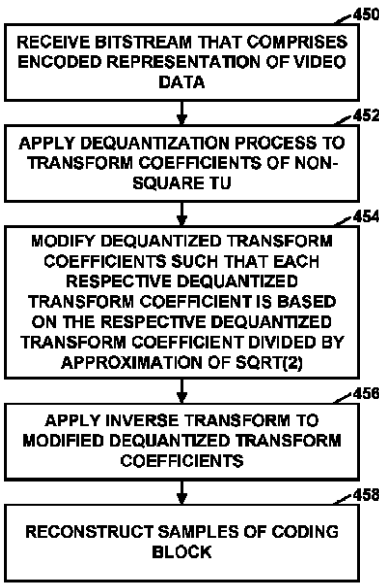


FIG. 31

【特許請求の範囲】

【請求項 1】

符号化されたビデオデータを復号する方法であって、
 ビデオデコードによって、前記ビデオデータの符号化された表現を備えるビットストリームを受信するステップと、
 前記ビデオデコードによって、逆量子化プロセスを前記ビデオデータのコーディング単位(CU)の非正方形変換単位(TU)の変換係数に適用するステップと、
 前記逆量子化プロセスを前記変換係数に適用した後で、前記ビデオデコードによって、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 1】

10

$$\sqrt{2}$$

の近似で割られた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正するステップと、

前記ビデオデコードによって、逆変換を前記修正された逆量子化された変換係数に適用して残差ブロックを再構築するステップと、

前記ビデオデコードによって、予測ブロックのサンプルを前記CUの前記TUの前記残差ブロックの対応するサンプルに加算することによってコーディングブロックのサンプルを再構築するステップとを備える、方法。

20

【請求項 2】

前記非正方形TUのサイズが $K \times L$ であり、前記逆変換を前記修正された逆量子化された変換係数に適用するステップが、前記ビデオデコードによって、サイズが $N \times N$ である変換を前記修正された逆量子化された変換係数に適用するステップを備え、 $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ であり、 $((\log_2(K) + \log_2(L)))$ が奇数である、請求項1に記載の方法。

【請求項 3】

前記非正方形TUのサイズが $K \times L$ であり、前記逆量子化された変換係数を修正するステップが、 $((\log_2(K) + \log_2(L)))$ が奇数であることに基づいて、前記ビデオデコードによって、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

30

【数 2】

$$\sqrt{2}$$

の前記近似で割られた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正するステップを備え、

$((\log_2(K) + \log_2(L)))$ が偶数であるとき、前記逆量子化された変換係数が、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 3】

40

$$\sqrt{2}$$

の前記近似で割られた前記それぞれの逆量子化された変換係数に基づくように修正されない、請求項1に記載の方法。

【請求項 4】

ビデオデータを符号化する方法であって、

ビデオエンコードによって、前記ビデオデータを受信するステップと、

前記ビデオエンコードによって、コーディング単位(CU)の非正方形変換単位(TU)の残差ブロックを、前記残差ブロックの各残差サンプルが前記CUのコーディングブロックの対応するサンプルと前記CUの予測単位(PU)の予測ブロックとの差を示すように生成するステッ

50

ブと、

前記ビデオエンコーダによって、変換を前記残差ブロックに適用して変換係数のブロックを生成するステップと、

前記ビデオエンコーダによって、変換係数の前記ブロックの各々のそれぞれの変換係数が

【数 4】

$$\sqrt{2}$$

の近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように、前記変換係数を修正するステップと、

10

前記変換係数を修正した後で、前記ビデオエンコーダによって、量子化プロセスを前記CUの前記非正方形TUの前記修正された変換係数に適用するステップと、

前記ビデオエンコーダによって、前記ビデオデータの符号化された表現を備えるビットストリームに、前記量子化された変換係数に基づくデータを含めるステップとを備える、方法。

【請求項 5】

前記非正方形TUのサイズが $K \times L$ であり、前記変換を前記残差ブロックに適用するステップが、前記ビデオエンコーダによって、サイズが $N \times N$ である変換を前記残差ブロックに適用するステップを備え、 $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ であり、 $((\log_2(K) + \log_2(L))$ が奇数である、請求項4に記載の方法。

20

【請求項 6】

前記非正方形TUのサイズが $K \times L$ であり、前記逆量子化された変換係数を修正するステップが、 $((\log_2(K) + \log_2(L))$ が奇数であることに基づいて、前記ビデオエンコーダによって、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 5】

$$\sqrt{2}$$

の前記近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正するステップを備え、

30

$((\log_2(K) + \log_2(L))$ が偶数であるとき、前記逆量子化された変換係数が、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 6】

$$\sqrt{2}$$

の前記近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように修正されない、請求項4に記載の方法。

【請求項 7】

40

ビデオデータを復号するための装置であって、

前記ビデオデータを記憶するように構成されるメモリと、

回路、ソフトウェア、またはこれらの組合せを使用して実装されるビデオデコーダとを備え、前記ビデオデコーダが、

逆量子化プロセスを前記ビデオデータのコーディング単位(CU)の非正方形変換単位(TU)の変換係数に適用し、

前記逆量子化プロセスを前記変換係数に適用した後で、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 7】

$$\sqrt{2}$$

の近似で割られた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正し、

逆変換を前記修正された逆量子化された変換係数に適用して残差ブロックを再構築し

、
予測ブロックのサンプルを前記CUの前記TUの前記残差ブロックの対応するサンプルに加算することによってコーディングブロックのサンプルを再構築する

10

ように構成される、装置。

【請求項 8】

前記非正方形TUのサイズが $K \times L$ であり、前記ビデオデコーダが、前記逆変換を前記修正された逆量子化された変換係数に適用することの一部として、サイズが $N \times N$ である変換を前記修正された逆量子化された変換係数に適用するように構成され、 $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ であり、 $((\log_2(K) + \log_2(L)))$ が奇数である、請求項7に記載の装置。

【請求項 9】

前記非正方形TUのサイズが $K \times L$ であり、前記ビデオデコーダが、 $((\log_2(K) + \log_2(L)))$ が奇数であることに基づいて、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

20

【数 8】

$$\sqrt{2}$$

の前記近似で割られた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正するように構成され、

$((\log_2(K) + \log_2(L)))$ が偶数であるとき、前記ビデオデコーダが、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 9】

30

$$\sqrt{2}$$

の前記近似で割られた前記それぞれの逆量子化された変換係数に基づくように前記逆量子化された変換係数を修正しない、請求項7に記載の装置。

【請求項 10】

ビデオデータを符号化するための装置であって、

前記ビデオデータを記憶するように構成されるメモリと、

回路、ソフトウェア、またはこれらの組合せを使用して実装されるビデオエンコーダとを備え、前記ビデオエンコーダが、

40

コーディング単位(CU)の非正方形変換単位(TU)の残差ブロックを、前記残差ブロックの各残差サンプルが前記CUのコーディングブロックの対応するサンプルと前記CUの予測単位(PU)の予測ブロックとの差を示すように生成し、

変換を前記残差ブロックに適用して変換係数のブロックを生成し、

変換係数の前記ブロックの各々のそれぞれの変換係数が

【数 10】

$$\sqrt{2}$$

の近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように、前記変換係

50

数を修正し、

前記変換係数を修正した後で、量子化プロセスを前記CUの前記非正方形TUの前記修正された変換係数に適用し、

前記ビデオデータの符号化された表現を備えるビットストリームに、前記量子化された変換係数に基づくデータを含める

ように構成される、装置。

【請求項 1 1】

前記非正方形TUのサイズが $K \times L$ であり、前記ビデオエンコーダが、前記変換を前記残差ブロックに適用することの一部として、サイズが $N \times N$ である変換を前記残差ブロックに適用するように構成され、 $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ であり、 $((\log_2(K) + \log_2(L))$ が奇数である、請求項10に記載の装置。 10

【請求項 1 2】

前記非正方形TUのサイズが $K \times L$ であり、前記ビデオエンコーダが、 $((\log_2(K) + \log_2(L)))$ が奇数であることに基づいて、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 1 1】

$$\sqrt{2}$$

の前記近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正するように構成され、 20

$((\log_2(K) + \log_2(L)))$ が偶数であるとき、前記ビデオエンコーダが、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 1 2】

$$\sqrt{2}$$

の前記近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように前記逆量子化された変換係数を修正しない、請求項10に記載の装置。

【請求項 1 3】

ビデオデータを復号するための装置であって、

前記ビデオデータの符号化された表現を備えるビットストリームを受信するための手段と、

逆量子化プロセスを前記ビデオデータのコーディング単位(CU)の非正方形変換単位(TU)の変換係数に適用するための手段と、

前記逆量子化プロセスを前記変換係数に適用した後で、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 1 3】

$$\sqrt{2}$$

の近似で割られた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正するための手段と、

逆変換を前記修正された逆量子化された変換係数に適用して残差ブロックを再構築するための手段と、

予測ブロックのサンプルを前記CUの前記TUの前記残差ブロックの対応するサンプルに加算することによってコーディングブロックのサンプルを再構築するための手段とを備える、装置。

【請求項 1 4】

前記非正方形TUのサイズが $K \times L$ であり、前記逆変換を前記修正された逆量子化された変 50

換係数に適用するための前記手段が、サイズが $N \times N$ である変換を前記修正された逆量子化された変換係数に適用するための手段を備え、 $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) > 1) < 1)$ であり、 $((\log_2(K) + \log_2(L))$ が奇数である、請求項13に記載の装置。

【請求項15】

前記非正方形TUのサイズが $K \times L$ であり、前記逆量子化された変換係数を修正するための前記手段が、 $((\log_2(K) + \log_2(L))$ が奇数であることに基づいて、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数14】

$$\sqrt{2}$$

10

の前記近似で割られた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正するための手段を備え、

$((\log_2(K) + \log_2(L))$ が偶数であるとき、前記逆量子化された変換係数が、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数15】

$$\sqrt{2}$$

の前記近似で割られた前記それぞれの逆量子化された変換係数に基づくように修正されない、請求項13に記載の装置。

20

【請求項16】

ビデオデータを符号化するための装置であって、

前記ビデオデータを受信するための手段と、

コーディング単位(CU)の非正方形変換単位(TU)の残差ブロックを、前記残差ブロックの各残差サンプルが前記CUのコーディングブロックの対応するサンプルと前記CUの予測単位(PU)の予測ブロックとの差を示すように生成するための手段と、

変換を前記残差ブロックに適用して変換係数のブロックを生成するための手段と、

変換係数の前記ブロックの各々のそれぞれの変換係数が

【数16】

30

$$\sqrt{2}$$

の近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように、前記変換係数を修正するための手段と、

前記変換係数を修正した後で、量子化プロセスを前記CUの前記非正方形TUの前記修正された変換係数に適用するための手段と、

前記ビデオデータの符号化された表現を備えるビットストリームに、前記量子化された変換係数に基づくデータを含めるための手段とを備える、装置。

【請求項17】

40

前記非正方形TUのサイズが $K \times L$ であり、前記変換を前記残差ブロックに適用するための前記手段が、サイズが $N \times N$ である変換を前記残差ブロックに適用するための手段を備え、 $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) > 1) < 1)$ であり、 $((\log_2(K) + \log_2(L))$ が奇数である、請求項16に記載の装置。

【請求項18】

前記非正方形TUのサイズが $K \times L$ であり、前記逆量子化された変換係数を修正するための前記手段が、 $((\log_2(K) + \log_2(L))$ が奇数であることに基づいて、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 1 7】

$$\sqrt{2}$$

の前記近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正するための手段を備え、

(($\log_2(K) + \log_2(L)$) が偶数であるとき、前記逆量子化された変換係数が、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 1 8】

10

$$\sqrt{2}$$

の前記近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように修正されない、請求項16に記載の装置。

【請求項 1 9】

命令を記憶したコンピュータ可読媒体であって、前記命令が、実行されると、前記ビデオデータの符号化された表現を備えるビットストリームを受信し、逆量子化プロセスを前記ビデオデータのコーディング単位(CU)の非正方形変換単位(TU)の変換係数に適用し、

前記逆量子化プロセスを前記変換係数に適用した後で、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

20

【数 1 9】

$$\sqrt{2}$$

の近似で割られた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正し、

逆変換を前記修正された逆量子化された変換係数に適用して残差ブロックを再構築し、予測ブロックのサンプルを前記CUの前記TUの前記残差ブロックの対応するサンプルに加算することによってコーディングブロックのサンプルを再構築する

30

ようにビデオデータを復号するための装置を構成する、コンピュータ可読媒体。

【請求項 2 0】

前記非正方形TUのサイズが $K \times L$ であり、前記命令が、前記逆変換を前記修正された逆量子化された変換係数に適用することの一部として、サイズが $N \times N$ である変換を前記修正された逆量子化された変換係数に適用するように前記装置を構成し、 $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ であり、(($\log_2(K) + \log_2(L)$) が奇数である、請求項19に記載のコンピュータ可読媒体。

【請求項 2 1】

前記非正方形TUのサイズが $K \times L$ であり、前記命令が、実行されると、(($\log_2(K) + \log_2(L)$) が奇数であることに基づいて、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

40

【数 2 0】

$$\sqrt{2}$$

の前記近似で割られた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正するように前記装置を構成し、

前記命令が、実行されると、(($\log_2(K) + \log_2(L)$) が偶数であるとき、前記逆量子化された変換係数が、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

50

【数 2 1】

$$\sqrt{2}$$

の前記近似で割られた前記それぞれの逆量子化された変換係数に基づくように修正されないように、前記装置を構成する、請求項19に記載のコンピュータ可読媒体。

【請求項 2 2】

命令を記憶したコンピュータ可読媒体であって、前記命令が、実行されると、前記ビデオデータを受信し、

コーディング単位(CU)の非正方形変換単位(TU)の残差ブロックを、前記残差ブロックの各残差サンプルが前記CUのコーディングブロックの対応するサンプルと前記CUの予測単位(PU)の予測ブロックとの差を示すように生成し、

10

変換を前記残差ブロックに適用して変換係数のブロックを生成し、
変換係数の前記ブロックの各々のそれぞれの変換係数が

【数 2 2】

$$\sqrt{2}$$

の近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように、前記変換係数を修正し、

20

前記変換係数を修正した後で、量子化プロセスを前記CUの前記非正方形TUの前記修正された変換係数に適用し、

前記ビデオデータの符号化された表現を備えるビットストリームに、前記量子化された変換係数に基づくデータを含める

ように、ビデオデータを符号化するための装置を構成する、コンピュータ可読媒体。

【請求項 2 3】

前記非正方形TUのサイズが $K \times L$ であり、前記命令が、前記変換を前記残差ブロックに適用することの一部として、サイズが $N \times N$ である変換を前記残差ブロックに適用するように前記装置を構成し、 $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ であり、 $((\log_2(K) + \log_2(L))$ が奇数である、請求項22に記載のコンピュータ可読媒体。

30

【請求項 2 4】

前記非正方形TUのサイズが $K \times L$ であり、前記命令が、実行されると、 $((\log_2(K) + \log_2(L))$ が奇数であることに基づいて、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 2 3】

$$\sqrt{2}$$

の前記近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように、前記逆量子化された変換係数を修正するように前記装置を構成し、

40

前記命令が、実行されると、 $((\log_2(K) + \log_2(L))$ が偶数であるとき、前記逆量子化された変換係数が、前記逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【数 2 4】

$$\sqrt{2}$$

の前記近似と乗じられた前記それぞれの逆量子化された変換係数に基づくように修正されないように、前記装置を構成する、請求項22に記載のコンピュータ可読媒体。

【発明の詳細な説明】

50

【技術分野】

【0001】

本出願は、2015年11月25日に出願された米国仮特許出願第62/260,103号、および2016年3月18日に出願された米国仮特許出願第62/310,271号の利益を主張するものであり、これらの各々の内容全体が、参照により本明細書に組み込まれている。

【0002】

本開示は、ビデオ符号化およびビデオ復号に関する。

【背景技術】

【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、携帯電話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、Advanced Video Coding(AVC)、High Efficiency Video Coding(HEVC)規格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の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、空間的(ピクチャ内)予測および/または時間的(ピクチャ間)予測を含む。ブロックベースのビデオコーディングのために、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)は、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、コーディングユニット(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。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】

空間的予測または時間的予測は、コーディングされるべきブロックの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。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。さらなる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ることがある。

【先行技術文献】

【特許文献】

【0006】

【特許文献1】米国特許出願公開第2016/0219290 A1号

【非特許文献】

【0007】

【非特許文献1】H. Liu、「Local Illumination Compensation」、ITU-Telecommunications Standardization Sector、Study Group 16 Question 6、Video Coding Experts Group(VCEG)、第52回会合、2015年6月19~26日、ワルシャワ、ポーランド、文書VCEG-AZ06

【非特許文献2】Wang他、「High Efficiency Video Coding (HEVC) Defect Report」、Joint Collaborative Team in Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、第14回会合、ウィーン、オーストリア、2013年7月25日~8月2日、文書JCTVC-N1003_v1

【非特許文献3】Marpe他、「Transform Coding Using the Residual Quadtree (RQT)」、Fraunhofer Heinrich Hertz Institute

10

20

30

40

50

【非特許文献4】Jianle Chen他、「Further improvements to HMKTA-1.0」、文書：VCEG-AZ07_v2、第52回会合、2015年6月19～26日、ワルシャワ、ポーランド

【非特許文献5】Xiaoran Cao他、「CE6.b1 Report on Short Distance Intra Prediction Method」、文書：JCTVC-E0278、Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、第5回会合、ジュネーブ、2011年3月16～23日

【非特許文献6】Liu他、「Rectangular ($2N \times N$, $N \times 2N$) Intra Prediction」、文書：JCTVC-G135、Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、第7回会合、ジュネーブ、2011年11月21～30日

【非特許文献7】A. FordおよびA. Roberts、「Colour space conversions」、ウェストミンスター大学、ロンドン、Tech. Rep、1998年8月

【非特許文献8】Xiaoran Cao他、「Short distance intra coding scheme for HEVC」、2012 Picture Coding Symposium (PCS)、pp. 501～504、2012年5月7～9日、クラクフ、ポーランド

【非特許文献9】J. Chen他、「CE6.a.4: Chroma intra prediction by reconstructed luma samples」、Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、JCTVC-E266、第5回会合、ジュネーブ、2011年3月16～23日

【非特許文献10】Yi-Jen Chiu他、「Cross-channel techniques to improve intra chroma prediction」、Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、JCTVC-F502、第6回会合、トリノ、イタリア、2011年7月14～22日

【非特許文献11】Yuan他、「Non-Square Quadtree Transform Structure for HEVC」、2012 Picture Coding Symposium (PCS)、pp. 505～508、2012年5月7～9日、クラクフ、ポーランド

【非特許文献12】X. Zhao他、「Video coding with rate-distortion optimized transform」、IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 1, pp. 138～151、2012年1月

【非特許文献13】Liu他、「3D-CE1.h related: Illumination Compensation for Inter-View Prediction」、Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、第1回会合、ストックホルム、スウェーデン、2012年7月16～20日、文書JCT3V-A0086

【発明の概要】

【課題を解決するための手段】

【0008】

本開示は、イントラおよびインター予測区分、非正方形変換、非正方形ブロックのためのイントラおよびインターコーディングモード、ならびに関連するエントロピーコーディングに関する。本開示の技法は、HEVCの拡張または次世代のビデオコーディング規格などの、先進的なビデオコーデックの文脈において使用され得る。一例では、量子化プロセスが、非正方形変換単位とともに使用するために適合される。

【0009】

一例では、本開示は符号化されたビデオデータを復号する方法を説明し、この方法は、ビデオデコードによって、ビデオデータの符号化された表現を備えるビットストリームを受信するステップと、ビデオデコードによって、逆量子化プロセスをビデオデータのコーディング単位(CU)の非正方形変換単位(TU)の変換係数に適用するステップと、逆量子化プロセスを変換係数に適用した後で、ビデオデコードによって、逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【0010】

10

20

30

40

【数 1】

$$\sqrt{2}$$

【0011】

の近似で割られたそれぞれの逆量子化された変換係数に基づくように、逆量子化された変換係数を修正するステップと、ビデオデコードによって、逆変換を修正された逆量子化された変換係数に適用して残差ブロックを再構築するステップと、ビデオデコードによって、予測ブロックのサンプルをCUのTUの残差ブロックの対応するサンプルに加算することによってコーディングブロックのサンプルを再構築するステップとを備える。

【0012】

別の例では、本開示はビデオデータを符号化する方法を説明し、この方法は、ビデオエンコードによって、ビデオデータを受信するステップと、ビデオエンコードによって、コーディング単位(CU)の非正方形変換単位(TU)の残差ブロックを、残差ブロックの各残差サンプルがCUのコーディングブロックの対応するサンプルとCUの予測単位(PU)の予測ブロックとの差を示すように生成するステップと、ビデオエンコードによって、変換を残差ブロックに適用して変換係数のブロックを生成するステップと、ビデオエンコードによって、変換係数のブロックの各々のそれぞれの逆変換係数が

【0013】

【数 2】

$$\sqrt{2}$$

【0014】

の近似と乗じられたそれぞれの逆量子化された変換係数に基づくように変換係数を修正するステップと、変換係数を修正した後で、ビデオエンコードによって、量子化プロセスをCUの非正方形TUの修正された変換係数に適用するステップと、ビデオエンコードによって、ビデオデータの符号化された表現を備えるビットストリームに、量子化された変換係数に基づくデータを含めるステップとを備える。

【0015】

別の例では、本開示はビデオデータを復号するための装置を説明し、この装置は、ビデオデータを記憶するように構成されるメモリと、回路、ソフトウェア、またはこれらの組合せを使用して実装されるビデオデコードとを備え、このビデオデコードは、逆量子化プロセスをビデオデータのコーディング単位(CU)の非正方形変換単位(TU)の変換係数に適用し、逆量子化プロセスを変換係数に適用した後で、逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

【0016】

【数 3】

$$\sqrt{2}$$

【0017】

の近似で割られたそれぞれの逆量子化された変換係数に基づくように、逆量子化された変換係数を修正し、逆変換を修正された逆量子化された変換係数に適用して残差ブロックを再構築し、予測ブロックのサンプルをCUのTUの残差ブロックの対応するサンプルに加算することによってコーディングブロックのサンプルを再構築するように構成される。

【0018】

10

20

30

40

50

別の例では、本開示はビデオデータを符号化するための装置を説明し、この装置は、ビデオデータを記憶するように構成されるメモリと、回路、ソフトウェア、またはこれらの組合せを使用して実装されるビデオエンコーダとを備え、このビデオエンコーダは、コーディング単位(CU)の非正方形変換単位(TU)の残差ブロックを、残差ブロックの各残差サンプルがCUのコーディングブロックの対応するサンプルとCUの予測単位(PU)の予測ブロックとの差を示すように生成し、変換を残差ブロックに適用して変換係数のブロックを生成し、変換係数のブロックの各々のそれぞれの変換係数が

【 0 0 1 9 】

【 数 4 】

10

$$\sqrt{2}$$

【 0 0 2 0 】

の近似と乗じられたそれぞれの逆量子化された変換係数に基づくように変換係数を修正し、変換係数を修正した後で、量子化プロセスをCUの非正方形TUの修正された変換係数に適用し、ビデオデータの符号化された表現を備えるビットストリームに、量子化された変換係数に基づくデータを含めるように構成される。

【 0 0 2 1 】

別の例では、本開示はビデオデータを復号するための装置を説明し、この装置は、ビデオデータの符号化された表現を備えるビットストリームを受信するための手段と、逆量子化プロセスをビデオデータのコーディング単位(CU)の非正方形変換単位(TU)の変換係数に適用するための手段と、逆量子化プロセスを変換係数に適用した後で、逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

20

【 0 0 2 2 】

【 数 5 】

$$\sqrt{2}$$

30

【 0 0 2 3 】

の近似で割られたそれぞれの逆量子化された変換係数に基づくように、逆量子化された変換係数を修正するための手段と、逆変換を修正された逆量子化された変換係数に適用して残差ブロックを再構築するための手段と、予測ブロックのサンプルをCUのTUの残差ブロックの対応するサンプルに加算することによってコーディングブロックのサンプルを再構築するための手段とを備える。

【 0 0 2 4 】

別の例では、本開示はビデオデータを符号化するための装置を説明し、この装置は、ビデオデータを受信するための手段と、コーディング単位(CU)の非正方形変換単位(TU)の残差ブロックを、残差ブロックの各残差サンプルがCUのコーディングブロックの対応するサンプルとCUの予測単位(PU)の予測ブロックとの差を示すように生成するための手段と、変換を残差ブロックに適用して変換係数のブロックを生成するための手段と、変換係数のブロックの各々のそれぞれの変換係数が

40

【 0 0 2 5 】

【 数 6 】

$$\sqrt{2}$$

【 0 0 2 6 】

50

の近似と乗じられたそれぞれの逆量子化された変換係数に基づくように変換係数を修正するための手段と、変換係数を修正した後で、量子化プロセスをCUの非正方形TUの修正された変換係数に適用するための手段と、ビデオデータの符号化された表現を備えるビットストリームに、量子化された変換係数に基づくデータを含めるための手段とを備える。

【0027】

別の例では、本開示は命令を記憶したコンピュータ可読媒体を説明し、この命令は、実行されると、ビデオデータの符号化された表現を備えるビットストリームを受信し、逆量子化プロセスをビデオデータのコーディング単位(CU)の非正方形変換単位(TU)の変換係数に適用し、逆量子化プロセスを変換係数に適用した後で、逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が

10

【0028】

【数7】

$$\sqrt{2}$$

【0029】

の近似で割られたそれぞれの逆量子化された変換係数に基づくように、逆量子化された変換係数を修正し、逆変換を修正された逆量子化された変換係数に適用して残差ブロックを再構築し、予測ブロックのサンプルをCUのTUの残差ブロックの対応するサンプルに加算することによってコーディングブロックのサンプルを再構築するように、ビデオデータを復号するための装置を構成する。

20

【0030】

別の例では、本開示は命令を記憶したコンピュータ可読媒体を説明し、この命令は、実行されると、ビデオデータを受信し、コーディング単位(CU)の非正方形変換単位(TU)の残差ブロックを、残差ブロックの各残差サンプルがCUのコーディングブロックの対応するサンプルとCUの予測単位(PU)の予測ブロックとの差を示すように生成し、変換を残差ブロックに適用して変換係数のブロックを生成し、変換係数のブロックの各々のそれぞれの逆変換係数が

【0031】

【数8】

$$\sqrt{2}$$

【0032】

の近似と乗じられたそれぞれの逆量子化された変換係数に基づくように変換係数を修正し、変換係数を修正した後で、量子化プロセスをCUの非正方形TUの修正された変換係数に適用し、ビデオデータの符号化された表現を備えるビットストリームに、量子化された変換係数に基づくデータを含めるように、ビデオデータを符号化するための装置を構成する。

40

【0033】

本開示の1つまたは複数の態様の詳細が、添付の図面および下記の説明において記載される。本開示において説明される技法の他の特徴、目的、および利点は、これらの説明および図面、ならびに特許請求の範囲から明らかになる。

【図面の簡単な説明】

【0034】

【図1】本開示の技法を実施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である。

【図2A】High Efficiency Video Coding(HEVC)における残差四分木に基づく例示的な変換方式を示す概念図である。

50

- 【図 2 B】図 2A のコーディング単位のための残差四分木を示す概念図である。
- 【図 3】HEVC における係数グループに基づく例示的な係数走査を示す概念図である。
- 【図 4】 16×16 ブロックに対するイントラ予測の例を示す概念図である。
- 【図 5】HEVC において定義される 35 個のイントラ予測モードの例を示す概念図である。
- 【図 6】HEVC において定義される平面モードを示す概念図である。
- 【図 7】HEVC において定義される例示的な角度モードの概念図である。
- 【図 8】HEVC におけるインター予測のためのコーディング単位を分割するための区分モードの概念図である。
- 【図 9】短距離イントラ予測 (SDIP) 単位区分の概念図である。
- 【図 10】走査され 8×8 行列へと並べ替えられる 16×4 係数行列の概念図である。 10
- 【図 11】64 個のイントラ予測モードの概念図である。
- 【図 12 A】イントラモード 34 に対する境界フィルタの概念図である。
- 【図 12 B】イントラモード 30 ~ 33 に対する境界フィルタの概念図である。
- 【図 13】線形モデル (LM) パラメータ および の導出のために使用されるサンプルの例示的な位置を示す概念図である。
- 【図 14】現在の予測単位 (PU) の再構築される輝度ブロックのサンプルをダウンサンプリングするための輝度場所および色差場所の例を示す概念図である。
- 【図 15】予測ブロックを生成するための輝度ブロックのサンプルをダウンサンプリングするための輝度場所および色差場所の例を示す概念図である。
- 【図 16】 $N \times N$ 変換を伴う $nR \times 2N$ 予測モードを示す概念図である。 20
- 【図 17】 $2N \times N$ 、 $2N \times nD$ 、および $2N \times nU$ 予測モードのための非正方形四分木 (NSQT) を示す概念図である。
- 【図 18】 $N \times 2N$ 、 $nR \times 2N$ 、および $nL \times 2N$ 予測モードのための NSQT を示す概念図である。
- 【図 19】照明補償 (IC) モデルにおけるパラメータを推定するために使用される隣接画素を示す図である。
- 【図 20】現在のコーディング単位の参照ブロックが現在の予測単位の視差ベクトルまたは動きベクトルを使用することによって発見される IC モデルにおけるパラメータを推定するために使用される例示的な隣接画素を示す概念図である。
- 【図 21】 $2N \times N$ に等しい区分サイズのための例示的な変換構造を示す概念図である。
- 【図 22】本開示の技法による、 $N \times N/4(U)$ に等しい区分サイズのための変換構造を示す概念図である。 30
- 【図 23】本開示の技法による、 $N \times N/4(U)$ に等しい区分サイズのための変換構造を示す概念図である。
- 【図 24】本開示の技法を実施し得る例示的なビデオエンコーダを示すブロック図である。
- 【図 25】本開示の技法を実施するように構成される例示的なビデオデコーダを示すブロック図である。
- 【図 26】本開示の技法による、LM ベースの符号化をサポートする例示的なビデオエンコーダを示すブロック図である。
- 【図 27】本開示の技法による、LM ベースの復号をサポートする例示的なビデオデコーダを示すブロック図である。 40
- 【図 28】本開示の LM ベースのコーディング技法による、ビデオエンコーダの例示的な動作を示すフローチャートである。
- 【図 29】本開示の LM ベースのコーディング技法による、ビデオデコーダの例示的な動作を示すフローチャートである。
- 【図 30】本開示の量子化技法による、ビデオエンコーダの例示的な動作を示すフローチャートである。
- 【図 31】本開示の量子化技法による、ビデオデコーダの例示的な動作を示すフローチャートである。
- 【図 32】IC を使用する本開示の技法による、ビデオエンコーダの例示的な動作を示すフ 50

ローチャートである。

【図 3 3】ICを使用する本開示の技法による、ビデオデコーダの例示的な動作を示すフローチャートである。

【図 3 4】柔軟性のある残差木を使用する本開示の技法による、ビデオエンコーダの例示的な動作を示すフローチャートである。

【図 3 5】柔軟性のある残差木を使用する本開示の技法による、ビデオデコーダの例示的な動作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】

【0035】

全般に、本開示は、イントラおよびインター予測区分、非正方形変換、非正方形ブロックのためのイントラおよびインターコーディングモード、ならびに関連するエン트로ピーコーディングに関する。本開示の技法は、High Efficiency Video Coding(HEVC)の拡張または次世代のビデオコーディング規格などの、先進的なビデオコーデックの文脈において使用され得る。

【0036】

HEVCでは、ビデオコーダ(すなわち、ビデオエンコーダまたはビデオデコーダ)が、ピクチャのコーディング単位(CU)を1つまたは複数の予測単位(PU)へと区分する。ビデオコーダは、CUの各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する。CUの残差データは、CUのPUの予測ブロックとCUの元のコーディングブロックとの差を表す。CUがイントラ予測される(すなわち、CUのPUの予測ブロックがイントラ予測を使用して生成される)事例では、CUの残差データは1つまたは複数の正方形の変換単位(TU)へと区分され得る。しかしながら、CUがインター予測される(すなわち、CUのPUの予測ブロックがインター予測を使用して生成される)事例では、CUの残差データは1つまたは複数の正方形または非正方形のTUへと区分され得る。本開示では、単位(たとえば、CU、PU、TU)の形状への言及は、対応するブロックの形状を指すことがある。したがって、非正方形PUは非正方形予測ブロックを指すものとして解釈されることがあり、非正方形TUは非正方形変換ブロックを指すものとして解釈されることがあり、これらの逆も成り立つ。さらに、予測ブロックは、HEVCにおいて定義されるようなPUの概念に縛られる必要はなく、むしろ、予測(たとえば、インター予測、イントラ予測)が実行されるサンプルのブロックという意味であることに留意されたい。同様に、変換ブロックは、HEVCにおいて定義されるようなTUの概念に縛られる必要はなく、むしろ、変換が適用されるサンプルのブロックという意味である。

【0037】

以下で説明されるように、非正方形TUの導入は、特定のコーディングツールとともに使用されると、いくつかの問題をもたらすことがある。

【0038】

たとえば、線形モデリング(LM)予測モードは、HEVCの開発の間に研究された、成分間の相関を減らすための技法である。ビデオコーダがLM予測モードを使用するとき、ビデオコーダは、CUのPUの再構築された輝度サンプルに基づいて、PUの色差サンプルを予測する。PUの色差サンプルは、PUの色差予測ブロックの色差サンプルである。色差サンプルの例示的なタイプには、CbサンプルおよびCrサンプルがある。ビデオコーダは、PUの輝度予測ブロックのサンプルをPUの対応する輝度残差サンプルと加算することによって、PUの再構築された輝度サンプルを生成することができる。

【0039】

具体的には、ビデオコーダがLM予測モードを使用するとき、ビデオコーダは、場所 (i, j) におけるPUの予測される色差サンプルを $\text{rec}_L(i, j) + \text{LM_pred}(i, j)$ として決定することができ、ここで $\text{rec}_L(i, j)$ は場所 (i, j) におけるPUの再構築された輝度サンプルであり、 LM_pred および LM_scale はパラメータである。4:2:0のカラーフォーマットなどのいくつかの場合には、1つの $M \times K$ 色差ブロックが $2M \times 2K$ 輝度ブロックに対応し、この場合、 $\text{rec}_L(i, j)$ は、 $2M \times 2K$ 輝度ブロックの $(M \times K)$ ダウンサンプリングされたバージョンの (i, j) に位置する値を示す

。ビデオコードは、再構築された輝度参照サンプルおよび再構築された色差参照サンプルの値に基づいて、および の値を決定する。再構築された輝度参照サンプルおよび再構築された色差参照サンプルは、PUの上辺および左辺に沿ったサンプルである。 を決定するための式は、参照サンプルの総数(1と表記され、MとKの合計に等しい)による除算演算を伴う。典型的な場合には、MおよびKは等しく、HEVCでは 2^l によって表すことができ、 l は正の整数値である。予測ブロックが正方形である限り、 l は 2^m に等しく、ここで m は異なる予測ブロックサイズに対して変化することがある。したがって、 l で割るために除算演算を実行する代わりに、ビデオコードは、 の値を計算するときに右シフト演算を実行することができる。右シフト演算は、除算演算よりも実施するのがはるかに速く、かつ複雑ではない。本開示では、CU、TU、およびPUなどの様々なタイプのブロックのサイズへの言及は、それぞれ、CU、TU、およびPUの、コーディングブロック、変換ブロック、および予測ブロックのサイズを指す。さらに、本開示では、CU、TU、およびPUなどの様々なタイプのビデオコーディング単位の辺への言及は、様々なタイプのブロックに対応するブロック(たとえば、コーディングブロック、変換ブロック、予測(prediction)/予測(predictive)ブロック)の辺を指す。

10

20

30

40

50

【0040】

しかしながら、輝度ブロック(たとえば、PUの輝度予測ブロック)が正方形ではない(たとえば、Mが12に等しくKが16に等しい)場合、 l は常に 2^m に等しいとは限らない。したがって、輝度ブロックが正方形ではない場合、 の値を計算するときに除算演算の代わりに右シフト演算を使用することは可能ではないことがある。したがって、ビデオコードは、 の値を計算するために、高価な除算演算を実施する必要がある。得る。

【0041】

本開示は、非正方形ブロックのためのLM予測モードを使用するときに の値を計算するときに、除算演算を実施する必要をなくすることができる技法を説明する。いくつかの場合、MがKに等しいがMが2のべき乗ではない正方形PUに対しても、ここで説明される技法が適用可能であり得る。本技法の例によれば、ビデオコードは、輝度参照サンプルのセットおよび色差参照サンプルのセットを再構築することができる。輝度参照サンプルのセットは、ビデオデータの現在のピクチャの非正方形輝度ブロックの上辺に隣接する輝度サンプルと、非正方形輝度ブロックの左辺に隣接する輝度サンプルとを備え得る。非正方形輝度ブロックは、PUの輝度予測ブロックであり得る。したがって、PUは非正方形PUと見なされ得る。色差参照サンプルのセットは、非正方形色差ブロックの上辺に隣接する色差サンプルと、非正方形色差ブロックの左辺に隣接する色差サンプルとを備え得る。非正方形色差ブロックは、PUの色差予測ブロックであり得る。加えて、ビデオコードは、非正方形予測ブロックの輝度サンプルを再構築し得る。さらに、ビデオコードは、非正方形輝度ブロックの長辺に隣接する輝度参照サンプルのセットの中の輝度参照サンプルの総数が非正方形輝度ブロックの短辺に隣接する輝度参照サンプルのセットの輝度参照サンプルの総数と同じになるように、輝度参照サンプルのセットをサブサンプリングし得る。ビデオコードは、

【0042】

【数9】

$$\alpha = \frac{I \sum x_i \cdot y_i - \sum x_i \cdot \sum y_i}{I \sum x_i \cdot x_i - \sum x_i \cdot \sum x_i}$$

【0043】

に等しい第1のパラメータを決定することができ、ここで l は輝度参照サンプルのセットの中の参照サンプルの総数であり、 x_i は輝度参照サンプルのセットの中の輝度参照サンプルであり、 y_i は色差参照サンプルのセットの中の色差参照サンプルである。加えて、ビデオコードは、

$$=(y_i - \cdot x_i)/l$$

に等しい第2のパラメータを決定することができる。予測色差ブロックの各々のそれぞれの色差サンプルに対して、ビデオコードは、それぞれの色差サンプルの値が、それぞれの色差サンプルに対応するそれぞれの再構築された輝度サンプルにより乗じられた と、とを足したものに等しくなるように、それぞれの色差サンプルの値を決定し得る。予測色差ブロックは、非正方形PUの予測色差ブロックであり得る。ビデオコードは、予測色差ブロックに一部基づいて、コーディングブロックを再構築し得る。

【 0 0 4 4 】

上の例では、非正方形輝度ブロックの長辺に隣接する輝度参照サンプルの総数が非正方形輝度ブロックの短辺に隣接する輝度参照サンプルの総数と同じになるように輝度参照サンプルのセットをサブサンプリングすることによって、ビデオコードは、輝度参照サンプルのセットの中の参照サンプルの総数が2のべき乗になることを確実にし得る。したがって、ビデオコードは、 の値を計算するときに除算演算の代わりに右シフト演算を使用することが可能であり得る。したがって、上の例を実施するビデオコードは、 の値を計算するときに除算演算を使用することを強いられるビデオデコードよりも、複雑ではなく、かつ/または高速であり得る。ビデオコードは、上の例で説明された活動を、シフト演算の代わりに除算演算を使用して実行することができるが、そのようなビデオコードは除算演算の代わりにシフト演算を使用することの利益を得られないことに留意されたい。いくつかの例では、非正方形予測ブロックの短辺または長辺に隣接する参照サンプルが利用可能ではないことがあり、この場合、他方の辺に位置する利用可能な参照サンプルに対してサブサンプリング処理を実行する必要はないことがある。

【 0 0 4 5 】

HEVCでは、ビデオエンコーダは、残差データのブロックを変換係数のブロックへと変換するために、残差データのブロック(すなわち、変換ブロック)に変換を適用する。高いレベルにおいて、ビデオエンコーダは、N点の1次元DCT変換を変換ブロックの列に適用することで中間値のブロックをまず生成することによって、変換係数のブロック(すなわち、変換係数ブロック)を生成し得る。Nは変換ブロックの高さおよび幅に等しい。ビデオエンコーダは次いで、同じN点の1次元DCT変換を中間値のブロックの行に適用することによって、変換係数のブロックを生成し得る。ビデオデコードは、変換ブロックを復元するのと同様の方法で変換を逆にする。

【 0 0 4 6 】

上の議論からわかるように、HEVCにおいて変換を適用するプロセスは、変換ブロックが正方形であることに依存する。しかしながら、変換ブロックが非正方形であることが望ましいことがある。たとえば、インター予測ブロックまたはイントラ予測ブロックの境界に変換ブロックの境界がまたがるとき、圧縮性能が低下することがある。非正方形予測ブロックの使用は、正方形の領域に入らないオブジェクトを捉えるのに価値があり得る。したがって、非正方形予測ブロックおよび/または非正方形変換は、コーディング性能の改善の点で有用であり得る。変換行列係数がN点の1次元DCT変換である場合、変換行列係数は

【 0 0 4 7 】

【 数 1 0 】

$$\sqrt{N}$$

【 0 0 4 8 】

に等しい分母を用いて定義される。本開示以前には、分母

【 0 0 4 9 】

【数 1 1】

$$\sqrt{N}$$

【0 0 5 0】

は、正規化係数として考えられており、量子化プロセスでは右シフトによって実装されていた。2次元DCT変換、たとえば $K \times L$ 変換を考慮すると、正規化係数は

【0 0 5 1】

【数 1 2】

10

$$(\sqrt{K} * \sqrt{L})$$

【0 0 5 2】

である。 N が式 $\log_2(N*N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ を満たすものにより定義される場合、利用される正規化係数 $(\text{sqrt}(N) * \text{sqrt}(N))$ と実際の正規化係数

【0 0 5 3】

【数 1 3】

20

$$(\sqrt{K} * \sqrt{L})$$

【0 0 5 4】

との比は、

【0 0 5 5】

【数 1 4】

$$1/\sqrt{2}$$

30

【0 0 5 6】

である。正方形変換(たとえば、列と行の両方に適用される N 点の変換)を非正方形変換ブロックに直接適用すると、正規化係数の増大により、得られる変換係数ブロックにおける総エネルギー(すなわち、量子化後のすべての変換された係数の平方根の合計)が変わることがあり、これは圧縮性能の低下をもたらす。

【0 0 5 7】

本開示の他の箇所で詳細に説明されるように、サイズ $K \times L$ の変換ブロックに対して、 $(\log_2(K) + \log_2(L))$ が奇数であるときに変換係数を

【0 0 5 8】

【数 1 5】

40

$$\sqrt{2}$$

【0 0 5 9】

と乗じるビデオエンコード、および、 $(\log_2(K) + \log_2(L))$ が奇数であるときに変換係数を

【0 0 6 0】

【数 1 6】

 $\sqrt{2}$

【0061】

で割るビデオデコーダが、この問題に対処することができる。

【0062】

3D-HEVCは、3次元(3D)ビデオデータのためのHEVCの拡張である。3D-HEVCは、異なる視点からの同じシーンの複数のビューを規定する。3D-HEVCの標準化作業には、HEVCに基づくマルチビュービデオコーデックの標準化が含まれる。3D-HEVCでは、異なるビューからの再構築されたビュー成分(すなわち、再構築されたピクチャ)に基づくビュー間予測が可能である。さらに、3D-HEVCは、ビュー間動き予測およびビュー間残差予測を実装する。

10

【0063】

ビデオの同じ時間インスタンスを表す各ビューのピクチャは、類似したビデオコンテンツを含む。しかしながら、ビューのビデオコンテンツは、互いに対して空間的にずれていることがある。具体的には、ビューのビデオコンテンツは、同じシーン上の異なる視点を表すことがある。たとえば、第1のビューにおけるピクチャの中のビデオブロックは、第2のビューにおけるピクチャの中のビデオブロックと類似したビデオコンテンツを含み得る。この例では、第1のビューにおけるピクチャの中のビデオブロックの位置と、第2のビューにおけるピクチャの中のビデオブロックの位置は異なることがある。たとえば、異なるビューの中のビデオブロックの位置の間には、何らかのずれがあり得る。

20

【0064】

ビデオブロックの視差ベクトルは、このずれの尺度を提供する。たとえば、第1のビューにおけるピクチャのビデオブロックは、第2のビューにおけるピクチャの中の対応するビデオブロックのずれを示す視差ベクトルと関連付けられ得る。

【0065】

異なるカメラ設定または光源からの異なる距離が原因で、同じ時間インスタンスに対応するが異なるビューの中にあるピクチャは、ほぼ同じ画像を含み得るが、一方のピクチャの中のオブジェクトが、他方のピクチャの中の対応するオブジェクトより明るいことがある。照明補償(IC)は、ビュー間予測を実行するときにビュー間のそのような照明の差を補償するための、3D-HEVCにおいて実装される技法である。3D-HEVCでは、ビデオデコーダは、現在のピクチャの現在のCUの現在のPUの視差ベクトルを決定する。加えて、ビデオデコーダは、現在のCUのために2つのICパラメータを計算し得る。本開示は、これらのICパラメータをaおよびbと表記する。加えて、現在のPUの輝度予測ブロックの各々のそれぞれのサンプルに対して、ビデオデコーダは、

30

$$p(i, j) = a * r(i + dv_x, j + dv_y + b)$$

を計算する。上の式において、 $p(i, j)$ は現在のPUの輝度予測ブロックのそれぞれのサンプルであり、 (i, j) は現在のピクチャの左上の角に対する相対的なそれぞれのサンプルの位置を示す座標であり、 dv_x は現在のPUの視差ベクトルの水平成分であり、 dv_y は現在のPUの視差ベクトルの垂直成分であり、aおよびbはICパラメータである。

40

【0066】

本開示の他の箇所により詳細に説明されるように、3D-HEVCにおいてICパラメータbを定義する式は、現在のCUの上辺および左辺に隣接する参照サンプルの数による除算演算を伴う。3D-HEVCでは、現在のCUの上辺および左辺に隣接する参照サンプルの数は常に2のべき乗である。その結果、ICパラメータbを定義する式における除算演算は、右シフト演算を使用して実施され得る。本開示の他の箇所により説明されるように、右シフト演算は、除算演算よりも実施するのがはるかに複雑ではないことがあり、かつ除算演算を実施するよりはるかに速いことがある。

【0067】

50

2次元ビデオコーディングでは、H. Liu、「Local Illumination Compensation」、ITU-Telecommunications Standardization Sector、Study Group 16 Question 6、Video Coding Experts Group(VCEG)、第52回会合、2015年6月19～26日、ワルシャワ、ポーランド、文書VCEG-AZ06 (以後、「VCEG-AZ06」)に記載されるように、各々のインターモードコーディングされるコーディング単位(CU)に対して、局所照明補償(LIC)が適応的に有効または無効にされ、LICは、スケーリング係数 a およびオフセット b を使用した、照明変化に対する線形モデルに基づく。LICがCUに適用されるとき、CUに属する各PU/サブPUに対して、LICパラメータが、CUのサブサンプリングされた(2:1のサブサンプリング)隣接サンプルと参照ピクチャの中の対応する画素(現在のPU/サブPUの動き情報によって特定される)とを使用する方法で導出される。 $N \times N$ に等しいサイズのCUでは、パラメータ計算に使用される境界画素の総数は $2N$ ではなく N である。例が図20に示されている。LICパラメータは、各予測方向に対して別々に導出され適用される。上述の隣接サンプルに基づいてパラメータ a および b を導出するために、最小二乗誤差法が利用される。

10

20

30

40

50

【0068】

ICは、CUが単一のPUのみを有するときだけに使用されていた。しかしながら、CUが2つまたは3つのPUへと区分される事例、および/またはCUが非対称に区分される事例を含めて、CUが複数のPUを有する事例において、ICを使用するのが望ましいことがある。そのような事例では、現在のCUの上辺および左辺に隣接する参照サンプルの数は、もはや2のべき乗ではないことがある。したがって、右シフト演算を使用してICパラメータ b を計算することは可能ではないことがある。むしろ、ビデオコードは、ICパラメータ b を計算するために、より低速でより複雑な除算演算を使用しなければならないことがある。

【0069】

この問題に対処するために、ビデオコードは、参照サンプルの第1のセットをサブサンプリングして、全体で 2^m 個の参照サンプルを含む参照サンプルの第1のサブサンプリングされたセットを生成することができ、ここで m は整数である。本開示では、サブサンプリングという用語は、サンプルのセットからの1つまたは複数のサンプルの選択を示し、ダウンサンプリングは、いくつかの参照サンプルがフィルタリングされたサンプルを導出するために一緒に使用され得るフィルタリングプロセスを示す。参照サンプルのセットは、PUの非正方形予測ブロックの外側の、非正方形予測ブロックの左辺および上辺に沿ったサンプルを備え得る。したがって、本明細書では、参照サンプルは近隣サンプルまたは隣接サンプルとも呼ばれることがある。加えて、ビデオコードは、参照サンプルの第2のセットをサブサンプリングして、全体で 2^m 個の参照サンプルを含む参照サンプルの第2のサブサンプリングされたセットを生成することができ、ここで m は整数である。参照サンプルの第2のセットは、参照ブロック(たとえば、ビュー間参照ブロックまたは時間参照ブロック)の外側の、参照ブロックの左辺および上辺に沿ったサンプルを備え得る。ビデオコードは次いで、参照サンプルの第1のサブサンプリングされたセットおよび参照サンプルの第2のサブサンプリングされたセットに基づいて、少なくともICパラメータ b を決定し得る。参照サンプルの第1のサブサンプリングされたセットおよび参照サンプルの第2のサブサンプリングされたセットは各々 2^m 個のサンプルを含むので、ビデオコードは、ICパラメータ b を計算するために除算演算の代わりに右シフト演算を使用し得る。このようにして、本開示の技法は、ビデオコードの複雑さを下げ、かつ/またはビデオコーディングを加速することができる。

【0070】

上で言及されたように、非正方形TUの導入はいくつかの問題をもたらすことがある。たとえば、CUをTUへと区分するこれまでの技法は、CUのPUが正方形ではない場合であっても、四分木の分割パターンに従っていた。本開示では、四分木(quad-tree)は四分木(quarter-tree)とも呼ばれることがある。常に四分木の分割パターンを使用すると、四分木の分割パターンによりTUの辺がCUのPUの辺と揃わない場合には特に、最適ではないビデオデータ圧縮性能をもたらすことがある。

【0071】

したがって、本開示の技法によれば、CUの変換木は、四分木の分割パターンに限定されない。むしろ、変換木の中のノードは2つの子ノードを有し得る。したがって、一例では、ビデオデコーダは、木構造に基づいてCUがTUへと区分されると決定し得る。この例では、ビデオデコーダは、木構造の中のノードが木構造の中で丁度2つの子ノードを有すると決定し得る。この例では、木構造のルートノードはCUのコーディングブロックに対応し、木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応し、木構造のリーフノードはCUのTUに対応する。いくつかの例では、変換木の中のノードは2つまたは4つの子ノードを有し得る。2つまたは4つの子ノードを有するというノードの柔軟性は、ビデオコーディングの圧縮性能を向上させ得る。

10

【0072】

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されたビデオデータを提供するソースデバイス12を含む。具体的には、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にビデオデータを提供する。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、いわゆる「スマート」フォンなどの電話ハンドセット、タブレットコンピュータ、テレビジョン、カメラ、ディスプレイデバイス、デジタルメディアプレーヤ、ビデオゲームコンソール、ビデオストリーミングデバイスなどを含む、広範囲のデバイスのうちのいずれかを備え得る。いくつかの場合、ソースデバイス12および宛先デバイス14は、ワイヤレス通信に対応し得る。

20

【0073】

図1の例では、ソースデバイス12は、ビデオソース18、ビデオデータを記憶するように構成される記憶媒体19、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。宛先デバイス14は、入力インターフェース28、ビデオデータを記憶するように構成される記憶媒体29、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。他の例では、ソースデバイスおよび宛先デバイスは、他の構成要素または構成を含み得る。たとえば、ソースデバイス12は、外部カメラなどの外部ビデオソースからビデオデータを受信し得る。同様に、宛先デバイス14は、統合された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てもよい。

30

【0074】

図1の図示されたシステム10は一例にすぎない。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/または復号デバイスによって実行され得る。一般に、本開示の技法はビデオ符号化デバイスによって実行されるが、技法はまた、通常は「コーデック」と呼ばれるビデオエンコーダ/デコーダによって実行され得る。ソースデバイス12および宛先デバイス14は、単に、ソースデバイス12が宛先デバイス14に送信するためのコーディングされたビデオデータを生成するようなコーディングデバイスの例である。いくつかの例では、デバイス12、14は、デバイス12、14の各々がビデオ符号化および復号構成要素を含むように実質的に対称的な方式で動作し得る。したがって、システム10は、たとえば、ビデオストリーミング、ビデオ再生、ビデオ放送、またはビデオ電話のための、ビデオデバイス12、14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。

40

【0075】

ソースデバイス12のビデオソース18は、ビデオカメラ、以前にキャプチャされたビデオを含むビデオアーカイブ、および/またはビデオコンテンツプロバイダからビデオデータを受信するためのビデオフィードインターフェースなどのビデオキャプチャデバイスを含み得る。さらなる代替として、ビデオソース18は、ソースビデオとしてのコンピュータグラフィックススペースのデータ、またはライブビデオとアーカイブされたビデオとコンピュータで生成されたビデオとの組合せを生成し得る。いくつかの場合、ソースデバイス12および宛先デバイス14は、いわゆるカメラ電話またはビデオ電話を形成し得る。ソースデバ

50

イス12は、ビデオデータを記憶するように構成される1つまたは複数のデータ記憶媒体(たとえば、記憶媒体19)を備え得る。本開示において説明される技法は、ビデオコーディング全般に適用可能であることがあり、ワイヤレスおよび/または有線の適用例において適用されることがある。各場合において、キャプチャされた、事前にキャプチャされた、またはコンピュータで生成されたビデオは、ビデオエンコーダ20によって符号化され得る。出力インターフェース22が次いで、符号化されたビデオ情報をコンピュータ可読媒体16へと出力し得る。

【0076】

出力インターフェース22は、様々なタイプの構成要素またはデバイスを備え得る。たとえば、出力インターフェース22は、ワイヤレス送信機、モデム、有線ネットワーク構成要素(たとえば、イーサネット(登録商標)カード)、または別の物理的な構成要素を備え得る。出力インターフェース22がワイヤレス受信機を備える例では、出力インターフェース22は、4G、4G-LTE、LTE Advanced、5Gなどのセルラー通信規格に従って変調される、ビットストリームなどのデータを受信するように構成され得る。出力インターフェース22がワイヤレス受信機を備えるいくつかの例では、出力インターフェース22は、IEEE 802.11仕様、IEEE 802.15仕様(たとえば、ZigBee(商標))、Bluetooth(登録商標)規格などの他のワイヤレス規格に従って変調される、ビットストリームなどのデータを受信するように構成され得る。いくつかの例では、出力インターフェース22の回路は、ビデオエンコーダ20の回路および/またはソースデバイス12の他の構成要素に組み込まれ得る。たとえば、ビデオエンコーダ20および出力インターフェース22は、システムオンチップ(SoC)の一部であり得る。SoCは、汎用マイクロプロセッサ、グラフィックス処理装置などの、他の構成要素も含み得る。

【0077】

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された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に符号化された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がリアルタイムで宛先デバイス14へ符号化されたビデオデータを直接送信することを可能にする通信媒体を備え得る。符号化されたビデオデータは、ワイヤレス通信プロトコルなどの通信規格に従って変調され、宛先デバイス14へ送信され得る。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、任意のワイヤレスまたは有線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の、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への通信を容易にするのに有用であり得る任意の他の機器を含み得る。宛先デバイス14は、符号化されたビデオデータおよび/または復号されたビデオデータを記憶するように構成される1つまたは複数のデータ記憶媒体を備え得る。

【0078】

いくつかの例では、出力インターフェース22は、符号化されたデータを記憶デバイスに出力し得る。同様に、入力インターフェース28は、記憶デバイスからの符号化されたデータにアクセスし得る。記憶デバイスは、ハードドライブ、Blu-ray(登録商標)ディスク、DVD、CD-ROM、フラッシュメモリ、揮発性メモリもしくは不揮発性メモリ、または符号化されたビデオデータを記憶するための任意の他の適切なデジタル記憶媒体などの、分散されるかまたは局所的にアクセスされる様々なデータ記憶媒体のいずれかを含み得る。さらなる例では、記憶デバイスは、ソースデバイス12によって生成された符号化されたビデオを記憶し得るファイルサーバまたは別の中間記憶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、記憶デバイスからの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、符号化されたビデオデータを記憶するとともにその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、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は、(たとえば、ウェブサイトのための)

ウェブサーバ、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、またはローカルディスクドライブを含む。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的なデータ接続を介して、符号化されたビデオデータにアクセスすることができる。これは、ファイルサーバに記憶された符号化されたビデオデータにアクセスするのに適した、ワイヤレスチャネル(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が含み得る。記憶デバイスからの符号化されたビデオデータの送信は、ストリーミング送信、ダウンロード送信、またはそれらの組合せであり得る。

【0079】

本技法は、オーバージ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、dynamic adaptive streaming over 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ているデジタルビデオ、データ記憶媒体上に記憶されたデジタルビデオの復号、または他の用途などの、様々なマルチメディア用途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、ビデオ放送、および/またはビデオ電話などの用途をサポートするために、一方向または双方向ビデオ送信をサポートするように構成され得る。

10

【0080】

コンピュータ可読媒体16は、ワイヤレス放送もしくは有線ネットワーク送信などの一時媒体、またはハードディスク、フラッシュドライブ、コンパクトディスク、デジタルビデオディスク、Blu-ray(登録商標)ディスク、もしくは他のコンピュータ可読媒体などの記憶媒体(すなわち、非一時的記憶媒体)を含み得る。いくつかの例では、ネットワークサーバ(図示せず)が、たとえば、ネットワーク送信を介して、ソースデバイス12から符号化されたビデオデータを受信し、符号化されたビデオデータを宛先デバイス14に提供し得る。同様に、ディスクスタンピング設備などの媒体製造設備のコンピューティングデバイスが、ソースデバイス12から符号化されたビデオデータを受信し、符号化されたビデオデータを含むディスクを製造し得る。したがって、コンピュータ可読媒体16は、様々な形態の1つまたは複数のコンピュータ可読媒体を含むものと理解され得る。

20

【0081】

宛先デバイス14の入力インターフェース28は、コンピュータ可読媒体16から情報を受信する。入力インターフェース28は、様々なタイプの構成要素またはデバイスを備え得る。たとえば、入力インターフェース28は、ワイヤレス受信機、モデム、有線ネットワーク構成要素(たとえば、イーサネット(登録商標)カード)、または別の物理的な構成要素を備え得る。入力インターフェース28がワイヤレス受信機を備える例では、入力インターフェース28は、4G、4G-LTE、LTE Advanced、5Gなどのセルラー通信規格に従って変調される、ビットストリームなどのデータを受信するように構成され得る。入力インターフェース28がワイヤレス受信機を備えるいくつかの例では、入力インターフェース28は、IEEE 802.11仕様、IEEE 802.15仕様(たとえば、ZigBee(商標))、Bluetooth(登録商標)規格などの他のワイヤレス規格に従って変調される、ビットストリームなどのデータを受信するように構成され得る。いくつかの例では、入力インターフェース28の回路は、ビデオデコーダ30の回路および/または宛先デバイス14の他の構成要素に組み込まれ得る。たとえば、ビデオデコーダ30および入力インターフェース28は、システムオンチップ(SoC)の一部であり得る。SoCは、汎用マイクロプロセッサ、グラフィックス処理装置などの、他の構成要素も含み得る。

30

40

【0082】

コンピュータ可読媒体16の情報は、ビデオエンコーダ20によって定義され、ビデオデコーダ30によっても使用される、シンタックス情報を含むことがあり、シンタックス情報は、ブロックおよび他のコーディングされたユニット、たとえばピクチャグループ(GOP)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。ディスプレイデバイス32は、復号されたビデオデータをユーザに表示し得る。たとえば、宛先デバイス14またはビデオ

50

オデコーダ30は、ディスプレイデバイス32による表示のために、ビデオデータの再構築されたピクチャを出力し得る。そのような再構築されたピクチャは、再構築されたブロックを備え得る。ディスプレイデバイス32は、陰極線管(CRT)、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のいずれかを備え得る。

【0083】

ビデオエンコーダ20およびビデオデコーダユニット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、ソフトウェア、ハードウェア、ファームウェア、またはそれらの任意の組合せなどの、様々な適切なエンコーダ回路のいずれかとして実装され得る。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、適切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の1つまたは複数のプロセッサを使用してハードウェアで命令を実行し得る。ビデオエンコーダ20およびビデオデコーダ30の各々は、1つまたは複数のエンコーダまたはデコーダに含まれることがあり、そのいずれもが、それぞれのデバイスにおいて複合エンコーダ/デコーダ(コーデック)の一部として統合されることがある。

【0084】

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、ビデオコーディング規格に従って動作し得る。例示的なビデオコーディング規格は、限定はされないが、そのスケーラブルビデオコーディング(SVC)拡張およびマルチビュービデオコーディング(MVC)拡張を含む、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 MPEG-4 AVCとも知られる)を含む。加えて、新しいビデオコーディング規格、すなわちHigh Efficiency Video Coding(HEVC)が、ITU-T Video Coding Experts Group(VCEG)およびISO/IEC Motion Picture Experts Group(MPEG)のJoint Collaboration Team on Video Coding(JCT-VC)によって最近開発された。Wang他、「High Efficiency Video Coding (HEVC) Defect Report」、Joint Collaborative Team in Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、第14回会合、ウィーン、オーストリア、2013年7月25日~8月2日、文書JCTVC-N1003_v1(以後、「JCTVC-N1003」)は、HEVC規格の草案である。JCTVC-N1003は、http://phenix.int-evry.fr/jct/doc_end_user/documents/14_Vienna/wg11/JCTVC-N1003-v1.zipから入手可能である。

【0085】

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1つまたは複数のサンプルアレイを含み得る。ピクチャの各々のそれぞれのサンプルアレイは、それぞれの色成分に対するサンプルのアレイを備え得る。HEVCでは、ピクチャは、 S_L 、 S_{Cb} 、および S_{Cr} と表記される3つのサンプルアレイを含み得る。 S_L は、輝度サンプルの2次元アレイ(すなわち、ブロック)である。 S_{Cb} は、Cb色差サンプルの2次元アレイである。 S_{Cr} は、Cr色差サンプルの2次元アレイである。他の事例では、ピクチャはモノクロームであってよく、輝度サンプルのアレイしか含まないことがある。

【0086】

ビデオデータを符号化することの一部として、ビデオエンコーダ20はビデオデータのピクチャを符号化し得る。言い換えると、ビデオエンコーダ20は、ビデオデータのピクチャの符号化された表現を生成し得る。ピクチャの符号化された表現は、「コーディングされたピクチャ」または「符号化されたピクチャ」と呼ばれ得る。

【0087】

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、コーディングツリー単位(CTU)のセットを生成し得る。CTUの各々は、輝度サンプルのCTBと、色差サンプルの2つの対応するCTBと、CTB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つの別々の色平面を有するピク

チャでは、CTUは、単一のCTBと、そのCTB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。CTB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「最大コーディング単位」(LCU)と呼ばれることもある。シンタックス構造は、指定された順序でビットストリームと一緒に現れる0個以上のシンタックス要素として定義され得る。スライスは、ラスタ走査順序で連続的に順序付けられた整数個のCTUを含み得る。

【0088】

本開示は、1つまたは複数のサンプル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1つまたは複数のサンプルブロックおよびシンタックス構造を指すために、「ビデオ単位」または「ビデオブロック」または「ブロック」という用語を使用することがある。例示的なタイプのビデオ単位は、CTU、CU、PU、変換単位(TU)、マクロブロック、マクロブロック区分などを含み得る。いくつかの文脈では、PUの議論はマクロブロックまたはマクロブロック区分の議論と相互に交換され得る。

【0089】

HEVCでは、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は、CTUのコーディングツリーブロック上で四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、「コーディングツリー単位」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、輝度サンプルアレイ、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輝度サンプルのコーディングブロックおよび色差サンプルの2つの対応する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。モノクロームピクチャまたは3つの別々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。

【0090】

ビデオエンコーダ20は、ビデオデータのピクチャのCUを符号化し得る。CUを符号化することの一部として、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)ブロックである。予測単位(PU)は、輝度サンプルの予測ブロック、色差サンプルの2つの対応する予測ブロック、および予測ブロックを予測するために使用されるシンタックス構造を備え得る。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の予測ブロック(たとえば、輝度予測ブロック、Cb予測ブロック、およびCr予測ブロック)の予測ブロック(たとえば、輝度予測ブロック、Cb予測ブロック、およびCr予測ブロック)を生成し得る。

【0091】

したがって、一般に、PUは、サンプルの1つまたは複数の予測ブロックと、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。HEVCなどのいくつかの例示的なコーデックでは、PUはCUのサブユニットであり得る。他の例示的なコーデックでは、CUとPUの間には区別がないことがある。いくつかの例では、他の用語がPUのために使用されることがある。

【0092】

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を含む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。ビデオエンコーダ20が現在のピクチャの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、参照ピクチャ(すなわち、現在のピクチャ以外のピクチャ)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。

10

20

30

40

50

【0093】

ビデオエンコーダ20がCUの1つまたは複数のPUの予測ブロック(たとえば、輝度予測ブロック、Cb予測ブロック、およびCr予測ブロック)を生成した後、ビデオエンコーダ20は、CUの1つまたは複数の残差ブロックを生成し得る。たとえば、ビデオエンコーダ20は、CUの輝度残差ブロックを生成し得る。CUの輝度残差ブロックの中の各サンプルは、CUの予測輝度ブロックの1つの中の輝度サンプルとCUの元の輝度コーディングブロック中の対応するサンプルとの差分を示す。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残差ブロックの中の各サンプルは、CUの予測Cbブロックのうちの1つの中のCbサンプルと、CUの元のCbコーディングブロックの中の対応するサンプルとの差分を示し得る。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残差ブロックの中の各サンプルは、CUの予測Crブロックのうちの1つの中のCrサンプルと、CUの元のCrコーディングブロックの中の対応するサンプルとの間の差分を示し得る。

10

【0094】

繰り返すと、HEVCでは、スライスの中の最大のコーディング単位はCTUと呼ばれる。各ピクチャはCTUへと分割され、このCTUは、特定のタイルまたはスライスに対するラスタ走査順序でコーディングされ得る。CTUは、正方形ブロックであり、四分木すなわちコーディングツリーのルートを表す。CTUは、そのノードがCUである四分木を含む。いくつかの事例では、CTUのサイズは、HEVCメインプロファイルでは 16×16 から 64×64 にわたり得る(ただし、技術的には 8×8 のCTBサイズがサポートされ得る)。いくつかの事例では、CTUサイズは、 8×8 から 64×64 までの輝度サンプルにわたり得るが、通常は 64×64 が使用される。各CTUは、CUと呼ばれるより小さい正方形ブロックにさらに分割され得る。CUは、CTUと同じサイズであり得るが、CUは 8×8 程度の小ささであり得る。各CUは、1つのモードでコーディングされる。たとえば、CUはインターコーディングまたはイントラコーディングされ得る。CUがインターコーディングされるとき、CUは、2つもしくは4つのPUへとさらに区分されることがあり、またはさらなる区分が適用されないときには1つだけのPUになることがある。1つのCUの中に2つのPUが存在するとき、2つのPUは、CUの半分のサイズの長方形であることがあり、またはCUの $1/4$ もしくは $3/4$ のサイズを有する2つの長方形であることがある。CUがインターコーディングされるとき、PUごとに動き情報の1つのセットが存在する。加えて、各PUは、動き情報のセットを導出するために固有のインター予測モードでコーディングされる。言い換えると、各PUは動き情報の固有のセットを有し得る。

20

30

【0095】

さらに、ビデオエンコーダ20は、CUの残差ブロックを1つまたは複数の変換ブロックに分解し得る。たとえば、ビデオエンコーダ20は、四分木区分を使用して、CUの残差ブロック(たとえば、輝度残差ブロック、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、輝度変換ブロック、Cb変換ブロック、およびCr変換ブロック)に分解し得る。変換ブロックは、同じ変換が適用されるサンプルの矩形(たとえば、正方形または非正方形)ブロックである。

【0096】

CUの変換単位(TU)は、輝度サンプルの変換ブロックと、色差サンプルの2つの対応する変換ブロックと、それらの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、輝度変換ブロック、Cb変換ブロック、およびCr変換ブロックを有し得る。TUの輝度変換ブロックは、CUの輝度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。モノクロームピクチャまたは3つの別々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、その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。

40

【0097】

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の輝度変換ブロックに適用して、TUの輝度係数ブロックを生成し得る。係数ブロックは、

50

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。いくつかの例では、1つまたは複数の変換は、画素領域から周波数領域に変換ブロックを変換する。

【0098】

いくつかの例では、ビデオエンコーダ20は、変換ブロックに変換を適用しない。言い換えると、ビデオエンコーダ20は、変換ブロックへの変換の適用をスキップする。そのような例では、ビデオエンコーダ20は、変換係数と同じ方法で残差サンプル値を扱い得る。したがって、ビデオエンコーダ20が変換の適用をスキップする例では、変換係数および係数ブロックの以下の議論は、残差サンプルの変換ブロックに適用可能であり得る。

【0099】

係数ブロック(たとえば、輝度係数ブロック、Cb係数ブロック、またはCr係数ブロック)を生成した後に、ビデオエンコーダ20は、係数ブロックを量子化し得る。いくつかの例では、ビデオエンコーダ20は係数ブロックを量子化しない。ビデオエンコーダ20が変換ブロックに変換を適用しない例では、ビデオエンコーダ20は、変換ブロックの残差サンプルを量子化することがあり、またはしないことがある。量子化は一般に、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、さらなる圧縮を実現するプロセスを指す。ビデオエンコーダ20が係数ブロックを量子化した後に、ビデオエンコーダ20は、量子化された変換係数または残差サンプル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または残差サンプル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。いくつかの例では、ビデオエンコーダ20は、パレットベースコーディングを使用してCUを符号化する。したがって、符号化されたブロック(たとえば、符号化されたCU)は、量子化された変換係数を示すエントロピー符号化されたシンタックス要素を含み得る。

【0100】

ビデオエンコーダ20は、ビデオデータの符号化されたピクチャおよび関連するデータ(すなわち、符号化されたピクチャと関連付けられるデータ)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。したがって、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ユニットは、NALユニットの中のデータのタイプを示すものと、必要に応じてエミュレーション防止ビットが散りばめられているローバイトシーケンスペイロード(RBSP)の形態でそのデータを含むバイトとを含む、シンタックス構造である。NALユニットの各々は、NALユニットヘッダを含むことがあり、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ている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ビットを含む。

【0101】

ビデオデコーダ30は、ビデオエンコーダ20によって生成されたビットストリームを受信し得る。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、ビットストリームから取得されたシンタックス要素に少なくとも一部に基づいて、ビデオデータのピクチャを再構築し得る。ビデオデータのピクチャを再構築するためのプロセスは、全般に、ピクチャを符号化するため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、ビデオデータのピクチャを再構築するために、ビデオデコーダ30は、ビットストリームから取得されるシンタックス要素および/または外部ソースからのデータに基づいて、ピクチャのCUなどのブロックを復号し得る。

【0102】

いくつかの例では、ピクチャの現在のCUを復号することの一部として、ビデオデコーダ30は、インター予測またはイントラ予測を使用して、現在のCUの各PUの1つまたは複数の

10

20

30

40

50

予測ブロックを生成し得る。インター予測を使用するとき、ビデオデコーダ30は、現在のCUのPUの予測ブロックを決定するために、PUの動きベクトルを使用し得る。加えて、ビデオデコーダ30は、いくつかの例では、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、いくつかの例では、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復号されたサンプル(たとえば、残差サンプル)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。

【0103】

10

その上、HEVCでは、タイルと呼ばれる長方形の領域へとピクチャを区分するための選択肢が規定されている。タイルの主な目的は、誤り耐性をもたらすことではなく、並列処理の能力を上げることである。タイルは、何らかの共有されるヘッダ情報を用いて符号化される、ピクチャの独立に復号可能な領域である。タイルは加えて、ビデオピクチャのローカル領域への空間的なランダムアクセスの目的で使用され得る。ピクチャの典型的なタイル構成は、各タイルの中に概ね等しい数のCTUを有する長方形領域へと、ピクチャをセグメント化することからなる。タイルは、より粗い粒度で(ピクチャ/サブピクチャ)並列性を実現し、タイルの使用には、スレッドの洗練された同期は不要である。

【0104】

20

残差ブロックの様々な特性を適応させるために、残差四分木(RQT)を使用した変換コーディング構造がHEVCにおいて適用され、これは、<http://www.hhi.fraunhofer.de/fields-of-competence/image-processing/researchgroups/image-video-coding/hevc-high-efficiency-video-coding/transform-coding-using-the-residual-quadtree-rqt.html>において入手可能な、Marpe他、「Transform Coding Using the Residual Quadtree (RQT)」、Fraunhofer Heinrich Hertz Instituteにおいて簡単に説明されている。CTUがCUに再帰的に分割された後、各CUは、PUおよびTUにさらに分割される。TUへのCUの区分は、四分木手法に基づいて再帰的に実行され、したがって、各CUの残差信号は、木構造すなわち残差四分木(RQT)によってコーディングされる。RQTは、 4×4 から 32×32 までの輝度サンプルのTUサイズを許容する。図2Aおよび図2Bは、HEVCにおける残差四分木に基づく例示的な変換方式を示す概念図である。具体的には、図2Aは、文字aからjによりラベルを付けられた10個のTUおよび対応するブロック区分をCU40が含む例を示す。図2Bは、図2AのCUのためのRQTを示す概念図である。

30

【0105】

ビデオコーダは、個々のTUを深度優先の木横断順序で処理し、この深度優先の木横断順序は、図2Aではアルファベットの順序として図示され、これは深度優先の横断を用いる再帰的なZ走査に従う。四分木手法は、残差信号の変化する空間周波数特性に対する変換の適応を可能にする。通常、空間的なサポートがより大きい、より大きな変換ブロックサイズは、より良い周波数分解能を提供する。しかしながら、空間的なサポートがより小さい、より小さな変換ブロックサイズは、より良い空間分解能を提供する。この2つ、すなわち空間分解能と周波数分解能との間のトレードオフは、たとえばレートひずみ最適化技法に基づいて、エンコーダのモード判断によって選ばれる。レートひずみ最適化技法は、コーディングモード(たとえば、特定のRQT分割構造)ごとにコーディングビットと再構築ひずみとの加重和、すなわちレートひずみコストを計算し、最小のレートひずみコストを有するコーディングモードを最良のモードとして選択する。

40

【0106】

3つのパラメータすなわち、木の最大深度、最小の許容される変換サイズ、および最大の許容される変換サイズが、RQTにおいて定義される。HEVCでは、最小変換サイズおよび最大変換サイズは、 4×4 サンプルから 32×32 サンプルまでの範囲内で変化することができ、このサイズは、前の段落で言及されたサポートされるブロック変換に対応する。RQTの最大の許容される深度は、TUの個数を制約する。0に等しい最大深度は、各々の含まれるT

50

Uが最大の許容される変換サイズ、たとえば 32×32 に達した場合に、CTUがそれ以上分解され得ないことを意味する。HEVCでは、より大きなサイズの変換、たとえば 64×64 の変換は、主に、利益が限られていることと、相対的により解像度が小さいビデオに対して相対的により複雑になることが原因で、採用されていなかった。

【0107】

HEVCでは、TUのサイズとは無関係に、TUの残差(たとえば、TUの係数ブロック)が、重複しない係数グループ(CG)を用いてコーディングされる。CGの各々は、TUの 4×4 ブロックの係数を含む。たとえば、 32×32 のTUは全体で64個のCGを有し、 16×16 のTUは全体で16個のCGを有する。TUのCGは、ある事前に定義された走査順序に従ってコーディングされる。各CGをコーディングするとき、現在のCGの内部の係数は、 4×4 ブロックのある事前に定義された走査順序に従って走査され、コーディングされる。図3は、HEVCにおける係数グループに基づく例示的な係数走査を示す概念図である。具体的には、図3は、4個の 4×4 のCGを含む 8×8 のTUに対する係数走査を示す。

10

【0108】

上で述べられたように、ビデオエンコーダ20およびビデオデコーダ30は、イントラ予測を実行して予測ブロックを生成し得る。イントラ予測は、空間的に隣接する再構築された画像サンプルを使用して、画像ブロック予測を実行する。図4は、 16×16 ブロックに対するイントラ予測の例を示す概念図である。図4では、ブロック正方形は 16×16 ブロック50を含む。図4では、ブロック50は、選択された予測方向に沿った上および左の隣接する再構築されたサンプル52、54(すなわち、参照サンプル)により予測される。図4では、黒いボックスの外側のサンプルが参照サンプルである。図4の白い矢印は選択された予測方向を示す。

20

【0109】

図5は、HEVCにおいて定義される35個のイントラ予測モードの例を示す概念図である。図5に示されるように、HEVCは、輝度ブロックのイントラ予測のために35個のモード(平面モード、DCモード、および33個の角度モードを含む)を定義する。HEVCにおいて定義されるイントラ予測の35個のモードは、以下の表において示されるようにインデックスを付けられる。

【0110】

【表1】

30

Table 1-イントラ予測モードおよび関連する名称の仕様

イントラ予測モード	関連する名称
0	INTRA_PLANAR
1	INTRA_DC
2..34	INTRA_ANGULAR2..INTRA_ANGULAR34

40

【0111】

HEVCイントラコーディングは、 $2N \times 2N$ および $N \times N$ という、2つのタイプのPU分割をサポートする。 $2N \times 2N$ は、CUを1つのPUへと分割する。言い換えると、CUはCUと同じサイズの1つのPUを有する。 $N \times N$ は、CUを4つの等しいサイズのPUへと分割する。しかしながら、区分タイプPART_ $N \times N$ により規定される4つの領域は、区分タイプPART_ $2N \times 2N$ を用いて4つのより小さいCUによっても表され得る。これにより、HEVCは、最小のCUサイズでのみ、イントラCUが4つのPUへと分割されることを許容する。

50

【0112】

図6は、HEVCにおいて定義される平面モードを示す概念図である。平面モードは通常、最も頻繁に使用されるイントラ予測モードである。N×Nブロックのために平面予測を実行するために、(x, y)に位置する各サンプル p_{xy} に対して、予測値は、双線形フィルタを用いて、4つの特定の隣接する再構築されたサンプル、すなわち参照サンプルを使用して計算される。4つの参照サンプルは、右上の再構築されたサンプルTR、左下の再構築されたサンプルBL、ならびに現在のサンプルと同じ列($r_{x, -1}$)および行($r_{-1, y}$)に位置する2つの再構築されたサンプル60、62を含む。平面モードは以下のように定式化され得る。

$$p_{xy} = ((N - 1 - x) \cdot L + (x + 1) \cdot TR + (N - 1 - y) \cdot T + (y + 1) \cdot BL + N) \gg (\text{Log2}(N) + 1) \quad (1)$$

上の式(1)において、Lは再構築されたサンプル60に対応し、Tは再構築されたサンプル62に対応する。DCモードでは、予測ブロックは単に、隣接する再構築されたサンプルの平均値により埋められる。一般に、平面モードとDCモードの両方が、滑らかに変化する画像領域および不変の画像領域をモデル化するために適用される。

【0113】

図7は、HEVCにおいて定義される例示的な角度モードの概念図である。HEVCにおける角度イントラ予測モードのイントラ予測プロセスは、次のように説明される。与えられる各角度イントラ予測モードに対して、イントラ予測方向がそれによって特定され得る。たとえば、与えられる角度イントラ予測モードは、図5に従って特定され得る。図5に示されるように、イントラモード18は純粋な水平の予測方向に対応し、イントラモード26は純粋な垂直の予測方向に対応する。ある特定のイントラ予測方向を仮定すると、予測ブロックの各々のそれぞれのサンプルに対して、それぞれのサンプルの座標(x, y)がまず、予測方向に沿った隣接する再構築されたサンプルの行または列に投影される。たとえば、図7の例に示されるように、予測ブロック72のサンプル70の座標(x, y)は、特定のイントラ予測方向74に沿って投影される。(x, y)が2つの隣接する再構築されたサンプルLとRとの間の小数位置に投影されると仮定する。そうすると、(x, y)の予測値は、次のように定式化される2タップの双線形補間フィルタを使用して計算される。

$$p_{xy} = (1 - \alpha) \cdot L + \alpha \cdot R$$

HEVCでは、浮動小数点数演算を避けるために、上の計算は次のように整数の算術を使用して近似される。

$$p_{xy} = ((32 - a) \cdot L + a \cdot R + 16) \gg 5$$

ここで、aは $32 \cdot \alpha$ に等しい整数である。

【0114】

図8は、HEVCにおけるインター予測のためのCUを分割するための区分モードの概念図である。図8に示されるように、HEVCでは、インターコーディングされるCUは、1つ、2つ、または4つの区分へと分割されることが可能であり、この分割の様々なタイプが可能である。インター予測されるコーディングブロックのための区分の可能性が、図8に示されている。上の4つの区分タイプは、サイズN×NのCUを分割するのではなく、CUをサイズN×N/2またはN/2×Nの2つの区分へと分割し、CUをサイズN/2×N/2の4つの区分へと分割するケースを、それぞれ示す。図8の下側の4つの区分タイプは、非対称動き区分(AMP)と呼ばれる。AMPモードの一方の区分はそれぞれ、高さまたは幅がN/4、かつ幅または高さがNであり、他方の区分は、高さまたは幅が3N/4で幅または高さがNであることによってCUの残りを埋める。各々のインターコーディングされる区分は、1つまたは2つの動きベクトルおよび参照ピクチャインデックスを割り当てられる。

【0115】

イントラスライスに対しては、イントラ予測モードのみが許可される。したがって、予測モードをシグナリングする必要はない。しかしながら、インタースライス(PまたはBスライス)に対しては、イントラ予測モードとインター予測モードの両方が許可される。したがって、HEVCでは、各CUに対して、1つのフラグpred_mode_flagが非スキップモードのためにシグナリングされる。CUのためにHEVCにおいて定義されるシンタックスおよびセマ

ンティクスの部分的な一覧が下で提示される。

7.3.8.5 コーディング単位シンタックス

【 0 1 1 6 】

【 表 2 】

<code>coding_unit(x0, y0, log2CbSize) {</code>	記述子	
<code>if(transquant_bypass_enabled_flag)</code>		
<code> cu_transquant_bypass_flag</code>	<code>ae(v)</code>	10
<code>if(slice_type != I)</code>		
<code> cu_skip_flag[x0][y0]</code>	<code>ae(v)</code>	
<code> nCbS = (1 << log2CbSize)</code>		
<code> if(cu_skip_flag[x0][y0])</code>		
<code> prediction_unit(x0, y0, nCbS, nCbS)</code>		20
<code> else {</code>		
<code> if(slice_type != I)</code>		
<code> pred_mode_flag</code>	<code>ae(v)</code>	
<code> if(CuPredMode[x0][y0] != MODE_INTRA log2CbSize == MinCbLog2SizeY)</code>		
<code> part_mode</code>	<code>ae(v)</code>	30
<code> ...</code>		
<code> }</code>		

【 0 1 1 7 】

1に等しい`cu_skip_flag[x0][y0]`は、現在のコーディング単位に対して、PまたはBスライスを復号するとき、`cu_skip_flag[x0][y0]`の後には、統合候補インデックス`merge_idx[x0][y0]`を除くシンタックス要素がそれ以上構文解析されないことを指定する。0に等しい`cu_skip_flag[x0][y0]`は、コーディング単位がスキップされないことを指定する。アレイインデックス`x0`、`y0`は、考慮されるコーディングブロックの左上輝度サンプルの、ピクチャの左上輝度サンプルに対する相対的な位置(`x0`, `y0`)を指定する。`cu_skip_flag[x0][y0]`が存在しないとき、`cu_skip_flag[x0][y0]`は0と等しいと推測される。

【 0 1 1 8 】

0に等しい`pred_mode_flag`は、現在のコーディング単位がインター予測モードでコーディングされることを指定する。1に等しい`pred_mode_flag`は、現在のコーディング単位がイントラ予測モードでコーディングされることを指定する。変数`CuPredMode[x][y]`は、 $x = x0..x0 + nCbS - 1$ および $y = y0..y0 + nCbS - 1$ に対して、次のように導出される。

- `pred_mode_flag`が0に等しい場合、`CuPredMode[x][y]`は`MODE_INTER`に等しく

設定される。

- それ以外の(pred_mode_flagが1に等しい)場合、CuPredMode[x][y]はMODE_INTRAに等しく設定される。

【 0 1 1 9 】

pred_mode_flagが存在しないとき、変数CuPredMode[x][y]は、 $x = x0..x0 + nCbS - 1$ および $y = y0..y0 + nCbS - 1$ に対して、次のように導出される。

- slice_typeがIに等しい場合、CuPredMode[x][y]はMODE_INTRAに等しいと推測される。

- それ以外の(slice_typeがPまたはBに等しい)場合、cu_skip_flag[x0][y0]が1に等しいとき、CuPredMode[x][y]はMODE_SKIPに等しいと推測される。

10

【 0 1 2 0 】

HEVCを開発する過程で、およびその後に、HEVCを改良するための様々な提案が行われている。たとえば、Jianle Chen他、「Further improvements to HMkTA-1.0」、文書: VCEG-AZ07_v2、第52回会合、2015年6月19~26日、ワルシャワ、ポーランド(以後、「VCEG-AZ07」)は、短距離イントラコーディング方式を記載する。イントラ予測のために正方形ブロックを常に產生する従来のブロック区分方法とは異なり、VCEG-AZ07の短距離イントラ予測(SDIP)方式は、HEVCの四分木ベースのブロック構造のもとで、非正方形のブロック分割を利用する。VCEG-AZ07において説明されるように、ブロックは、幅または高さが4分の1である4つの非正方形ブロックへと分割され、各々の非正方形ブロックが、予測の基本単位として扱われる。非正方形ブロックは、順番にコーディングおよび再構築され、次の隣接ブロックに対するイントラ予測のための参照画素を提供することができる。したがって、参照画素とローカル画素との間の距離を減らすことができ、イントラ予測の精度を大幅に改善することができる。

20

【 0 1 2 1 】

図9は、SDIP単位区分の概念図である。SDIP方式では、 64×64 より小さいCUは、サイズが $N/2 \times 2N$ または $2N \times N/2$ である4つの垂直または水平な長方形PUへと分割され得る(これらの区分モードは $hN \times 2N$ および $2N \times hN$ と本開示では呼ばれることがあり、hは半分(half)を意味する)。4つのPUが、 $hN \times 2N$ モードで左から右に、かつ $2N \times hN$ モードで上から下に、順番にコーディングおよび再構築される。図9では、点線はPU/TUの分割を表し、影付きの領域78は、 32×8 のTUがRQT構造において4つの 32×2 のTUへと分割されることを示す。区分モードが $2N \times hN$ である右上の 32×32 のCUは4つの 32×8 のPUへと分割され、モードが $hN \times 2N$ である左下の 16×16 のCUは4つの 4×16 のPUへと分割され、右下の 8×8 のCUは 8×2 のPUへと分割され、以下同様である。 $2N \times 2N$ モードの左下の 16×16 のCUおよび $N \times N$ モードの右下の 8×8 のCUなどの、HEVCにおけるCUの正方形の分割も存在し得る。

30

【 0 1 2 2 】

さらに、SDIP方式では、 $M \times N (M > N)$ のTUは、サイズが $M \times N/4$ である、または $M < N$ であるときにはサイズが $M/4 \times N$ である、4つのTUへと分割され得る。言い換えると、SDIP方式における分割は常に、CUにおける同じ方向(垂直または水平)に沿って行われるべきである。以下のTable 2(表3)は、SDIP方式に存在するすべての非正方形単位と、対応するRQT深度とを列挙する。

40

【 0 1 2 3 】

【表 3】

Table 2-SDIP 方式における単位および対応する RQT 深度のリスト

CU サイズ	深度=1(PU サイズに等しい)のときの単位サイズ	深度=2 のときの単位サイズ
32×32	32×8	32×2
	8×32	2×32
16×16	16×4	-
	4×16	-
8×8	8×2	-
	2×8	-

10

20

【 0 1 2 4 】

Xiaoran Cao他、「CE6.b1 Report on Short Distance Intra Prediction Method」、文書.JCTVC-E0278、Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、第5回会合、ジュネーブ、2011年3月16～23日(以後、「Cao 1」)では、 $1 \times N$ および $N \times 1$ のような区分がさらに含まれ、対応するRQT深度および変換サイズが下のTable 3(表4)に列挙される。

【 0 1 2 5 】

【表 4】

Table 3-Cao 1 における SDIP の変換サイズのリスト

CU サイ ズ	深度=1(PU サイズに等しい)のときの単位サ イズ	深度=2 のときの単位サイズ
32×32	32×8	32×2
	8×32	2×32
16×16	16×4	16×1
	4×16	1×16
8×8	8×2	-
(2N×2N)	2×8	-
8×8	4×4	4×1
(N×N)		1×4

10

20

【 0 1 2 6 】

さらに、一部のSDIP方式は、非正方形変換およびエントロピーコーディングを使用する。たとえば、 $n \times m$ 変換が非正方形ブロックのために使用される。 $n \times m$ ($n > m$) ブロックに対して、順方向変換は次のように記述される。

30

【 0 1 2 7 】

【数 1 7】

$$C_{n \times m} = T_m \cdot B_{n \times m} \cdot T_n^T \quad (1)$$

【 0 1 2 8 】

上の式において、 $B_{n \times m}$ は n 個の行および m 個の列を有するブロックを表し、 T_n および T_m はそれぞれ、サイズが $n \times n$ および $m \times m$ である変換行列であり、 $C_{n \times m}$ は変換されるブロックを表す。 T_n および T_m は、HEVCにおける変換行列と同じである。したがって、ハードウェア実装では、非正方形ブロックのために変換部分が再使用され得る。 $n \times m$ ($n < m$) ブロックに対して、ブロックはまず $m \times n$ ($m > n$) ブロックへと転置され、次いで式(1)のように変換される。エントロピーコーディングでは、重複する実装を避けるために、正方形ブロックの係数コーディングも再使用される。たとえば、図10は、走査され 8×8 行列82へと並べ替えられる 16×4 係数行列80の概念図である。この例では、図10に示されるように、係数行列80の係数はまず、高周波から低周波へと走査されて1Dバッファ84へ入れられ、次いで、ジグザグの順序で 8×8 行列82へと並べ替えられ、 8×8 行列82はHEVCにおける既存の方法を使用してコーディングされる。

40

【 0 1 2 9 】

HEVCに対する提案された改良の別の例である、Liu他、「Rectangular ($2N \times N$, $N \times 2N$)

50

Intra Prediction」、文書 JCTVC-G135、Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、第7回会合、ジュネーブ、2011年11月21～30日(以後、「JCTVC-G135」)は、インターコーディングに対する $2N \times N$ および $N \times 2N$ の区分サイズの使用をイントラコーディングへと拡張することを説明する。追加のPUサイズおよび対応するTUは、下のTable 2(表5)で与えられる。JCTVC-G135では、従来の変換木構造が利用される。

【 0 1 3 0 】

【表 5】

Table 2-JCTVC-G135 における変換サイズのリスト

CU サイズ	PU サイズ	深度=1 のときの単位サイズ	深度=2 のときの単位サイズ
32×32	32×16	32×8	32×2
	16×32	8×32	2×32
16×16	16×8	16×4	-
	8×16	4×16	-
8×8	8×4	8×2	-
	4×8	2×8	-

【 0 1 3 1 】

VCEG-AZ07では、以下の技法が説明された。自然なビデオに現れるより鮮明なエッジの方向を捉えるために、VCEG-AZ07は、方向性イントラモードを、HEVCにおいて定義されるような33個から65個へと拡張することを提案した。図11は、提案された67個のイントラ予測モードを示す概念図である。VCEG-AZ07において説明される方向性モードが図11では点線の矢印として示されており、平面モードおよびDCモードは同じままである。VCEG-AZ07において提案されたより密な方向性イントラ予測モードは、すべてのPUサイズに、かつ輝度イントラ予測と色差イントラ予測の両方に適用される。

【 0 1 3 2 】

方向性イントラモードの数の増大に対応するために、VCEG-AZ07は、6つの最確モード(MPM)を使用した、改善されたイントラモードコーディング方法を提案した。1)6つのMPMの導出、および2)6つのMPMのエントロピーコーディングという、2つの主要な技術的な態様が関与する。6つのMPMのセットを導出するとき、VCEG-AZ06は、左および上の隣接イントラモードの定義を変更した。HEVCのように上および左の隣接ブロックからのイントラモードを直接使用するのではなく、上の隣接行に沿った、かつ左の隣接列に沿った最も頻繁に使用されるイントラモードが計算され、次いで、左および上の隣接モードとしてそれぞれ使用される。

【 0 1 3 3 】

さらに、VCEG-AZ07において説明されるように、方向性イントラ予測の精度を改善するために、4タップのイントラ補間フィルタが利用される。たとえば、図7に関して上で説明されたように、HEVCは、2タップの線形補間フィルタを使用して、方向性予測モード(すな

わち、平面予測子およびDC予測子を除くイントラ予測モード)でイントラ予測ブロックを生成する。具体的には、図7の例では、ビデオコーダは、2タップのフィルタをサンプルLおよびRに適用して、サンプル50の予測値を決定する。フィルタを2つの参照サンプルに適用して予測ブロックのサンプルの予測値を決定するHEVCの手法とは対照的に、VCEG-AZ07は、フィルタを4つの参照サンプルに適用して予測ブロックのサンプルの予測値を決定する。VCEG-AZ07では、 4×4 および 8×8 ブロックに対するキュービック補間フィルタ、および 16×16 以上のブロックに対するガウシアン補間フィルタという、2つのタイプの4タップの補間フィルタが使用される。VCEG-AZ07では、フィルタのパラメータはブロックサイズに従って固定され、すべての予測される画素に対して、すべての方向性モードにおいて、同じフィルタが使用される。

10

【0134】

HEVCでは、イントラ予測ブロックが垂直イントラモードおよび水平イントラモードのために生成された後で、予測サンプル(すなわち、予測ブロックのサンプル)の一番左の列および一番上の行がそれぞれ、さらに調整される。最大で4つの列または行の境界サンプルがさらに、2タップのフィルタ(イントラモード2および34に対する)または3タップのフィルタ(イントラモード3~6および30~33に対する)を使用して調整される。

【0135】

図12Aおよび図12Bは、イントラモード30~34に対する境界フィルタの概念図である。具体的には、図12Aは、イントラモード34に対する境界フィルタの概念図である。図12Bは、イントラモード30~33に対する境界フィルタの概念図である。図12Aおよび図12Bにおいて、ブロックの一番左の列は参照サンプルのセットであり、ブロックの残りはイントラ予測されるブロックのサンプルである。ビデオコーダは、従来の方式でイントラ予測ブロックのサンプルを生成し得る。しかしながら、イントラ予測モード30~34に対して、ビデオエンコーダは、1つまたは複数の追加のフィルタを影付きの画素に適用し得る。したがって、イントラモード34および30~33に対する境界予測フィルタの例が図12Aおよび図12Bに示されており、イントラモード2および3~6に対する境界予測フィルタは同様である。

20

【0136】

具体的には、図12Aにおいて、イントラモード34に対して、ビデオコーダは、それぞれのサンプルの上および右の参照サンプルに基づいて、イントラ予測ブロックの各々のそれぞれのサンプルを生成する。しかしながら、これは、左の参照サンプルから入手可能な情報を無視することがある。したがって、ビデオコーダは、左から4番目までの列に4つの異なるフィルタを適用し得る。イントラ予測ブロックの線1~4の中の各々のそれぞれのサンプルに対して、ビデオコーダは、イントラモード34の反対方向(すなわち、左および下)に、それぞれのサンプルおよび参照サンプルに基づいてフィルタを適用する。線1に対して、得られるサンプルは $(8*a + 8*b)/16$ として計算することができ、ここでaはそれぞれのサンプルであり、bは参照サンプルである。線2に対して、得られるサンプルは $(12*a + 4*b)/16$ として計算することができ、ここでaはそれぞれのサンプルであり、bは参照サンプルである。図12Bの例では、イントラ予測モード30~33の方向は、完全に整数の場所の画素と揃わない。むしろ、予測ブロックの各サンプルに対して、イントラ予測モード30~33に対する方向は、参照サンプルのうちの2つの間の小数場所における参照サンプルと交差する。したがって、イントラ予測モード30~33に境界フィルタを適用するとき、予測ブロックの一番左の列の各サンプルに対して2つの参照サンプルがある。図12Bの例では、予測ブロックの一番左の列の各々のそれぞれのサンプルに対して、イントラ予測モードが33であるとき、ビデオコーダは、それぞれのサンプルの値を $(8*a + 8*b + 2*c)$ として計算することができ、ここでaはそれぞれのサンプルであり、bは参照サンプルのうちの一方であり、cは参照サンプルのうちの他方である。

30

40

【0137】

ビデオコーディングは、色空間およびカラーフォーマットに基づいて実行され得る。たとえば、カラービデオは、色を効率的に表すために様々な色空間が使用されるマルチメディアシステムにおいて、欠かせない役割を果たしている。色空間は、複数の成分を使用し

50

て、数値を用いて色を指定する。普及している色空間は、RGB色空間であり、RGB色空間では、色が3つの原色成分値(すなわち、赤色、緑色、および青色)の組合せとして表現される。カラービデオの圧縮のために、A. FordおよびA. Roberts、「Colour space conversions」、ウェストミンスター大学、ロンドン、Tech. Rep、1998年8月に記載されるような、YCbCr色空間が広く使用されている。

【0138】

YCbCrは、線形変換を介してRGB色空間から容易に変換されることが可能であり、異なる成分間の冗長性、すなわち成分間冗長性は、YCbCr色空間において大きく低減される。YCbCrの1つの利点は、Y信号が輝度情報を伝えるので、白黒TVとの後方互換性があることである。加えて、色差帯域幅は、RGBにおけるサブサンプリングよりも主観的影響が著しく少ないように4:2:0色差サンプリングフォーマットにおいてCb成分およびCr成分をサブサンプリングすることによって、低減され得る。これらの利点のために、YCbCrは、ビデオ圧縮において主要な色空間となっている。また、YCoCgなどの、ビデオ圧縮において使用される他の色空間もある。本開示では、使用される実際の色空間にかかわらず、YCbCr色空間が、各ビデオ圧縮方式における3つの色成分を表すために使用される。

10

【0139】

成分間冗長性はYCbCr色空間において大きく低減されるが、3つの色成分の間の相関は依然として存在する。3つの色成分の間の相関をさらに低減することによってビデオコーディング性能を改善するために、様々な技法が研究されている。

【0140】

たとえば、Xiaoran Cao他、「Short distance intra coding scheme for HEVC」、2012 Picture Coding Symposium (PCS)、pp. 501~504、2012年5月7~9日、クラクフ、ポーランド(以後、「Cao 2」)は、短距離イントラコーディング方式を説明する。Cao 2は、線形モデル(LM)予測モードと名付けられる、4:2:0色差ビデオコーディングにおける方法を説明し、これはHEVC規格の開発中に研究された。たとえば、以後JCTVC-E266と呼ばれる、J. Chen他、「CE6.a.4: Chroma intra prediction by reconstructed luma samples」、Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC 1/SC29/WG11、JCTVC-E266、第5回会合、ジュネーブ、2011年3月16~23日を参照されたい。4:2:0サンプリングでは、2つの色差アレイの各々が、輝度アレイの半分の高さおよび半分の幅を有する。LM予測モードでは、色差サンプルは、次のように線形モデルを使用することによって、同じブロックの再構築された輝度サンプルに基づいて予測される。

20

30

$$\text{pred}_c(i, j) = \text{rec}_L(i, j) + \quad (2)$$

ここで、 $\text{pred}_c(i, j)$ は、現在のブロックの中の色差サンプルの予測を表し、 $\text{rec}_L(i, j)$ は、現在のブロックのダウンサンプリングされた再構築された輝度サンプルを表す。パラメータ α および β は、現在のブロックの周りの、原因となる再構築されたサンプルから導出される。ブロックの原因となるサンプルは、復号順序でそのブロックより前にあるサンプルである。色差ブロックサイズが $N \times N$ により表される場合、 i と j の両方が範囲 $[0, N]$ の中にある。

【0141】

式(2)におけるパラメータ α および β は、現在のブロックの周りの隣接する再構築された輝度サンプルと色差サンプルとの間の回帰誤差を最小化することによって導出される。

40

【0142】

【数18】

$$E(\alpha, \beta) = \sum_i (y_i - (\alpha \cdot x_i + \beta))^2 \quad (3)$$

【0143】

パラメータ α および β は、次のように解かれる。

50

【 0 1 4 4 】

【 数 1 9 】

$$\alpha = \frac{I \sum x_i \cdot y_i - \sum x_i \cdot \sum y_i}{I \sum x_i \cdot x_i - \sum x_i \cdot \sum x_i} \quad (4)$$

$$\beta = (\sum y_i - \alpha \cdot \sum x_i) / I \quad (5)$$

10

【 0 1 4 5 】

上の式において、 x_i はカラーフォーマットが4:4:4ではないダウンサンプリングされた再構築された輝度参照サンプルであり(すなわち、カラーフォーマットは、1つの色差サンプルが複数の輝度サンプルに対応するようなカラーフォーマットである)、 y_i はダウンサンプリングを伴わない再構築された色差参照サンプルであり、 I は参照サンプルの数である。言い換えると、ビデオコーデは、カラーフォーマットが4:4:4ではないことに基づいて、再構築された輝度参照サンプルをダウンサンプリングするが、カラーフォーマットが4:4:4であることに基づいて、再構築された輝度参照サンプルをダウンサンプリングするのを控え得る。ターゲットの $N \times N$ 色差ブロックに対して、左と上の両方の原因となるサンプルが利用可能であるとき、関与するサンプルの総数 I は $2N$ に等しい。左または上の原因となるサンプルのみが利用可能であるとき、関与するサンプルの総数 I は N に等しい。ここで、 N は常に 2^m に等しい(m は異なるCUサイズに対しては異なり得る)。したがって、複雑さを下げるために、式(4)および(5)における除算演算を実施するために、シフト演算が使用され得る。

20

【 0 1 4 6 】

図13は、およびの導出のために使用されるサンプルの例示的な位置を示す概念図である。具体的には、図13は、PUの色差予測ブロック90および同じPUの輝度予測ブロック92を示す。色差サンプルは輝度サンプルに対して相対的にダウンサンプリングされているので、色差予測ブロック90の幅および高さ(すなわち、 N)は、輝度予測ブロック92の幅および高さ(すなわち、 $2N$)の半分である。図13の例では、大きな暗い正方形の外側の小さい正方形が参照サンプルである。図13の例では、小さい円は、LMパラメータおよびを決定するために使用されるサンプル値を示す。図13の例に示されるように、LMパラメータおよびを決定するために使用される色差サンプル値は、色差予測ブロック90の参照サンプルと同じである。しかしながら、LMパラメータおよびを決定するために使用される輝度サンプル値は、輝度参照サンプルから補間される。LMパラメータおよびを決定するために使用される輝度サンプルの得られるセットの総数は、LMパラメータおよびを決定するために使用される色差サンプルの数と同じである。

30

【 0 1 4 7 】

一般に、LM予測モードが現在のPUに対して適用されるとき、ビデオコーデは以下のステップを実行し得る。まず、ビデオコーデが、現在のPUの輝度ブロックを再構築し得る。現在のPUの輝度ブロックを再構築することの一部として、ビデオコーデは、イントラ予測を実行して現在のPUの輝度予測ブロックを決定し得る。さらに、現在のPUの輝度ブロックを再構築することの一部として、ビデオコーデは、現在のPUの輝度予測ブロックに残差データを加算して、現在のPUの輝度ブロックを再構築し得る。第2に、ビデオコーデが、現在のPUの上辺および左辺に隣接する参照輝度サンプルをダウンサンプリングし得る。第3に、ビデオコーデが、上の式(4)および(5)を使用して、現在のPUの上辺および左辺に隣接する色差参照サンプルならびにダウンサンプリングされた輝度参照サンプルに基づいて線形パラメータ(すなわち、および)を導出し得る。本開示は、線形パラメータを「スケーリング係数」とも呼ぶことがある。第4に、ビデオコーデが、現在のPUの再構築された輝度ブロックをダウンサンプリングし得る。第5に、ビデオコーデは、上の式(2)を使用して

40

50

、現在のPUのダウンサンプリングされた輝度ブロックおよび線形パラメータから色差サンプルを予測し得る(たとえば、予測色差ブロックを導出し得る)。

【0148】

上で述べられたように、ビデオコードは、現在のPUの再構築された輝度ブロックをダウンサンプリングし得る。ビデオコードは、現在のPUの再構築された輝度ブロックを様々な方法でダウンサンプリングし得る。たとえば、色差成分の典型的なサンプリング比は、輝度成分のサンプリング比の半分であり、4:2:0サンプリングでは垂直方向に0.5サンプルの位相差を有するので、現在のPUの再構築された輝度サンプルは、色差信号のサイズおよび位相と一致するように、垂直方向にはダウンサンプリングされ、水平方向にはサブサンプリングされる。たとえば、0から現在のPUの予測色差ブロックの幅から1を引いた数までの各値*i*、および0から現在のPUの予測色差ブロックの高さから1を引いた数までの各値*j*に対して、ビデオコードは次を計算し得る。

$$\text{rec}_L(i, j) = (\text{Rec}_{L\text{Orig}}[2i, 2j] + \text{Rec}_{L\text{Orig}}[2i, 2j + 1]) \gg 1 \quad (6)$$

上の式において、 $\text{rec}_L(i, j)$ は、現在のPUのダウンサンプリングされた再構築された輝度ブロックの左上の角に対する相対的な、場所(*i*, *j*)に対応する輝度サンプルである。 $\text{Rec}_{L\text{Orig}}[2i, 2j]$ および $\text{Rec}_{L\text{Orig}}[2i, 2j+1]$ は、現在のPUの元の再構築された輝度ブロックの左上の角に対する相対的な、場所(*2i*, *2j*)および(*2i*, *2j+1*)における再構築された輝度サンプルである。したがって、式(6)において、現在のPUのダウンサンプリングされた再構築された輝度ブロックの場所(*i*, *j*)における輝度サンプルは、現在のPUの元の再構築された輝度ブロックの場所(*2i*, *2j*)における輝度サンプルと、現在のPUの元の再構築された輝度ブロックの場所(*2i*, *2j+1*)における輝度サンプルとの平均である。

【0149】

図14は、現在のPUの再構築される輝度ブロックのサンプルをダウンサンプリングするための輝度場所および色差場所の例を示す概念図である。図14は、色差サンプルを三角形として、輝度サンプルを円として示す。ビデオコードは、[1, 1]フィルタを適用することによって、2つの輝度サンプル(図14では2つのベタの円により表される)から現在の色差サンプル(図14ではベタの三角形により表される)の値を予測する。[1, 1]フィルタは2タップフィルタの一例である。[1, 1]フィルタでは、2つのタップが等しく重み付けられる。図14の各々のそれぞれの三角形に対して、ビデオコードは、それぞれの三角形の上および下の円によって表されるサンプルに式(6)を適用して、それぞれの三角形によって表されるサンプルのそれぞれの輝度値を決定し得る。

【0150】

さらに、上で述べられたように、ビデオコードは、輝度参照サンプルをダウンサンプリングし得る。ビデオコードは、輝度参照サンプルを様々な方法でダウンサンプリングし得る。図14に示されるように、現在のPUの予測色差ブロックの列は、現在のPUの予測輝度ブロックの列と揃っている。4:2:0カラーフォーマットを使用する一例では、現在の輝度ブロックの上辺に隣接するダウンサンプリングされた輝度参照サンプルは、輝度参照サンプルのセットの中のインデックスが偶数の場所における各輝度参照サンプルからなり得る。したがって、0から現在のPUの予測色差ブロックの幅から1を引いた数までの範囲にある*i*の各々のそれぞれの値に対して、ダウンサンプリングプロセスは次のように定義され得る。

$$\text{rec}_L(i, -1) = \text{Rec}_{L\text{Orig}}[2i, -1] \quad (7)$$

上の式において、 $\text{rec}_L(i, -1)$ は、現在のPUの色差予測ブロックの左上の角に対する相対的な場所(*i*, -1)におけるダウンサンプリングされた輝度参照サンプルである。 $\text{Rec}_{L\text{Orig}}[2i, -1]$ は、現在のPUの元の予測輝度ブロックの左上の角に対する相対的な場所(*2i*, -1)における輝度参照サンプルである。

【0151】

図14に示されるように、現在のPUの予測色差ブロックの行は、4:2:0カラーフォーマットでは、現在のPUの予測輝度ブロックの行と揃っていない。しかしながら、LMベースの予測のためのパラメータ および を計算するための式(4)および(5)は、各色差参照サン

ルに対して1つの輝度参照サンプルがあることに基づいている。したがって、現在のPUの予測色差ブロックのそれぞれの行に対して、ビデオコードは、予測色差ブロックの行の上の現在のPUの予測輝度ブロックの行の中の輝度参照サンプルと、予測色差ブロックの行の下現在のPUの予測輝度ブロックの行の中の輝度参照サンプルとの平均を計算し得る。たとえば、0から予測色差ブロックの中の行の数から1を引いた数までの範囲にある各値jに対して、ビデオコードは、左の隣接輝度参照サンプルの値を次のように計算し得る。

$$\text{rec}_L(-1, j) = (\text{Rec}_{\text{Lorig}}[-2, 2j] + \text{Rec}_{\text{Lorig}}[-2, 2j + 1]) \gg 1 \quad (8)$$

上の式において、 $\text{rec}_L(-1, j)$ は、現在のPUの予測色差ブロックの左上の角に対する相対的な場所(-1, j)におけるダウンサンプリングされた輝度参照サンプルである。 $\text{Rec}_{\text{Lorig}}[-2, 2j]$ および $\text{Rec}_{\text{Lorig}}[-2, 2j+1]$ は、現在のPUの元の予測輝度ブロックの左上の角に対する相対的な場所(-2, 2j)および(-2, 2j+1)における元の輝度サンプルである。

【 0 1 5 2 】

他のダウンサンプリング技法も提案されている。たとえば、Yi-Jen Chiu他、「Cross-channel techniques to improve intra chroma prediction」、Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、JCTVC-F502、第6回会合、トリノ、イタリア、2011年7月14～22日(本明細書では「JCTVC-F502」と呼ばれる)では、2タップフィルタを使用する代わりに、ビデオコードは、現在の輝度ブロックと隣接する輝度ブロックの両方に2次元の6タップフィルタリングを適用する。2次元フィルタの係数セットは、

【 0 1 5 3 】

【 数 2 0 】

$$\begin{bmatrix} 1 & 2 & 1 \\ 1 & 2 & 1 \end{bmatrix} / 8 \quad (9)$$

【 0 1 5 4 】

である。ダウンサンプリングされた輝度サンプルは式(10)によって導出される。

$$\begin{aligned} \text{rec}_L(i, j) = & (\text{Rec}_{\text{Lorig}}[2i, 2j] * 2 + \text{Rec}_{\text{Lorig}}[2i, 2j+1] + \text{Rec}_{\text{Lorig}}[2i, 2j-1] \\ & + \text{Rec}_{\text{Lorig}}[2i+1, 2j] * 2 + \text{Rec}_{\text{Lorig}}[2i+1, 2j+1] + \text{Rec}_{\text{Lorig}}[2i+1, 2j-1]) \gg 3 \end{aligned} \quad (10)$$

上の式において、 $\text{rec}_L(i, j)$ は現在のPUのダウンサンプリングされた再構築された輝度ブロックの左上の角に対する相対的な場所(i, j)における再構築された輝度サンプルであり、 $\text{Rec}_{\text{Lorig}}[\dots]$ は現在のPUの元の再構築された輝度ブロックの左上の角に対する相対的な場所における現在のPUの元の再構築された輝度ブロックの再構築された輝度サンプルである。

【 0 1 5 5 】

たとえば、ビデオコードは、式(10)の演算を実行してダウンサンプリングされた輝度ブロックを決定し得る。式(10)は、 $\text{Rec}_{\text{Lorig}}[2i, 2j]$ 、 $\text{Rec}_{\text{Lorig}}[2i, 2j+1]$ 、 $\text{Rec}_{\text{Lorig}}[2i, 2j-1]$ 、 $\text{Rec}_{\text{Lorig}}[2i+1, 2j]$ 、 $\text{Rec}_{\text{Lorig}}[2i+1, 2j+1]$ 、および $\text{Rec}_{\text{Lorig}}[2i+1, 2j-1]$ を6つの入力サンプルとして伴う、[1, 2, 1; 1, 2, 1]によって表されるようなビルトイン6タップフィルタを含む。フィルタのタップ数は、どれだけの入力サンプルがフィルタを適用するために使用されるかを示す。たとえば、式(10)では、ビデオコードは、再構築された輝度ブロックからの6つの値を使用して、ダウンサンプリングされた輝度ブロックを生成する。

【 0 1 5 6 】

図15は、予測ブロックを生成するための輝度ブロックのサンプルをダウンサンプリングするための輝度場所および色差場所の例を示す概念図である。図15に示されるように、ビデオコードは、6タップフィルタを適用することによって、6つのベタの円により表される6つの輝度サンプルから、ベタの三角形により表される色差サンプルを予測する。色差サンプルの予測子は、式(2)で定義されるような線形関数を使用して導出されるので、6タッ

プフィルタが適用されるとき、1つの色差サンプルの予測子は6つの隣接輝度サンプルに依拠することがわかり得る。式(2)と(10)を組み合わせると、結果は以下の式(11)である。

$$\text{pred}_C(i, j) = \cdot (\text{Rec}_{\text{Lorig}}[2i, 2j] * 2 + \text{Rec}_{\text{Lorig}}[2i, 2j+1] + \text{Rec}_{\text{Lorig}}[2i, 2j-1] + \text{Rec}_{\text{Lorig}}[2i+1, 2j] * 2 + \text{Rec}_{\text{Lorig}}[2i+1, 2j+1] + \text{Rec}_{\text{Lorig}}[2i+1, 2j-1]) >> 3) + \quad (11)$$

【0157】

以下の文章では、(i, j)に位置する色差サンプルの対応するダウンサンプリングされた輝度サンプルのことを、ダウンサンプリングされた再構築された輝度サンプル $\text{rec}_L(i, j)$ と呼ぶ。たとえば、4:2:0サンプリングであるので、 $2N \times 2N$ 輝度ブロックは $N \times N$ 色差ブロックに対応する。ダウンサンプリングにより、 $2N \times 2N$ 輝度ブロックは $N \times N$ のダウンサンプリングされた輝度ブロックになる。この $N \times N$ のダウンサンプリングされた輝度ブロックは、 $\text{rec}_L(i, j)$ と呼ばれ、 $N \times N$ 色差ブロックに対応する。

10

【0158】

さらに、上の例は4:2:0サンプリングに関して説明されたが、本開示において説明される技法は、それに限定されない。たとえば、本開示において説明される技法は、4:2:2サンプリングにも適用可能であり得る。したがって、4:2:0に関する例は、理解を助けるために与えられたものにすぎない。

【0159】

さらに、いくつかの例では、本開示において説明される技法は、4:4:4サンプリングにも適用可能であり得る。たとえば、4:4:4サンプリングでは、色差ブロックは輝度ブロックに対して相対的にサブサンプリングされない。しかしながら、そのような例でも、色差ブロックの予測ブロックを決定することは可能であり得る。たとえば、輝度ブロックがフィルタリングされてよく、このフィルタリングされたブロックが色差ブロックの予測ブロックとして使用されてよい。これらの例では、輝度ブロックのダウンサンプリングは必要とされないことがある。より詳細に説明されるように、例示的な技法は、色差ブロックの位置に基づいて輝度ブロックのサンプルに適用されるフィルタの選択を記述する。輝度ブロックのサンプルに適用されるフィルタを選択するための技法は、4:4:4サンプリングなどの、LM予測のためにダウンサンプリングが必要ではない例に拡張され得る。そのような例では、フィルタは、4:4:4サンプリングが保たれるように、どのようなダウンサンプリングも含まないことがある。したがって、4:2:0サンプリングについての説明は例であり、本技法は4:4:4サンプリングにも適用可能である。

20

30

【0160】

たとえば、輝度ブロックをダウンサンプリングするために2タップフィルタまたは6タップフィルタのみを使用することに限定されるのではなく、ビデオコーデ(たとえば、ビデオエンコーデ20またはビデオデコーデ30)は、輝度ブロックをダウンサンプリングするために使用されるフィルタのセットからフィルタを決定し得る。ある例として、ビデオコーデはダウンサンプリングのために使用できる、ある数Xの異なるフィルタがあり得る。たとえば、1タップフィルタ、2タップフィルタ、3タップフィルタなどがあり得る。その上、各フィルタに対して、具体的なタップは異なり得る(たとえば、第1の2タップフィルタのために使用される輝度サンプルは、第2の2タップフィルタのために使用される輝度サンプルとは異なる)。本開示において説明される例のいくつかでは、フィルタのセットは2つのフィルタを含むが、輝度ブロックをダウンサンプリングするためにどのフィルタを適用するかをビデオコーデがそこから決定する3つ以上のフィルタが可能である。

40

【0161】

ビデオコーデは、どのフィルタを適用するかを決定するために様々な基準を使用し得る。一例として、ビデオコーデは、色差ブロックの位置に基づいて、フィルタのセットのうちのどのフィルタを適用するかを決定する。色差ブロックがピクチャ、CU、PU、またはTUの左の境界に接する(たとえば、ピクチャ、CU、PU、またはTUの左の境界が色差ブロックのエッジと同じである)場合、ビデオコーデは、左の境界に接する色差ブロックの色差サンプルに対応する輝度ブロックの輝度サンプルをダウンサンプリングするために第1のフ

50

フィルタを使用し得る。左の境界に接する色差ブロックのサンプルとは、境界の直上にあるサンプルを含む左の境界に最も近い色差ブロックのサンプルを指す。第1のフィルタは、境界に最も近いN個のサンプル(たとえば、境界に最も近いサンプル、そのサンプルの隣のサンプル、およびN個のそのようなサンプル)に適用され得る。

【0162】

いくつかの場合、ビデオコードは、左の境界に隣接する色差サンプルに対応するサンプルだけではなく、輝度ブロックのすべての輝度サンプルに対して第1のフィルタを適用し得る。しかしながら、本開示で説明する技法はそのように限定されない。すべての他の場合に、ビデオコードは、輝度ブロックをダウンサンプリングするために第2の異なるフィルタを使用し得る。

10

【0163】

たとえば、4:2:0サンプリングでは、4つの輝度サンプルは1つの色差サンプルに対応する。したがって、ビデオコードは、どの色差サンプルがどの輝度サンプルに対応するかを決定し得る。よりタップ数が大きいフィルタが使用されるとき、1つの色差サンプルは4つよりも多くの輝度サンプルに対応し得る。左の境界に接する(すぐ隣の、またはある数のサンプル内にある)色差サンプルに対応する輝度サンプルに対して、ビデオコードは、第1のフィルタに対応する輝度サンプルに適用して輝度ブロックをダウンサンプリングすることができ、左の境界に接しない(すぐ隣にない、またはある数のサンプル内にない)色差サンプルに対応する輝度サンプルに対して、ビデオコードは、第2のフィルタに対応する輝度サンプルに適用して輝度ブロックをダウンサンプリングし得る。

20

【0164】

いくつかの例では、第1のフィルタは、第2のフィルタよりも少数のタップ(たとえば、フィルタが及ぶサンプルの数)を含み得る。一例として、第1のフィルタは2タップフィルタであり、第2のフィルタは6タップフィルタである。この例では、ビデオコードは、色差ブロックの対応する色差サンプルが左の境界に接する場合には、式(6)の演算を実行して輝度ブロックのダウンサンプリングされた輝度サンプルを決定することができ、色差ブロックの対応する色差サンプルが左の境界に接しない場合には、式(10)の演算を実行して輝度ブロックのダウンサンプリングされた輝度サンプルを決定することができる。したがって、色差サンプルの対応するダウンサンプリングされた輝度サンプルの導出プロセスの間、ビデオコードは、CU/PU/TUの左のピクチャ境界または左の境界(すなわち、辺)に位置する色差ブロックの色差サンプルに対応する輝度ブロックの輝度サンプルに、CU、PU、またはTUの左のピクチャ境界もしくは左の境界にない色差サンプルに対応する輝度ブロックの他のサンプルに適用されるフィルタと比較して、異なるフィルタを適用することができる。左の境界にある色差サンプルとは、左の境界のすぐ隣の、または左の境界からいくつかの数のサンプル内にある色差サンプルを指す。

30

【0165】

異なるフィルタを使用することで、ビデオコードは利用可能なサンプル値を適切に使用することが可能になる。たとえば、ピクチャ、CU、PU、またはTUの左の境界における色差サンプルに対応する輝度サンプルに6タップフィルタを使用すると、ダウンサンプリングのために輝度ブロックの一部ではない輝度サンプル値をビデオコードが使用しなければならないことがあり、輝度サンプルの欠如に対処するためにビデオコードが何らかの追加の処理(たとえば、輝度ブロックの一部ではないサンプルの値を生成するために輝度サンプル値をパディングすること)を実行しなければならないことがある。しかしながら、左の境界において2タップフィルタを使用すると、ビデオコードはダウンサンプリングのために輝度ブロックの一部ではない輝度サンプル値を使用しなくてもよくなることもある。したがって、2タップおよび6タップのフィルタが説明されるが、輝度ブロックの一部ではない輝度サンプルを要求する必要をなくすこと(たとえば、左の境界に接する輝度サンプルをパディングする必要をなくすこと)を考慮して、ダウンサンプリングのための他のサイズのフィルタが可能であり得る。

40

【0166】

50

一例として、色差サンプルの対応するダウンサンプリングされた輝度サンプルの導出プロセスの間、ビデオコードは、左のピクチャ境界に位置する色差サンプルに対応する輝度サンプルに、左のピクチャ境界に位置しない色差サンプルに対応する他の輝度サンプルに適用されるフィルタと比較して、異なるフィルタを適用する。一例では、左のピクチャ境界における色差サンプルの対応するダウンサンプリングされた輝度サンプルを導出するためのフィルタの長さ(たとえば、タップ)(すなわち、フィルタが及ぶサンプルの数)は、左のピクチャ境界にない色差サンプルの対応するダウンサンプリングされた輝度サンプルを導出するためのフィルタの長さより小さい(たとえば、左の境界に対しては2タップ、およびすべての他の部分に対しては6タップ)。

【0167】

10

一例として、色差サンプルの対応するダウンサンプリングされた輝度サンプルの導出プロセスの間、ビデオコードは、左のCU境界に位置する色差サンプルの輝度サンプルに、現在のCU内の他の輝度サンプルに適用されるフィルタと比較して、異なるフィルタを適用する。一例では、左のCU境界における色差サンプルの対応するダウンサンプリングされた輝度サンプルを導出するためのフィルタの長さ(たとえば、タップ)(すなわち、フィルタが及ぶサンプルの数)は、左のCU境界にない色差サンプルの対応するダウンサンプリングされた輝度サンプルを導出するためのフィルタの長さより小さい(たとえば、左の境界に対しては2タップ、およびすべての他の部分に対しては6タップ)。

【0168】

20

一例として、色差サンプルの対応するダウンサンプリングされた輝度サンプルの導出プロセスの間、ビデオコードは、左のPU境界に位置する色差サンプルに、現在のPU内の他のサンプルに適用されるフィルタと比較して、異なるフィルタを適用する。一例では、左のPU境界における色差サンプルの対応するダウンサンプリングされた輝度サンプルを導出するためのフィルタの長さ(たとえば、タップ)(すなわち、フィルタが及ぶサンプルの数)は、左のPU境界にない色差サンプルの対応するダウンサンプリングされた輝度サンプルを導出するためのフィルタの長さより小さい(たとえば、左の境界に対しては2タップ、およびすべての他の部分に対しては6タップ)。

【0169】

30

一例として、色差サンプルの対応するダウンサンプリングされた輝度サンプルの導出プロセスの間、ビデオコードは、左のTU境界に位置する色差サンプルに、現在のTU内の他のサンプルに適用されるフィルタと比較して、異なるフィルタを適用し得る。一例では、左のTU境界における色差サンプルの対応するダウンサンプリングされた輝度サンプルを導出するためのフィルタの長さ(たとえば、タップ)(すなわち、フィルタが及ぶサンプルの数)は、左のTU境界にない色差サンプルの対応するダウンサンプリングされた輝度サンプルを導出するためのフィルタの長さより小さい(たとえば、左の境界に対しては2タップ、およびすべての他の部分に対しては6タップ)。

【0170】

40

いくつかの場合、同じピクチャの中に対応する輝度サンプルがないことがある。以下では、そのような状況に対処するためのいくつかの例示的な技法を説明する。たとえば、パディングを避けることはいくつかの場合には有益であり得るが、いくつかの事例では、パディングを避けることが可能ではないことがある。たとえば、いくつかの輝度サンプルが利用不可能であるので(たとえば、ピクチャから離れているので)、ビデオコードは、代わりにこれらの利用不可能なサンプルのサンプル値をパディングし、これらのパディングサンプル値を用いてダウンサンプリングを実行し得る(たとえば、利用可能な輝度サンプルのために実際の輝度サンプル値を使用し、利用不可能な輝度サンプルのためにパディングサンプル値を使用してダウンサンプリングし得る)。パディングサンプル値は、デフォルトの値(たとえば、 2^{bitdepth} 、ここでbitdepthは輝度成分のビット深度を示す)、ビデオエンコード20によって決定されビデオデコード30にシグナリングされる値、または情報のシグナリングを必要としない何らかの暗黙的な技法に基づいて決定される値であり得る。パディングサンプル値を加算することで、別個のフィルタの必要がなくなり得るので複雑

50

さが下がり得る。

【0171】

色差サンプルの対応するダウンサンプリングされた輝度サンプルの導出プロセスの間、輝度サンプルがピクチャの外側にあるとき、またはCU/PU/TUがダウンサンプリングプロセスに参与する必要があるとき、ビデオコードはまず、パディング動作を適用し、続いてダウンサンプリングプロセスを適用し得る。サンプルのパディングにおいて、ビデオコードは、画面上にないサンプルを、パディングサンプル値で置き換え得る。

【0172】

一例として、色差サンプルの対応するダウンサンプリングされた輝度サンプルの導出プロセスの間、ビデオコードは、現在のピクチャの外側に位置する輝度サンプル(たとえば、輝度サンプルのみ)をパディングし得る。すべての他の場所では、再構築されたサンプルが使用される。一例として、色差サンプルの対応するダウンサンプリングされた輝度サンプルの導出プロセスの間、ビデオコードは、現在のCUの外側に位置する輝度サンプルをパディングし得る。すべての他の場所では、再構築されたサンプルが使用される。一例として、色差サンプルの対応するダウンサンプリングされた輝度サンプルの導出プロセスの間、ビデオコードは、現在のPUの外側に位置する輝度サンプルをパディングし得る。すべての他の場所では、再構築されたサンプルが使用される。一例として、色差サンプルの対応するダウンサンプリングされた輝度サンプルの導出プロセスの間、ビデオコードは、現在のTUの外側に位置する輝度サンプルをパディングし得る。すべての他の場所では、再構築されたサンプルが使用される。パディングの上の例では、同じダウンサンプリングプロセスがすべての場所に適用される。

10

20

【0173】

LM予測モードにおいて使用される輝度が再構築されたサンプルの場所が現在のスライスまたは現在のタイルの外側に位置するとき、ビデオコードは、そのようなサンプルを利用不可能なものとしてマークし得る(たとえば、ビデオコードはそのようなサンプルを利用不可能なものとして決定し得る)。サンプルが利用不可能であるものとしてマークされるとき、ビデオコードは次の1つまたは複数を実行し得る。

【0174】

利用不可能なサンプルは、隣接輝度ブロックのためのダウンサンプリングプロセスにおいて使用される場合、隣接輝度ブロックのためのダウンサンプリングプロセスにおいては使用されない。代わりに、または加えて、フィルタは、他のサンプルのために使用されるフィルタと異なり得る。利用不可能なサンプルは、現在の輝度ブロックのためのダウンサンプリングプロセスにおいて使用される場合、現在の輝度ブロックのためのダウンサンプリングプロセスにおいては使用されない。代わりに、または加えて、フィルタは、他のサンプルのために使用されるフィルタと異なり得る。利用不可能なサンプルは利用可能なものとして再マークされるが、サンプル値はパディングされるサンプル値またはデフォルト値に修正される。代わりに、または加えて、フィルタは、他のサンプルのために使用されるフィルタと同じままにされる。一例では、デフォルト値はビット深度に依存する。別の例では、パディングは、利用可能であるものとしてマークされる左/右/上/下のサンプルからのものであり得る。

30

40

【0175】

一般に、別のタイルの中にある輝度サンプルに対して、ビデオコードは、タイル境界の外側の画素を利用不可能なものとしてマークし、ダウンサンプリングプロセスにそれらを含めなくてもよい。いくつかの例では、ビデオコードは、別のタイルの中の輝度サンプルを利用可能なものとしてマークするが、別のタイルの中のそのような輝度サンプルのためにパディングされた画素を使用してもよい。別の例として、ビデオコードは、別のタイルの中の輝度サンプルのために、サンプルを利用不可能なものとしてマークするのではなく、パディングされた「拡張された」値を使用し得る(たとえば、ビット深度に基づく2分の1の可能な値、よって8ビットでは128を使用する)。

【0176】

50

いくつかの例では、ビデオコードは、異なるフィルタを異なる色差成分(CbまたはCr)に適用し得る。いくつかの例では、LM予測モードが有効にされるとき、ダウンサンプリングフィルタの1つまたは複数のセットはさらに、シーケンスパラメータセット(SPS)、ピクチャパラメータセット(PPS)、またはスライスヘッダのいずれかにおいてシグナリングされ得る。代わりに、または加えて、補足強化情報(SEI)メッセージシンタックスが、ダウンサンプリングフィルタを記述するために導入される。代わりに、または加えて、さらに、デフォルトのダウンサンプリングフィルタ、たとえば6タップフィルタ[1, 2, 1; 1, 2, 1]が、シグナリングなしで定義される。代わりに、または加えて、1つのPU/CU/最大CUが、LM予測モードにおいて使用されるフィルタのインデックスをシグナリングし得る。代わりに、または加えて、フィルタタップの使用は、シグナリングせずにビデオデコーダ30によってオンザフライで導出され得る。フィルタをサポートするための他の方法もあり得る。

10

【0177】

一例では、さらに、 i が $(i+3)$ に等しいという制約が適用される。一例では、さらに、 i が0または3に等しい場合に i が $(i+2)$ に等しいという制約が適用される。一例では、この例示的な技法は、より大きなコーディングされたCU、たとえば 16×16 より大きなCUサイズだけに対して有効にされ得る。一例では、パラメータの1つまたは複数が0であるように制約される。

【0178】

その上、ビデオコードは、成分間残差予測のためにも上の技法のうちの1つまたは複数適用することができ、成分間残差予測において、ダウンサンプリングされた輝度残差が色差残差を予測するために使用される。この場合、ダウンサンプリングプロセスは、一例として、再構築された輝度残差に適用される。

20

【0179】

以下は、本開示において説明される技法がビデオコードによって実施され得る例示的な方式である。例示的な実施技法は限定するものと見なされるべきではない。

【0180】

以下は、左のピクチャ境界にある複数のサンプルに異なるダウンサンプリングプロセスを適用するための例である。現在の輝度ブロックのためのダウンサンプリングプロセスは次のように定義される。

30

- 色差サンプルがピクチャの左の境界に位置しない場合、6タップフィルタ、たとえば[1 2 1; 1 2 1]が対応するダウンサンプリングされた輝度サンプルを導出するために適用される。

$$\text{rec}_L(i, j) = (\text{Rec}_{\text{Lorig}}[2i, 2j] * 2 + \text{Rec}_{\text{Lorig}}[2i, 2j+1] + \text{Rec}_{\text{Lorig}}[2i, 2j-1] + \text{Rec}_{\text{Lorig}}[2i+1, 2j] * 2 + \text{Rec}_{\text{Lorig}}[2i+1, 2j+1] + \text{Rec}_{\text{Lorig}}[2i+1, 2j-1] + \text{offset0}) \gg 3 \quad (13)$$

- そうではなく、色差サンプルがピクチャの左の境界に位置する場合、2タップフィルタ、たとえば[1; 1]が対応するダウンサンプリングされた輝度サンプルを導出するために適用される。

$$\text{rec}_L(i, j) = (\text{Rec}_{\text{Lorig}}[2i, 2j] + \text{Rec}_{\text{Lorig}}[2i, 2j+1] + \text{offset1}) \gg 1 \quad (14)$$

40

【0181】

一例では、offset0とoffset1はともに0に等しく設定される。別の例では、offset0は4に等しく設定され、offset1は1に等しく設定される。

【0182】

HEVCでは、長方形のPUに対しても、正方形変換が常に適用される。たとえば、図16は、 $N \times N$ 変換を伴う $nR \times 2N$ 予測モードを示す概念図である。図16の例では、 $nR \times 2N$ 予測モードは、ブロックサイズが $2N \times 2N$ であるコーディングブロック100を、それぞれ、サイズが $0.5N \times 2N$ および $1.5N \times 2N$ である2つの予測ブロックへと区分する。しかしながら、図16の例では、変換ブロックサイズは $N \times N$ である。

【0183】

50

図17は、 $2N \times N$ 、 $2N \times nD$ 、および $2N \times nU$ 予測モードのための非正方形四分木(NSQT)を示す概念図である。図17では、レベル0の $2N \times 2N$ ブロックは、レベル1に位置する4つの $2N \times 0.5N$ ブロックへと分割され、レベル1のブロックはレベル2に位置する4つの $N \times 0.25N$ ブロックへとさらに分割される。図18は、 $N \times 2N$ 、 $nR \times 2N$ 、および $nL \times 2N$ 予測モードのためのNSQTを示す概念図である。図18では、レベル0の $2N \times 2N$ ブロックは、レベル1に位置する4つの $0.5N \times 2N$ ブロックへと分割され、レベル1のブロックはレベル2に位置する4つの $0.25N \times N$ ブロックへとさらに分割される。

【0184】

残差が2つの連結する予測ブロックの境界において不連続であり得ることを考慮すると、高周波変換係数が産生される可能性が高く、コーディング性能が影響を受けるであろう。本開示では、連結する予測ブロックは、4つの境界のうちの少なくとも1つを共有する予測ブロックである。したがって、Yuan他、「Non-Square Quadtree Transform Structure for HEVC」、2012 Picture Coding Symposium (PCS)、pp. 505 ~ 508、2012年5月7 ~ 9日、クラクフ、ポーランド(以後、「Yuan」)において、非正方形四分木変換(NSQT)構造が説明されている。

【0185】

NSQTでは、 $2N \times 0.5N$ および $0.5N \times 2N$ という2つの追加の変換ブロックサイズが追加される。本開示では、変換ブロックは $2N \times 0.5N$ および $0.5N \times 2N$ へと分割され、変換行列は、 $0.5N \times 0.5N$ および $2N \times 2N$ の変換行列を再使用することによって取得され得る。本開示では、変換行列は変換コアとも呼ばれることがある。Yuanでは、 $2N \times 0.5N$ および $0.5N \times 2N$ の変換

【0186】

上で言及されたように、ビデオコードは、サンプルを周波数領域に変換するために、またはその逆を行うために、変換を適用し得る。HEVCにおいて適用される変換の具体的なタイプは、2つのタイプの離散コサイン変換、すなわちDCT-IIおよび 4×4 DST-VIIである。Xin Zhao他、米国特許出願公開第2016/0219290 A1号は、インターコーディングされたブロックとイントラコーディングされたブロックの両方に対して、DCT-IIおよび 4×4 DST-VIIに加えてEnhanced Multiple Transform(EMT)方式を提案した。EMT方式は、HEVCにおける現在の変換以外の、DCT/離散サイン変換(DST)ファミリーからの複数の選択された変換を利用する。米国特許出願公開第2016/0219290号において新たに導入された変換行列は、DS

【0187】

米国特許出願公開第2016/0219290 A1号における提案されたEMTは、 64×64 より小さいCUに適用され、EMTが適用されるかどうかは、CU内のすべてのTUに対して、フラグ、すなわちEMTフラグを使用して、CUレベルで制御される。EMT対応CU内の各TUに対して、使用されるべき水平方向または垂直方向の変換は、選択された変換セットに対するインデックス、すなわちEMTインデックスによってシグナリングされる。各変換セットは、前述の変換行列から2つの変換を選択することによって形成される。

【0188】

イントラ予測残差に対して、変換セットが、X. Zhao他、「Video coding with rate-distortion optimized transform」、IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 1, pp. 138 ~ 151、2012年1月において説明されるように、イントラ予測モードに基づいて事前に定義されるので、各イントラ予測モードは固有の変換セットを有する。たとえば、1つの変換セットは{DCT-VIII, DST-VII}であり得る。水平方向の変換のための変換セットは、同じイントラ予測モードに対しても、垂直方向の変換のための変換セットとは異なり得ることに留意されたい。しかしながら、すべてのイントラ予測モードに対する異なる変換セットの総数、ならびに新しく導入される変換の数は限られている。しかしながら、インター予測残差に対して、1つだけの変換セットが、すべてのインターモードのために、および水平方向の変換と垂直方向の変換の両方のために使用される。

【0189】

各カメラの光源に対する露出は異なり得るので、マルチビュービデオコーディングにおける照明補償(IC)が、異なるビュー間の照明の不一致を補償するために使用される。通常、加重係数および/またはオフセットが、コーディングされるブロックと異なるビュー中の予測ブロックとの差分を補償するために使用される。照明補償は、ビュー間参照ピクチャから予測されるブロックのコーディング効率を改善するために導入された。したがって、照明補償は、ビュー間参照ピクチャによって予測されるブロックだけに適用され得る。

【0190】

Liu他、「3D-CE1.h related: Illumination Compensation for Inter-View Prediction」、Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11、第1回会合、ストックホルム、スウェーデン、2012年7月16～20日、文書JCT3V-A0086(以後、JCT3V-A0086)が、照明補償(IC)を説明する。JCT3V-A0086では、ICがビュー間予測のために有効にされる。さらに、JCT3V-A0086において説明されるように、ICプロセスは、現在のCUの隣接サンプルおよび参照ブロックの隣接サンプルに基づいてICパラメータを導出する。JCT3V-A0086では、ICは $2N \times 2N$ 区分モードのみに適用される。さらに、JCT3V-A0086では、AMVPモードに対して、ビュー間参照ピクチャから予測される各CUのために、1つのICフラグがシグナリングされる。マージモードでは、ビットを節約するために、PUのマージインデックスが0に等しくないときだけ、ICフラグがシグナリングされる。ICフラグは、ICがCUのために使用されるかどうかを示す。ICは、時間参照ピクチャだけから予測されるCUには適用されない。

【0191】

JCT3V-A0086において説明されるように、ビュー間予測において使用される線形ICモデルが式(6)に示される。

$$p(i, j) = a * r(i + dv_x, j + dv_y + b), \text{ただし}(i, j) \in PU_c \quad (15)$$

ここで、 PU_c は現在のPUであり、 (i, j) は PU_c の中の画素の座標であり、 (dv_x, dv_y) は PU_c の視差ベクトルであり、 $p(i, j)$ は PU_c の予測であり、 r は隣接ビューからの現在のPUの参照ピクチャである。 a および b は線形ICモデルのパラメータである。

【0192】

JCT3V-A0086において、図19に示されるような画素の2つのセットが、現在のPUのためのパラメータ a および b を推定するために使用される。画素の第1のセットは、現在のCU(すなわち、現在のPUを含むCU)の左の列および上の行の中の、利用可能な再構築された隣接画素を含む。画素の第2のセットは、現在のCUの参照ブロックの対応する隣接画素を含む。現在のPUの視差ベクトルは、現在のCUの参照ブロックを見つけるために使用される。

【0193】

図19は、ICモデルにおけるパラメータを推定するために使用される隣接画素を示す。具体的には、図19は現在のCU110および参照ブロック112を含む。図19の各々のそれぞれの正方形はそれぞれのサンプルに対応する。したがって、現在のCU110および参照ブロック112は各々64個のサンプルを含む。現在のCU110の隣の円を囲む正方形は、現在のCU110の隣接サンプル(すなわち、 Rec_{neig})に対応する。参照ブロックCU112の隣の円を囲む正方形は、隣接ブロック112の隣接サンプル(すなわち、 $Rec_{refneig}$)に対応する。本開示の他の箇所では説明されるように、ビデオコードは、 Rec_{neig} および $Ref_{refneig}$ を使用してICのためのパラメータを推定し得る。

【0194】

さらに、JCT3V-A0086において説明されるように、 Rec_{neig} を、現在のCUにより使用される隣接画素セットを表すものとする。 $Rec_{refneig}$ を、現在のCUの参照ブロックにより使用される隣接画素セットを表すものとする。現在のCUのサイズおよび現在のCUの参照ブロックのサイズがともに $N \times N$ に等しいものとする。 $2N$ を、 Rec_{neig} および $Rec_{refneig}$ の中の画素の数を表すものとする。そうすると、 a および b は次のように計算され得る。

【0195】

【数 2 1】

$$a = \frac{2N \cdot \sum_{i=0}^{2N-1} \text{Rec}_{neig}(i) \cdot \text{Rec}_{refneig}(i) - \sum_{i=0}^{2N-1} \text{Rec}_{neig}(i) \cdot \sum_{i=0}^{2N-1} \text{Rec}_{refneig}(i)}{2N \cdot \sum_{i=0}^{2N-1} \text{Rec}_{refneig}(i) \cdot \text{Rec}_{refneig}(i) - \left(\sum_{i=0}^{2N-1} \text{Rec}_{refneig}(i) \right)^2} \quad (16)$$

$$b = \frac{\sum_{i=0}^{2N-1} \text{Rec}_{neig}(i) - a \cdot \sum_{i=0}^{2N-1} \text{Rec}_{refneig}(i)}{2N} \quad (17)$$

10

【0196】

いくつかの場合、aだけが線形モデルにおいて使用されてbは常に0に等しく設定され、または、bだけが使用されてaは常に1に等しく設定される。

【0197】

VCEG-AZ06において、局所照明補償(LIC)が、各々のインターモードコーディングされたCUに対して適応的に有効または無効にされる。VCEG-AZ06において、LICは、スケーリング係数aおよびオフセットbを使用する、照明変化のための線形モデルに基づく。図20は、VCEG-AZ06において説明されるようなICパラメータを導出するために使用される例示的な隣接サンプルを示す概念図である。

20

【0198】

VCEG-AZ06において、LICがCUに適用されるとき、CUに属する各PU/サブPUに対して、ビデオコーダは、CUのサブサンプリングされる(2:1サブサンプリング)隣接サンプルおよび参照ピクチャの中に対応する画素(現在のPU/サブPUの動き情報によって特定される)を使用する方法で、LICパラメータを導出する。N×Nに等しいサイズのCUでは、式(16)および(17)において使用される境界画素の総数は2NではなくNである。例が図20に示されている。したがって、図20は、現在のCU116の参照ブロック114が現在のPUの視差ベクトルを使用することによって発見される、照明補償モデルにおけるパラメータを推定するために使用される例示的な隣接画素を示す概念図である。VCEG-AZ06において、ICパラメータは、各予測方向に対して別々に導出され適用される。上述の隣接サンプルに基づいてパラメータaおよびbを導出するために、ビデオコーダは最小二乗誤差法を利用し得る。

30

【0199】

HEVCにおける現在のRQT設計、ならびにNSQTおよびICなどの他の技法は、以下の欠点を有し得る。たとえば、NSQTが使用されるか、HEVCの変換木が使用されるかにかかわらず、PU情報を考慮しないと最適ではないことがある四分木構造が常に利用される。しかしながら、HEVCは、イントラ予測モードに対しては正方形PUしかサポートしない。

【0200】

40

JCTVC-G135において行われるように、イントラモードに2N×NおよびN×2Nの区分を導入すると、以下の問題があり得る。まず、AMPが許可されない。第2に、高いコーディング効率を達成するために変換木構造をどのように定義すべきかが研究されていない。第3に、LM予測モードが正方形PUだけと使用されてきたので、非正方形PUについて、LM予測モードにおいて使用されるパラメータおよびをどのように導出すべきかが知られていない。第4に、これまでの技法では、係数を正方形になるように並べ替えなければならない、これは隣接する係数の間の相関を少なくすることがある。さらに、現在のEMT設計には、EMTがCUレベルで制御されるという問題がある。しかしながら、CUレベルでEMTを制御することは、CUの中の各PUの残差特性(たとえば、分布)が異なる場合、効率的ではない。

【0201】

50

上で言及された問題を解決するために、本開示は以下の技法を提案する。以下の箇条書きにされる技法は、個別に適用され得る。代わりに、それらの任意の組合せが適用され得る。以下の説明では、CUサイズは $M \times M$ により表され、PUサイズは $K \times L$ により表され、 K と L はともに M 以内である。

【0202】

本開示の第1の例示的な技法によれば、変換木が四分木に限定されないことが提案される。たとえば、変換四分木と変換二分木が組み合わされ得る。すなわち、少なくともある変換深度に対して、1つのTUが2つのより小さいTUまたは4つのより小さいTUへと分割され得る。本開示では、変換木の各々のそれぞれのノードに対して、それぞれのノードのそれぞれの変換深度は、それぞれのノードと変換木のルートノードとの間の、変換木の中のノードの数を指す。TUを2つのTUへと分割するか、または4つのTUへと分割するかという柔軟性により、TU境界をPU境界と揃えるように変換木を構築するビデオエンコーダ20の能力が向上し得る。TU境界をPU境界と揃えることで、圧縮性能が上がり得る。

10

【0203】

したがって、この例では、ビデオエンコーダ20は、木構造に基づいてCUのTUへとビデオデータのCUを区分し得る。この例では、木構造のルートノードはCUのコーディングブロックに対応する。さらに、この例では、木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応する。この例では、木構造のリーフノードはCUのTUに対応する。この例では、木構造の中の少なくとも1つのノードが、木構造の中で丁度2つの子ノードを有する。いくつかの事例では、木構造の中の少なくとも1つのノードが、木構造の中で丁度4つの子ノードを有し得る。この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、CUのTUのうちの1つまたは複数を表すデータを含め得る。

20

【0204】

対応する例では、ビデオデコーダ30は、木構造に基づいてCUのTUへとCUが区分されると決定し得る。この例では、木構造のルートノードはCUのコーディングブロックに対応する。さらに、この例では、木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応する。この例では、木構造のリーフノードはCUのTUに対応する。この例では、木構造の中の少なくとも1つのノードは木構造の中で丁度2つの子ノードを有し、木構造の中の少なくとも1つのノードは木構造の中で丁度4つの子ノードを有する。この例では、ビデオデコーダ30は、CUのTUのうちの少なくとも1つに対するデータに基づいて、CUのコーディングブロックを再構築し得る。

30

【0205】

さらに、変換木が四分木に限定されない(すなわち、すべての非リーフノードが4つの子ノードを有するような木であることが求められない)例では、0に等しい変換深度に対して、サイズが $M \times M$ に等しい正方形変換が適用される。1に等しい変換深度に対して、変換は(PUの数に応じて)2つまたは4つへと分割され、変換サイズは $K \times L$ に等しい。残りの変換深度に対して、四分木構造が依然として適用され、ここで1つのTUが4つのより小さいTUへと分割され、すなわち、2に等しい変換深度に対して、変換サイズは $K/2 \times L/2$ に設定される。例が図21に与えられる。変換深度1において2または4へと変換の分割を制限する1つの理由は、変換サイズをPUサイズと揃えるためであり、たとえば、PUサイズが $2N \times N$ または $N \times 2N$ である場合、2への分割が好ましいことがある。PUが $N \times N$ の区分である場合、4ウェイの変換分割がより良好な結果を生むことがある。別の理由では、対応する変換行列が知られていなかった場合、たとえばAMPが使用される場合、1つの 16×16 のCUが 4×16 および 12×16 のPUへと分割され得るが、 12×12 の変換は定義されないので、4への分割がこの場合には使用され得る。

40

【0206】

図21は、 $2N \times N$ に等しい区分サイズに対する例示的な変換構造を示す概念図である。図21および以下の図では、破線は次の変換深度に対する分割情報を示す。具体的には、図21

50

では、変換ブロック130はCUのコーディングブロックと同じサイズを有する。変換ブロック130は、変換ブロック132および134へと区分される。さらに、図21の例では、変換ブロック132は、変換ブロック136、137、138、および139へと区分される。変換ブロック134は、変換ブロック140、141、142、および143へと区分される。したがって、図21に示されるように、ルートノードは2個の子ノードを有し得るが、他の変換深度にあるノードは、0または4個の子ノードを有することが求められ得る。

【0207】

CUの変換木が四分木であることに制約されないいくつかの例では、二分木または四分木のいずれかが適用される。ビデオコーデは、CUの中のPUの数に基づいて、二分木が適用されるか四分木が適用されるかを決定し得る。たとえば、2つのPUがあるとき、ビデオコーデは二分変換木を利用する。CUが4つのPUを有する場合、ビデオコーデは、四分木構造を使用してCUをTUに区分し得る。一例では、二分木と四分木のいずれかを選択する方法は、1などのいくつかの変換深度にのみ適用される。

10

【0208】

したがって、この例では、ビデオエンコーダ20は、木構造に基づいてCUのTUへとビデオデータのCUを区分し得る。この例では、木構造のルートノードはCUのコーディングブロックに対応し、木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応し、木構造のリーフノードはCUのTUに対応し、CUは1つまたは複数のPUを有する。さらに、この例では、CUのPUの数に応じて、以下のうちの1つだけが適用される。すなわち、木構造の中の各ノードが木構造の中で丁度2つの子ノードを有し、または、木構造の中の各ノードが木構造の中で丁度4つの子ノードを有する。この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、CUのTUのうちの1つまたは複数を表すデータを含め得る。

20

【0209】

対応する例では、ビデオデコーダ30は、木構造に基づいてCUのTUへとビデオデータのCUが区分されると決定し得る。この例では、木構造のルートノードはCUのコーディングブロックに対応し、木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応し、木構造のリーフノードはCUのTUに対応し、CUは1つまたは複数のPUを有する。この例では、CUのPUの数に応じて、以下のうちの1つだけが適用される。すなわち、木構造の中の各ノードが木構造の中で丁度2つの子ノードを有し、または、木構造の中の各ノードが木構造の中で丁度4つの子ノードを有する。さらに、この例では、ビデオデコーダ30は、CUのTUのうちの少なくとも1つに対するデータに基づいて、CUのコーディングブロックを再構築し得る。

30

【0210】

変換木が四分木であることに制約されないいくつかの例では、二分木または四分木のいずれかの分割方法がシグナリングされる。たとえば、ビデオエンコーダ20は、CUが二分木に従ってTUへと区分されるか、四分木に従ってTUへと区分されるかを示す、シンタックス要素を表すデータをビットストリームに含め得る。この例では、ビデオデコーダ30は、ビットストリームの中のデータに基づいて、シンタックス要素の値を決定し得る。さらに、この例では、ビデオデコーダ30は、CUが二分木に従ってTUへと区分されるか、四分木に従ってTUへと区分されるかを、そのシンタックス要素の値に基づいて決定し得る。

40

【0211】

代わりに、さらに、シグナリングはいくつかのPU区分に対してスキップされ得る。言い換えると、ビデオエンコーダ20は、ブロックがPUへとどのように分割されるかに基づいて、そのブロックに対する変換木分割のシグナリングをスキップし得る。たとえば、一例では、 $2N \times N$ に等しいPU区分に対して、二分木が常に使用されるので、対応する変換木において、二分木分割が使用されるか四分木分割が使用されるかをシグナリングする必要はない。

【0212】

50

本開示の第2の技法によれば、ある変換深度において、変換サイズが長方形PUのためのPUサイズに等しいことが提案される。本開示では、変換サイズはTUの変換ブロックのサイズを指す。したがって、この例では、CUの変換木の特定の深度におけるノードに対応する変換ブロックは、CUのPUの予測ブロックと同じサイズを有する。前に論じられたように、TU境界をPU境界と揃えることで、圧縮性能が改善し得る。

【0213】

第2の技法の一例では、上の方法はインターコーディングされたCUにのみ適用される。言い換えると、ビデオコーディング規格は、インターコーディングされたCUに対するPUと変換サイズが確実に等しくなることをビデオエンコーダ20に求め得るが、この要件はイントラコーディングされたCUには適用されない。

10

【0214】

さらに、いくつかの例では、ビデオエンコーダ20は、3つの色成分(たとえば、Y、Cb、およびCr)に対して少なくとも1つの0ではない係数が存在するかどうかを示すために、各PUのために1つのフラグをシグナリングし得る。上で言及されたように、本開示の第2の技法は、CUのTUのサイズが、変換木において特定の深度にあるCUのPUのサイズに等しいことを求める。したがって、その特定の深度において、変換木は、CUの各々のそれぞれのPUのためのそれぞれの変換木ノードを含む。特定の深度における変換木の各々のそれぞれの変換木ノードに対して、それぞれの変換木ノードは、対応するPUの輝度予測ブロックおよび色差予測ブロックと同じサイズおよび形状を有する、輝度変換ブロックおよび色差変換ブロックに対応する。したがって、エンコーダ20は、対応するPUにおいて情報をシグナリングすることによって、その特定の深度における変換木ノード(およびその特定の深度における変換木ノードの子孫の変換木ノード)についての情報をシグナリングし得る。たとえば、ビデオエンコーダ20は、PUのための第1のシンタックス要素、PUのための第2のシンタックス要素、およびPUのための第3のシンタックス要素を、ビットストリームにおいてシグナリングし得る。この例では、PUのための第1のシンタックス要素は、対応する変換木ノードまたはその子孫の変換木ノードの輝度係数ブロックの中に0ではない変換係数が存在するかどうかを示し、PUのための第2のシンタックス要素は、対応する変換木ノードまたはその子孫の変換木ノードのCb係数ブロックの中に0ではない変換係数が存在するかどうかを示し、PUのための第3のシンタックス要素は、対応する変換木ノードまたはその子孫の変換木ノードのCr係数ブロックの中に0ではない変換係数が存在するかどうかを示す。

20

30

【0215】

従来の技法では、長方形PUは、インター予測されたCUにしか許可されていなかった。しかしながら、本開示の第2の技法のいくつかの例では、長方形PU($2N \times N$ 、 $N \times 2N$ など)がイントラコーディングされたCUに導入されるとき、上の方法(すなわち、変換サイズが特定の変換深度におけるPUサイズに等しいことを必要とする)も適用される。

【0216】

第3の技法によれば、1つのTUは複数のより小さいTUへと分割され得るが、より小さいTUのサイズは異なり得る。言い換えると、ビデオコーダは、TUを2つの異なるサイズの子TUへと分割し得る。いくつかの事例では、TUを2つ以上の異なるサイズの子TUへと分割することは、AMPが有効にされる場合にビデオコーディング性能を向上させることができ、それは、2つ以上の異なるサイズの子TUへとTUを分割することで、子TUの境界がPUの境界とより揃うようになり得るからである。本開示の他の箇所で説明されるように、TUの境界をPUの境界と揃えることで、予測ブロック間の境界における不連続性と関連付けられる高周波変換係数の発生が減り、したがって、圧縮効率が向上し得る。たとえば、ブロック(たとえば、CU)が 12×16 のPUを有する場合、 12×16 のPUに対応するブロックの区分は、2つの 8×8 のTUおよび2つの 4×4 のTUへと、または2つの 8×8 のTUおよび1つの 4×16 のTUへと分割され得る。

40

【0217】

第3の技法の一例では、AMPモードが1つのCUに対して有効にされるとき、より大きなPU

50

に対する変換木は、小さいPUに等しい1つの部分と、別のPUとしての残りとを伴う、2つの部分に分割され得る。例が図22に与えられる。図22は、本開示の技法による、 $N \times N/4(U)$ に等しい区分サイズのための変換構造を示す概念図である。図22の例では、変換ブロック150は、変換ブロック152および154へと区分される。さらに、図22の例では、変換ブロック152は、変換ブロック156、158、160、および162へと区分される。変換ブロック154は、変換ブロック164および166へと区分される。変換深度2における右側の分岐は、2つの分割変換サイズが異なることを示す。すなわち、変換ブロック164および変換ブロック166は異なるサイズを有する。

【0218】

第3の技法のいくつかの例では、TUの非対称な分割は、2つのPUサイズが異なる、または、1つのCUが複数のPUを含みそれらのPUのうちの少なくとも2つが異なるサイズを有する、AMPのケースにのみ適用可能である。言い換えると、ビデオコーデは、TUを含むCUが異なるサイズのPUへと分割される場合、TUを異なるサイズの子TUへと分割するだけであり得る。

10

【0219】

したがって、TUが複数の異なるサイズのTUへと分割され得る例では、ビデオエンコーダ20は、木構造に基づいてCUのTUへとCUを区分し得る。この例では、木構造のルートノードはCUのコーディングブロックに対応する。さらに、この例では、木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応する。この例では、木構造のリーフノードはCUのTUに対応する。この例では、木構造のうちの少なくとも1つのノードの子ノードは、異なるサイズのブロックに対応する。さらに、この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、CUのTUのうちの1つまたは複数を表すデータを含め得る。

20

【0220】

対応する例では、ビデオデコーダ30は、木構造に基づいてCUのTUへとCUが区分されると決定し得る。この例では、木構造のルートノードはCUのコーディングブロックに対応する。さらに、この例では、木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応する。この例では、木構造のリーフノードはCUのTUに対応する。この例では、木構造のうちの少なくとも1つのノードの子ノードは、異なるサイズのブロックに対応する。さらに、この例では、ビデオデコーダ30は、CUのTUのうちの少なくとも1つに対するデータに基づいて、CUのコーディングブロックを再構築し得る。

30

【0221】

本開示の第4の技法によれば、変換の分割は、CUにおいて同じ方向(垂直または水平)に沿わずに行われることが許可される。言い換えると、CUの変換木は、水平に分割された変換ブロックと垂直に分割された変換ブロックとを含み得る。変換ブロックの水平の分割と垂直の分割の両方を許可することで、CUのTUの境界がCUのPUの境界とより揃うようになり得る。本開示の他の箇所でも論じられるように、CUのTUの境界をCUのPUの境界と揃えることで、予測ブロック間の境界における不連続性と関連付けられる高周波変換係数の発生が減り、したがって、圧縮効率が向上し得る。第4の技法の一例では、CUのTUの水平の分割と垂直の分割の両方の使用は、いくつかの区分モード、たとえばAMPにのみ適用可能である。

40

【0222】

図23は、本開示の技法による、 $N \times N/4(U)$ に等しい区分サイズのための変換構造を示す概念図である。図23の例では、CU区分は水平方向に沿っており、TU区分は水平方向および/または垂直方向のいずれかからであり得る。具体的には、TU180はTU182およびTU184へと水平に分割される。TU182は、TU186、188、190、および192へと分割される。TU184は、TU194、196、および198へと水平および垂直に分割される。

【0223】

50

CUにおける異なる方向に沿った変換ブロックの分割が許可される例では、ビデオエンコーダ20は、木構造に基づいてCUのTUへとCUを区分し得る。この例では、木構造のルートノードはCUのコーディングブロックに対応する。さらに、この例では、木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応する。この例では、木構造のリーフノードはCUのTUに対応する。この例では、木構造の中の第1のノードは丁度2つの子ノードを有し、第1のノードの子ノードに対応するブロック間の境界は垂直である。加えて、この例では、木構造の中の第2のノードは丁度2つの子ノードを有し、第2のノードの子ノードに対応するブロック間の境界は水平である。この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、CUのTUのうちの1つまたは複数を表すデータを含め得る。

10

【0224】

同様に、ビデオデコーダ30は、木構造に基づいてCUのTUへとCUが区分されると決定し得る。この例では、木構造のルートノードはCUのコーディングブロックに対応し、木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応し、木構造のリーフノードはCUのTUに対応する。さらに、この例では、木構造の中の第1のノードは丁度2つの子ノードを有し、第1のノードの子ノードに対応するブロック間の境界は垂直である。この例では、木構造の中の第2のノードは丁度2つの子ノードを有し、第2のノードの子ノードに対応するブロック間の境界は水平である。この例では、ビデオデコーダ30は、CUのTUのうちの少なくとも1

20

【0225】

本開示の第5の技法によれば、1つのCUがイントラPUとインターPUの両方を含むことがあり、これは以下の説明ではcomb_modeと呼ばれる。いくつかの事例では、comb_modeの使用は、CUの予測ブロックの精度を向上させ得るので、最終的には圧縮性能の向上につながり得る。CUのPUの予測ブロックの精度は、PUの予測ブロックの対応するサンプルとCUのコーディングブロックのサンプルとの差の尺度である。

【0226】

したがって、第5の技法によれば、ビデオエンコーダ20は、イントラ予測を実行してCUの第1のPUの第1の予測ブロックを取得し得る。加えて、この例では、ビデオエンコーダ20は、インター予測を実行して同じCUの第2のPUの第2の予測ブロックを取得し得る。この例では、ビデオエンコーダ20は、第1の予測ブロックおよび第2の予測ブロックに基づいて、CUの残差データを取得し得る。さらに、この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、CUの残差データを表すデータを含め得る。

30

【0227】

同様に、第5の技法によれば、ビデオデコーダ30は、イントラ予測を実行してCUの第1のPUの第1の予測ブロックを取得し得る。この例では、ビデオデコーダ30は、インター予測を実行して同じCUの第2のPUの第2の予測ブロックを取得し得る。さらに、この例では、ビデオデコーダ30は、第1の予測ブロックおよび第2の予測ブロックに基づいて、CUのコーディングブロックを再構築し得る。

40

【0228】

1つのCUがcomb_modeでコーディングされ、CUが、 $N \times 2N$ 区分モードなどにおいて、垂直方向に2つのPUへと分割されるとき、ビデオコーダは、CUの変換木の変換深度を次のように決定し得る。左のPUがイントラコーディングされたPUである場合、変換深度は0以上であり得る。言い換えると、左のPUがイントラコーディングされることに基づいて、CUの変換木の深度は、0以上であることが許容される。したがって、CUの変換木の深度が0に等しい事例では、TUサイズはCUサイズに等しいことがあり、1つのTUは2つのPUをカバーすることがあり、すなわちPUの境界にまたがる可能性がある。それ以外の(左のPUがインターコーディングされたPUである)場合、変換深度は1以上に制約される。言い換えると、CUの変換

50

木の深度は1以上であり得るが、0に等しくはない。この例では、左のPUがインターコーディングされたPUであるとき、TUサイズはPUサイズよりも大きくてはならない。

【0229】

したがって、この例では、ビデオエンコーダ20は、ビデオコーディング規格に準拠するビットストリームを生成し得る。この例では、CUが第1のPUおよび第2のPUへと垂直な境界に沿って分割され、CUの左のPUがイントラコーディングされたPUであることに基づいて、ビデオコーディング規格は、CUのTUが第1のPUと第2のPUの両方をカバーすることを許容する。同様の例では、ビデオデコーダ30は、ビデオデータの符号化された表現を備えるビットストリームを取得し得る。この例では、ビットストリームは、CUが第1のPUおよび第2のPUへと垂直な境界に沿って分割され、CUの左のPUがイントラコーディングされたPUであることに基づいて、CUのTUが第1のPUと第2のPUの両方をカバーすることを許容する、ビデオコーディング規格に準拠し得る。ビデオエンコーダ20とビデオデコーダ30の両方の例において、ビデオコーディング規格は、CUが第1のPUおよび第2のPUへと垂直な境界に沿って分割されることと、CUの左のPUがインターコーディングされたPUであることに基づいて、CUのTUのTUサイズが第1のPUまたは第2のPUのサイズより大きくないことを要求する制約を設け得る。

10

【0230】

さらに、 $2N \times N$ 区分モードが使用されるときなどの、1つのCUが水平方向に2つのPUへと分割されるとき、ビデオコーダは、CUがイントラPUとインターPUの両方を含むときに使用される変換深度を次のように決定し得る。上のPUがイントラコーディングされたPUである場合、変換深度は0以上であり得る。言い換えると、CUの変換木の深度は0以上である。この例では、上のPUは水平に分割されたCUの上側PUである。変換深度が0である事例では、TUサイズはCUサイズに等しく、1つのTUは2つのPUをカバーする。それ以外の(すなわち、上のPUがインターコーディングされたPUである)場合、変換深度は1以上に制約される。言い換えると、CUの変換木の深度は1以上であり、0であってはならない。この例では、上のPUがインターコーディングされたPUであるとき、TUサイズはPUサイズよりも大きくてはならない。

20

【0231】

したがって、この例では、ビデオエンコーダ20は、ビデオコーディング規格に準拠するビットストリームを生成し得る。この例では、CUが第1のPUおよび第2のPUへと水平な境界に沿って分割され、CUの上のPUがイントラコーディングされたPUであることに基づいて、ビデオコーディング規格は、CUのTUが第1のPUと第2のPUの両方をカバーすることを許容する。その上、いくつかの例では、ビットストリームは、CUが第1のPUおよび第2のPUへと水平な境界に沿って分割されることと、CUの上のPUがインターコーディングされたPUであることに基づいて、CUのTUのTUサイズが第1のPUまたは第2のPUのサイズより大きくないことを要求する制約を設けるビデオコーディング規格に準拠する。

30

【0232】

同様の例では、ビデオデコーダ30は、ビデオデータの符号化された表現を備えるビットストリームを取得し得る。この例では、ビットストリームは、CUが第1のPUおよび第2のPUへと水平な境界に沿って分割され、CUの上のPUがイントラコーディングされたPUであるとき、CUのTUが第1のPUと第2のPUの両方をカバーすることを許容する、ビデオコーディング規格に準拠する。その上、いくつかの例では、ビデオデコーダ30は、CUが第1のPUおよび第2のPUへと水平な境界に沿って分割され、CUの上のPUがインターコーディングされたPUであるときに、CUのTUのTUサイズが第1のPUまたは第2のPUのサイズより大きくないことを要求する制約を設けるビデオコーディング規格に準拠するビットストリームを取得する。

40

【0233】

いくつかの例では、1つのCUがcomb_modeでコーディングされるとき、TUがPUの境界にまたがってはならないという制約が追加される。この制約は、TUがPUの境界にまたがり得る場合のレートひずみコストを確認する必要がなくなるので、エンコーダの複雑さを下げることができる。したがって、この例では、ビデオエンコーダ20は、CUがイントラコーディ

50

ングされたPUおよびインターコーディングされたPUを有することに基づいて、CUのTUがCUのPU境界をまたがないことを要求する制約を設ける、ビデオコーディング規格に準拠するビットストリームを生成し得る。同様の例では、ビデオデコーダ30は、ビデオデータの符号化された表現を備えるビットストリームを取得し得る。この例では、ビットストリームは、CUがイントラコーディングされたPUおよびインターコーディングされたPUを有することに基づいて、CUのTUがCUのPU境界をまたがないことを要求する制約を設ける、ビデオコーディング規格に準拠する。

【0234】

さらに、comb_modeを伴ういくつかの例では、comb_modeが 8×8 などのあるサイズより大きい(そのサイズを含まない)CUにのみ適用されるように制約される。より小さいブロックに対しては、comb_modeがCUに適用されるかどうかをシグナリングするビットを増やすことは、comb_modeによりもたらされるレートひずみコストの低減とつり合いがとれないことがあることに留意されたい。したがって、ある小さいサイズに対しては、comb_modeは追加のシグナリングなしで常に無効にされ得る。本開示では、ある制約が、ビデオエンコーダが何らかの活動を実行すること、またはビットストリームを何らかの方法で生成することを妨げることがある。たとえば、ビデオエンコーダ20は、イントラコーディングされたPUとインターコーディングされたPUの両方を有するために特定のサイズより小さいCUが許可されないことを要求する制約を設ける、ビデオコーディング規格に準拠するビットストリームを生成し得る。同様の例では、ビデオデコーダ30は、ビデオデータの符号化された表現を備えるビットストリームを取得し得る。この例では、ビットストリームは、イントラコーディングされたPUとインターコーディングされたPUの両方を有するために特定のサイズより小さいCUが許可されないことを要求する制約を設ける、ビデオコーディング規格に準拠する。

【0235】

comb_modeを伴ういくつかの例では、1つの 8×8 のCUは、1つの 8×4 イントラPUおよび1つの 8×4 インターPUを用いて、または1つの 4×8 イントラPUおよび1つの 4×8 インターPUを用いて、comb_modeでコーディングされ得る。この例では、この 8×8 のCUの対応する 4×4 色差ブロック(4:2:0カラーフォーマットにおける)は、インターコーディングされた輝度PUのインター予測モードのみを使用して、もしくは、イントラコーディングされた輝度PUのイントラ予測モードのみを使用してコーディングされることが可能であり、または、 4×4 色差ブロックはさらに、それに対応して輝度PU区分に基づいて2つの 4×2 または 2×4 ブロックとして区分され、2つの 4×2 または 2×4 の各々は対応する輝度予測モードによって予測され、 4×4 残差ブロックが生成され、 2×2 変換の導入を避けるために 4×4 変換が生成された44個の残差ブロックに対して実行される。 2×2 変換の導入は不必要に複雑さを上げることがある。

【0236】

したがって、上の例では、ビデオエンコーダ20は、イントラ予測を実行してビデオデータのCUの第1のPUの第1の予測ブロックを取得し得る。加えて、この例では、ビデオエンコーダ20は、インター予測を実行して同じCUの第2のPUの第2の予測ブロックを取得し得る。この例では、CUのサイズは $2N \times 2N$ であり、第1のPUのサイズは $2N \times N$ であり、第2のPUのサイズは $N \times 2N$ であり、または、第1のPUのサイズは $N \times 2N$ であり、第2のPUのサイズは $2N \times N$ である。さらに、この例では、CUは4:2:0カラーフォーマットを使用してコーディングされる。この例では、第1のPUの第1の予測ブロックは、第1のPUの輝度予測ブロックである。この例では、第2のPUの第2の予測ブロックは、第2のPUの輝度予測ブロックである。この例では、ビデオエンコーダ20は、インター予測のみを使用して、サイズ $N \times N$ の色差予測ブロックである第3の予測ブロックを取得する。この例では、ビデオエンコーダ20は、第1、第2、および第3の予測ブロックに基づいて、CUの残差データを取得する。同様の例は、第1、第2、および第3の予測ブロックに基づいてCUの残差データを取得するビデオエンコーダ20の代わりに、イントラ予測のみを使用してイントラ予測の代わりに第3の予測ブロックを取得するビデオエンコーダ20を代わりに用いる。

【 0 2 3 7 】

その上、対応する例では、ビデオデコーダ30は、イントラ予測を実行してビデオデータのCUの第1のPUの第1の予測ブロックを取得し得る。この例では、ビデオデコーダ30は、インター予測を実行して同じCUの第2のPUの第2の予測ブロックを取得し得る。この例では、CUのサイズは $2N \times 2N$ であり、第1のPUのサイズは $2N \times N$ であり、第2のPUのサイズは $N \times 2N$ であり、または、第1のPUのサイズは $N \times 2N$ であり、第2のPUのサイズは $2N \times N$ であり、CUは4:2:0カラーフォーマットを使用してコーディングされる。さらに、この例では、第1のPUの第1の予測ブロックは第1のPUの輝度予測ブロックであり、第2のPUの第2の予測ブロックは第2のPUの輝度予測ブロックである。この例では、ビデオデコーダ30は、インター予測のみを使用して、サイズ $N \times N$ の色差予測ブロックである第3の予測ブロックを取得する。さらに、この例では、ビデオデコーダ30は、第1、第2、および第3の予測ブロックに基づいて、CUのコーディングブロックを再構築し得る。同様の例は、イントラ予測のみを使用してイントラ予測の代わりに第3の予測ブロックを取得するビデオデコーダ30の異なる構成を、代わりに用いる。

10

【 0 2 3 8 】

上で言及されたように、いくつかの例では、1つの 8×8 のCUは、1つの 8×4 イントラPUおよび1つの 8×4 インターPUを用いて、または1つの 4×8 イントラPUおよび1つの 4×8 インターPUを用いて、comb_modeでコーディングされることが可能であり、この 8×8 のCUの対応する 4×4 色差ブロックは、インターコーディングされた輝度PUのインター予測モードのみを使用してコーディングされることが可能であり、ビデオコーダは、それに対応して輝度PU区分に基づいて2つの 4×2 または 2×4 ブロックとして 4×4 色差ブロックを区分することができ、ビデオコーダは対応する輝度予測モードによって2つの 4×2 または 2×4 の各々を予測し、ビデオコーダは 4×4 残差ブロックを生成し、生成された 4×4 残差ブロックに対して 4×4 変換を実行する。

20

【 0 2 3 9 】

したがって、そのような例では、ビデオエンコーダ20は、イントラ予測を実行してビデオデータのCUの第1のPUの第1の予測ブロックを取得し得る。加えて、この例では、ビデオエンコーダ20は、インター予測を実行して同じCUの第2のPUの第2の予測ブロックを取得し得る。この例では、CUのサイズは $2N \times 2N$ であり、第1のPUのサイズは $2N \times N$ であり、第2のPUのサイズは $N \times 2N$ であり、または、第1のPUのサイズは $N \times 2N$ であり、第2のPUのサイズは $2N \times N$ であり、CUは4:2:0カラーフォーマットを使用してコーディングされる。さらに、第1のPUの第1の予測ブロックは第1のPUの輝度予測ブロックであり、第2のPUの第2の予測ブロックは第2のPUの輝度予測ブロックである。この例では、ビデオエンコーダ20は、第1のPUのイントラ予測モードを使用して、第1のPUの色差予測ブロックを生成し得る。さらに、この例では、ビデオエンコーダ20は、インター予測を使用して、第2のPUの色差予測ブロックを生成し得る。この例では、ビデオエンコーダ20は、第1の予測ブロック、第2の予測ブロック、第1のPUの色差予測ブロックおよび第2のPUの色差予測ブロックに基づいて、CUの残差データを取得し得る。

30

【 0 2 4 0 】

同様の例では、ビデオデコーダ30は、イントラ予測を実行してビデオデータのCUの第1のPUの第1の予測ブロックを取得し得る。この例では、ビデオデコーダ30は、インター予測を実行して同じCUの第2のPUの第2の予測ブロックを取得し得る。この例では、CUのサイズは $2N \times 2N$ であり、第1のPUのサイズは $2N \times N$ であり、第2のPUのサイズは $N \times 2N$ であり、または、第1のPUのサイズは $N \times 2N$ であり、第2のPUのサイズは $2N \times N$ であり、CUは4:2:0カラーフォーマットを使用してコーディングされる。さらに、この例では、第1のPUの第1の予測ブロックは第1のPUの輝度予測ブロックであり、第2のPUの第2の予測ブロックは第2のPUの輝度予測ブロックである。この例では、ビデオデコーダ30は、第1のPUのイントラ予測モードを使用して、第1のPUの色差予測ブロックを生成し得る。ビデオデコーダ30は、インター予測を使用して、第2のPUの色差予測ブロックを生成し得る。さらに、ビデオデコーダ30は、第1の予測ブロック、第2の予測ブロック、第1のPUの色差予測ブロックおよび

40

50

第2のPUの色差予測ブロックに基づいて、CUのコーディングブロックを再構築し得る。

【0241】

加えて、comb_modeを伴ういくつかの例では、1つのCUが2つ以上のPUを使用してコーディングされ、インター予測モードとイントラ予測モードの両方が使用されるとき、インターコーディングされるPUに対して、CUは現在のHEVC設計と同じ方法で扱われる。すなわち、再構築は、あり得る逆量子化/変換後の復号された残差と、動き情報を使用する動き補償された予測ブロックとの合計として定義される。加えて、イントラコーディングされたPUに対して、ビデオコーデは2つの予測子を伴うプロセスを使用し、すなわち、再構築は、あり得る逆量子化/変換後の復号された残差と、近隣のインターコーディングされたPUからの動き情報を使用する動き補償された予測ブロックと、現在のPUと関連付けられるイントラ予測モードを使用するイントラ予測ブロックとの合計として定義される。

10

【0242】

したがって、この例では、ビデオエンコーダ20は、イントラ予測を実行してビデオデータのCUの第1のPUの第1の予測ブロックを取得し得る。この例では、ビデオエンコーダ20は、インター予測を実行して同じCUの第2のPUの第2の予測ブロックを取得し得る。さらに、この例では、CUの残差データを取得することの一部として、ビデオエンコーダ20は、第1のPUに対応する残差データの各々のそれぞれのサンプルに対して、それぞれのサンプルが、CUのコーディングブロックのそれぞれのサンプルから、第2のPUの動き情報を使用して取得される予測サンプルを引き、第1の予測ブロックのサンプルを引いたものに等しくなるように、それぞれのサンプルを取得し得る。動き情報を使用して取得される予測サンプルは、インター予測PUの予測ブロックのサンプルであり得る。

20

【0243】

対応する例では、ビデオデコーダ30は、イントラ予測を実行してビデオデータのCUの第1のPUの第1の予測ブロックを取得し得る。この例では、ビデオデコーダ30は、インター予測を実行して同じCUの第2のPUの第2の予測ブロックを取得し得る。さらに、この例では、CUのコーディングブロックを再構築することの一部として、ビデオデコーダ30は、第1のPUに対応するコーディングブロックの各々のそれぞれのサンプルに対して、それぞれのサンプルが、それぞれの復号された残差サンプルと、第2のPUの動き情報を使用して取得される予測サンプルと、第1の予測ブロックのサンプルとの合計に等しくなるように、それぞれのサンプルを取得し得る。

30

【0244】

代わりに、comb_modeを伴ういくつかの例では、1つのCUが2つ以上のPUを使用してコーディングされ、インター予測モードとイントラ予測モードの両方が使用されるとき、イントラコーディングされたPUに対して、CUのコーディングブロックを再構築するためのプロセスは現在のHEVC設計と同じであり、すなわち、再構築は、あり得る逆量子化/変換後の復号された残差と、イントラ予測モードを使用するイントラ予測ブロックとの合計として定義される。加えて、CUのインターコーディングされたPUに対して、インターコーディングされたPUに対応するコーディングブロックの部分を再構築するためのプロセスは、2つの予測子がインターコーディングされたPUのために定義されるという点でHEVCの再構築プロセスと異なる。さらに、インターコーディングされたPUのサンプルに対応するCUのコーディングブロックの各サンプルに対して、サンプルは、復号された残差サンプル(たとえば、あり得る逆量子化/変換後の)と、インターコーディングされたPUの動き情報を使用して生成されるインターコーディングされたPUの動き補償された予測ブロックのサンプルと、インターコーディングされたPUに隣接するイントラコーディングされたPUと関連付けられるイントラ予測モードを使用して生成されるイントラ予測ブロックのサンプルとの合計として定義される。

40

【0245】

したがって、この例では、ビデオエンコーダ20は、イントラ予測を実行してビデオデータのCUの第1のPUの第1の予測ブロックを取得し得る。この例では、ビデオエンコーダ20は、インター予測を実行して同じCUの第2のPUの第2の予測ブロックを取得し得る。さらに、

50

この例では、CUの残差データを取得することの一部として、ビデオエンコーダ20は、第2のPUに対応する残差データの各々のそれぞれのサンプルに対して、それぞれのサンプルが、CUのコーディングブロックのそれぞれのサンプルから、第1のPUのイントラ予測モードを使用して取得される予測サンプルを引き、第2の予測ブロックのサンプルを引いたものに等しくなるように、それぞれのサンプルを取得し得る。

【0246】

対応する例では、ビデオデコーダ30は、イントラ予測を実行してビデオデータのCUの第1のPUの第1の予測ブロックを取得し得る。この例では、ビデオデコーダ30は、インター予測を実行して同じCUの第2のPUの第2の予測ブロックを取得し得る。さらに、この例では、CUのコーディングブロックを再構築することの一部として、ビデオデコーダ30は、第2のPUに対応するコーディングブロックの各々のそれぞれのサンプルに対して、それぞれのサンプルが、それぞれの復号された残差サンプルと、第1のPUのイントラ予測モードを使用して取得される予測サンプルと、第2の予測ブロックのサンプルとの合計に等しくなるように、それぞれのサンプルを取得し得る。

【0247】

一例では、2つの予測子を許容するとき、2つの予測ブロックが、線形加重関数、たとえば2つの平均を用いて合成される。たとえば、ビデオエンコーダ20またはビデオデコーダ30などのビデオコードは、イントラ予測を使用してPUの第1の予測ブロックを生成することができ、インター予測を使用してPUの第2の予測ブロックを生成することができる。この例では、ビデオコードは、PUの最終予測ブロックを、最終予測ブロックの各々のそれぞれのサンプルに対して、最終予測ブロックのそれぞれのサンプルに対応する第1の予測ブロックおよび第2の予測ブロックのサンプルの加重平均を決定することによって、決定し得る。この例では、加重平均に使用される重みは、インター予測される予測ブロックよりもイントラ予測される予測ブロックを重んじることがあり、またはその逆であることがある。いくつかの事例では、そのような線形加重関数を使用することは、より正確な最終予測ブロックをもたらすことがあり、これは最終的には圧縮性能を向上させることがある。線形加重係数は、サイド情報としてシグナリングされ、またはあるコーディングされた情報から導かれ得る。

【0248】

たとえば、CUのPUに対応するCUのコーディングブロックの各々のそれぞれのサンプルに対して、ビデオエンコーダ20は、それぞれのサンプルの第1の予測サンプルとそれぞれのサンプルの第2の予測サンプルとを取得し得る。たとえば、それぞれのサンプルの第1の予測サンプルはインター予測を使用して生成されることがあり、それぞれのサンプルの第2の予測サンプルはイントラ予測を使用して生成されることがある。この例では、ビデオエンコーダ20は、線形加重関数をそれぞれのサンプルの第1の予測サンプルおよびそれぞれのサンプルの第2の予測サンプルに適用することによって、それぞれのサンプルの加重予測サンプルを決定し得る。加えて、この例では、ビデオエンコーダ20は、それぞれのサンプルの元の値とそれぞれのサンプルの加重予測サンプルとの差に等しい、それぞれのサンプルの残差サンプルを決定し得る。

【0249】

同様に、コーディングブロックの各々のそれぞれのサンプルに対して、ビデオデコーダ30は、それぞれのサンプルの残差サンプルを取得し得る。たとえば、ビデオデコーダ30は、変換係数を示すシンタックス要素をビットストリームから取得し、逆量子化を変換係数に適用し、逆変換を変換係数に適用して残差サンプルを取得し得る。さらに、この例では、ビデオデコーダ30は、それぞれのサンプルの第1の予測サンプルとそれぞれのサンプルの第2の予測サンプルとを決定し得る。たとえば、それぞれのサンプルの第1の予測サンプルはインター予測を使用して生成されることがあり、それぞれのサンプルの第2の予測サンプルはイントラ予測を使用して生成されることがある。この例では、ビデオデコーダ30は、線形加重関数をそれぞれのサンプルの第1の予測サンプルおよびそれぞれのサンプルの第2の予測サンプルに適用することによって、それぞれのサンプルの加重予測サンプル

を決定し得る。加えて、この例では、ビデオデコーダ30は、それぞれのサンプルの残差サンプルとそれぞれのサンプルの加重予測サンプルとの合計として、それぞれのサンプルを再構築し得る。

【0250】

以下の例は、comb_modeが1つのスライス、ピクチャ、またはシーケンス(すなわち、コーディングされたビデオシーケンス)に対して有効にされるとき、comb_modeの使用法を示す。HEVCでは、CUは1ビットのpred_mode_flagシンタックス要素を含む。0に等しいCUのpred_mode_flagシンタックス要素は、CUがインター予測モードでコーディングされることを指定する。1に等しいCUのpred_mode_flagシンタックス要素は、CUがイントラ予測モードでコーディングされることを指定する。本開示の一例によれば、CUの1ビットのpred_mode_flagは、3つのあり得る値をとるシンタックス要素により置き換えられる。この例では、3つの値はそれぞれ、従来のイントラモード、従来のインターモード、およびcomb_modeに対応する。この例では、従来のイントラモードは、CUのすべてのPUがイントラ予測モードを使用してコーディングされる事例を指す。さらに、この例では、従来のインター予測モードは、CUのすべてのPUがインター予測モードを使用してコーディングされる事例を指す。いくつかの例では、comb_modeがあるCUに対して有効にされるとき、そのCUの1つだけのPUに対して、ビデオエンコーダ20は、そのPUがイントラ予測モードでコーディングされるかインター予測モードでコーディングされるかを示すための1ビット値をシグナリングする。そのCUはcomb_modeでコーディングされるので、他のPUは、その1ビット値がシグナリングされた対象のPUとは異なる予測モードである。別の例では、comb_modeは従来のインターモードとして扱われる。この例では、CUの各PUに対して、イントラ予測モードまたはインター予測モードの使用を示すために、追加のフラグが加えられる。

【0251】

本開示の第7の技法では、1つのPUはイントラ予測とインター予測の両方から予測されることが可能であり、イントラ予測およびインター予測からの2つの予測ブロックが、そのPUの最終予測ブロックを導出するために使用される。このようにして最終予測ブロックを導出することは、PUのより正確な予測ブロックをもたらすことがあり、これは圧縮性能を向上させることがある。

【0252】

したがって、第7の技法によれば、ビデオエンコーダ20は、イントラ予測を実行してCUのPUの第1の予測ブロックを取得し得る。加えて、この例では、ビデオエンコーダ20は、インター予測を実行して同じCUの同じPUの第2の予測ブロックを取得し得る。この例では、ビデオエンコーダ20は、第1の予測ブロックおよび第2の予測ブロックに基づいて、PUの最終予測ブロックを導出し得る。さらに、ビデオエンコーダ20は、PUの最終予測ブロックに基づいて、CUの残差データを取得し得る。たとえば、ビデオエンコーダ20は、PUの最終予測ブロックのサンプルとCUのコーディングブロックの対応するサンプルとの差を計算することによって、CUの残差データの少なくとも一部分を生成し得る。この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、CUの残差データを表すデータを含め得る。

【0253】

対応する例では、ビデオデコーダ30は、イントラ予測を実行してCUのPUの第1の予測ブロックを取得し得る。加えて、この例では、ビデオデコーダ30は、インター予測を実行して同じCUの同じPUの第2の予測ブロックを取得し得る。この例では、ビデオエンコーダ30は、第1の予測ブロックおよび第2の予測ブロックに基づいて、PUの最終予測ブロックを導出し得る。さらに、この例では、ビデオデコーダ30は、PUの最終予測ブロックに基づいて、CUのコーディングブロックを再構築し得る。たとえば、ビデオデコーダ30は、PUの最終予測ブロックをCUの残差データに加算して、CUのコーディングブロックの少なくとも一部分を再構築し得る。

【0254】

さらに、第7の技法のいくつかの例では、ビデオコーダは、線形加重関数を2つの予測ブ

10

20

30

40

50

ロックに適用し、たとえば、2つの予測ブロックの同じ相対的な場所に位置する画素の加重係数は固定されている。いくつかの例では、異なる場所に対する加重係数は可変であり得る。さらに、いくつかの例では、加重係数はイントラ予測モードに依存する。一例では、ブロック内の左上の場所に対して、イントラ予測モードがDCモードである場合、インター予測ブロックおよびイントラ予測ブロックの中の左上のサンプルの重みは等しく、すなわち(0.5, 0.5)であり、一方、イントラ予測が垂直予測モードである場合、イントラ予測ブロックの中の左上のサンプルの重みは、インター予測ブロックの中の左上のサンプルの重みより大きいことがある。

【0255】

第7の技法のいくつかの例では、すべての場所ではないが、1つまたは複数の場所における1つの画素の最終予測値は、イントラ予測されたブロックとインター予測されたブロックのいずれかから複製されることがあり、すなわち、2つの加重係数のうち的一方が0であり他方が1である。

【0256】

いくつかの例では、第7の技法は、特定の区分サイズ(たとえば、 $2N \times 2N$)、および/または特定の予測モード(マージ/スキップモード)に適用される。さらに、いくつかの例では、第7の技法が適用されるとき、イントラ予測モードは、従来のイントラ予測に使用されるイントラ予測モードのサブセットであるように制約される。一例では、サブセットはMPM(最確モード)のみを含むように定義される。

【0257】

上で論じられたように、ビデオコーデは、線形モデル(LM)予測モードを使用して、同じブロックの再構築された輝度サンプルに基づいてブロックの色差サンプルを予測し得る。さらに、上で説明されたように、LM予測モードは非正方形PUとともに使用されてこなかった。本開示の第8の技法によれば、ビデオコーデは非正方形PUとともにLM予測モードを使用することができる。より一般的には、LM予測モードを適用するための技法と同じものが、非正方形の輝度ブロックおよび色差ブロックとともに機能する。したがって、非正方形PUに関する第8の技法についての本開示における議論は、PUの輝度予測ブロックおよび色差予測ブロックなどの、非正方形の輝度ブロックおよび色差ブロックに対して一般に当てはまり得る。さらに、第8の技法の例は、LM予測モードにおいて使用されるパラメータをいくつかの方法で導出することができる。

【0258】

たとえば、第8の技法のいくつかの例では、非正方形PUの長辺における境界は、ダウンサンプリングまたはサブサンプリングされた境界の中の画素の数が短辺の境界における画素の数に等しくなるように、ダウンサンプリングまたはサブサンプリングされる。このプロセスは、デシメーションまたは補間されたサンプリングであり得る。ビデオデコーダ30がデシメーションを使用してサブサンプリングを実行する例では、ビデオデコーダ30は、通常の間隔でサンプルを除去し(たとえば、1つおきにサンプルを除去し)、残りのサンプルの値を変更せずにサンプルの数を減らすことができる。別の例では、ビデオデコーダ30は、補間を使用してサブサンプリングを実行することができる。ビデオデコーダ30が補間を使用してサブサンプリングを実行する例では、隣り合うサンプルのそれぞれのペアに対して、ビデオデコーダ30は、それぞれのペアのサンプルの間の値を補間することができ、サンプルのサブサンプリングされたセットにその補間された値を含めることができる。

【0259】

したがって、本開示の第8の技法の例では、ビデオエンコーダ20は、線形モデル予測演算を実行して、PUのダウンサンプリングまたはサブサンプリングされた再構築された輝度サンプルから、CUの非正方形PUの予測色差ブロックを予測することができる。さらに、この例では、ビデオエンコーダ20は、予測色差ブロックに基づいて、CUの残差データを取得し得る。この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、CUの残差データを表すデータを含め得る。対応する例では、ビデオデコーダ30は、線形モデル予測演算を実行して、PUのダウンサンプリングされた再構築さ

れた輝度サンプルから、ビデオデータの現在のピクチャのCUの非正方形PUの予測色差ブロックを予測することができる。この例では、ビデオデコーダ30は、予測色差ブロックに一部基づいて、CUのコーディングブロックを再構築し得る。この段落の例のいずれでも、ビデオエンコーダ20またはビデオデコーダ30が、非正方形PUの長辺に接したダウンサンプリングまたはサブサンプリングされた輝度サンプルの数が、非正方形PUの短辺に接した輝度サンプルと同じであるように、非正方形PUの長辺の輝度サンプルをダウンサンプリングまたはサブサンプリングし得る。

【0260】

一例では、式(4)および式(5)を使用して線形モデルパラメータを計算するとき、輝度成分と色差成分の両方に対して、非正方形PUの長辺における境界の画素は、ダウンサンプリングまたはサブサンプリングされた境界における画素の数がより短い境界における画素の数(すなわち、 $\min(K, L)$)に等しくなるように、サブサンプリングされる。このサブサンプリングプロセスは、デシメーションまたは補間されたサンプリングであり得る。

10

【0261】

したがって、この例では、LM予測動作を実行することの一部として、ビデオコーダは、予測色差サンプルが、併置輝度サンプルと乗じられた第1のパラメータと、第2のパラメータとを足したものに等しくなるように、予測色差サンプルを取得することができ、ここで第1のパラメータは

【0262】

【数22】

20

$$\alpha = \frac{I \sum x_i \cdot y_i - \sum x_i \cdot \sum y_i}{I \sum x_i \cdot x_i - \sum x_i \cdot \sum x_i}$$

【0263】

に等しく、第2のパラメータは

$$= (\sum y_i - \alpha \cdot \sum x_i) / I$$

30

に等しく、Iは非正方形PUの左および上の境界における参照サンプルの数であり、 x_i はダウンサンプリングまたはサブサンプリングされた再構築された輝度参照サンプルであり、 y_i は再構築された色差参照サンプルである。

【0264】

代わりに、いくつかの例では、PUの長辺と短辺の両方に位置する画素はサブサンプリングされることがあり、サブサンプリング比は異なることがある。サブサンプリング比は、サブサンプリングの後のサンプルに対するサブサンプリングの前のサンプルの比である。しかしながら、サブサンプリングの後の2つの辺における画素の総数は 2^m に等しくなければならないことが要求され得る(ここでmは整数であり、mは輝度成分および色差成分に対して異なり得る)。mの値はブロックサイズKおよびLに依存し得る。

40

【0265】

したがって、この例では、LM予測動作を実行することの一部として、ビデオコーダは、予測色差サンプルが、併置輝度サンプルと乗じられた第1のパラメータと、第2のパラメータとを足したものに等しくなるように、予測色差サンプルを取得することができ、ここで第1のパラメータは

【0266】

【数 2 3】

$$\alpha = \frac{I \sum x_i \cdot y_i - \sum x_i \cdot \sum y_i}{I \sum x_i \cdot x_i - \sum x_i \cdot \sum x_i}$$

【0 2 6 7】

に等しく、第2のパラメータは

$$= (\sum y_i - \sum x_i) / I$$

に等しく、Iは参照サンプルのセットの中の参照サンプルの数であり、 x_i は再構築された輝度参照サンプルであり、 y_i は再構築された色差参照サンプルである。この例では、参照サンプルのセットは、左の参照サンプルおよび上の参照サンプルのサブサンプリングされたセットであり、左の参照サンプルは現在のPUの左の境界のすぐ左にあり、上の参照サンプルは現在のPUの上の境界のすぐ上にある。

【0 2 6 8】

第8の技法の別の例では、式(4)および(5)における画素の数Iは、境界の中の画素の実際の数に基づいて調整される。たとえば、 $2N \times N$ のPUに対しては $I=3N$ である。左または上のみの原因となるサンプルが利用可能であるとき、全体の関与するサンプル数Iは、左または上の境界の長さに等しい。したがって、ビデオコーデは、

【0 2 6 9】

【数 2 4】

$$\alpha = \frac{3N \sum x_i \cdot y_i - \sum x_i \cdot \sum y_i}{3N \sum x_i \cdot x_i - \sum x_i \cdot \sum x_i}$$

【0 2 7 0】

として計算することができる。加えて、ビデオコーデは、

$$= (\sum y_i - \sum x_i) / 3N$$

として計算することができる。

【0 2 7 1】

LMが1つの非正方形色差PU(サイズが $K \times L$ に等しい、ここでKはLと等しくない)に対して有効にされるとき、パラメータ(すなわち、aおよびb)は様々な方法で導出され得る。たとえば、式(4)および式(5)を使用して線形モデルパラメータを計算するとき、輝度成分と色差成分の両方に対して、非正方形PUの短辺における境界の画素は、アップサンプリングされた境界における画素の数がより長い境界における画素の数(すなわち、 $\max(K, L)$)に等しくなるように、アップサンプリングされる。このアップサンプリングプロセスは、デュプリケータまたは補間されたサンプリングであり得る。デュプリケータアップサンプリングプロセスは、新しいサンプルを生成するために既存のサンプルが複製されるアップサンプリングプロセスである。補間されたアップサンプリングプロセスは、2つ以上の既存のサンプルに基づいて新しいサンプルの値を補間することによってサンプルの数を増やす。

【0 2 7 2】

したがって、この例では、LM予測動作を実行することの一部として、ビデオコーデは、予測色差サンプルが、併置輝度サンプルと乗じられた第1のパラメータと、第2のパラメータとを足したものに等しくなるように、予測色差サンプルを取得することができ、ここで第1のパラメータおよび第2のパラメータは式(4)および(5)において定義される。この例で

は、参照サンプルのセットは、左の参照サンプルおよび上の参照サンプルのアップサンプリングされたセットであり、左の参照サンプルは現在のPUの左の境界のすぐ左にあり、上の参照サンプルは現在のPUの上の境界のすぐ上にある。この例では、ビデオコーデは、左の参照サンプルおよび/または上の参照サンプルにアップサンプリング方法を適用することによって、参照サンプルのセットを決定することができる。たとえば、アップサンプリング方法は、左の参照サンプルまたは上の参照サンプルのうちで、現在のPUの左の境界と現在のPUの上の境界の短い方に相当するものをアップサンプリングしてよいが、現在のPUの左の参照サンプルと現在のPUの上の参照サンプルの長い方をアップサンプリングしなくてよい。

【0273】

10

いくつかの例では、PUの長辺と短辺の両方に位置する画素はアップサンプリングされることがあり、アップサンプリング比は異なることがある。しかしながら、アップサンプリングの後の2つの辺における画素の総数は 2^m に等しくなければならないことが要求され得る(ここで m は整数であり、 m は輝度成分および色差成分に対して異なり得る)。 m の値は、ブロックサイズ K および L に依存し得る。言い換えると、 m はPUの高さおよび/または幅に依存する。たとえば、PUは 8×16 であることがあり、ビデオコーデは、PUの左辺に沿った32個の参照サンプルおよびPUの上辺に沿った32個の参照サンプルがあるように、参照サンプルをアップサンプリングすることがある。この例では、 m は6に等しい。別の例では、PUは 4×8 であることがあり、ビデオコーデは、PUの左辺に沿った16個の参照サンプルおよびPUの上辺に沿った16個の参照サンプルがあるように、参照サンプルをアップサンプリングすることがある。この例では、 m は4に等しい。

20

【0274】

さらに、第8の技法のいくつかの例では、式(4)および式(5)を使用してLMパラメータを計算するとき、輝度成分と色差成分の両方に対して、非正方形PUの短辺における境界の画素がアップサンプリングされ、より長い境界の画素(すなわち、 $\max(K, L)$)は、アップサンプリングされたより短い境界における画素の数がサブサンプリングされたより長い境界における画素の数に等しくなるように、サブサンプリングされる。このアップサンプリングプロセスは、デュプリケートまたは補間されたサンプリングであり得る。このサブサンプリングプロセスは、デシメーションまたは補間されたサンプリングであり得る。

【0275】

30

したがって、この例では、LM予測動作を実行することの一部として、ビデオコーデは、予測色差サンプルが、併置輝度サンプルと乗じられた第1のパラメータと、第2のパラメータとを足したものに等しくなるように、予測色差サンプルを取得することができ、ここで第1のパラメータおよび第2のパラメータは式(4)および(5)において定義される。この例では、参照サンプルのセットは、参照サンプルのアップサンプリングされたセットと参照サンプルのサブサンプリングされたセットの連合であり、参照サンプルのアップサンプリングされたセットは、左の参照サンプルと上の参照サンプルのうちで含まれるサンプルがより少ない方の、アップサンプリングされたバージョンである。この例では、参照サンプルのサブサンプリングされたセットは、左の参照サンプルと上の参照サンプルのうちで含まれるサンプルがより多い方の、サブサンプリングされたバージョンである。この例では、左の参照サンプルは現在のPUの左の境界のすぐ左にあり、上の参照サンプルは現在のPUの上の境界のすぐ上にある。

40

【0276】

いくつかの例では、上で言及された第8の技法の例について、サブサンプリングまたはアップサンプリングプロセスの後で、輝度成分だけのためのダウンサンプリングプロセス(たとえば、本開示の他の箇所で説明されるような)がさらに、カラーフォーマットが4:4:4ではない場合をカバーするために適用され得る。したがって、現在のピクチャのカラーフォーマットが4:4:4以外であることに基づいて、ビデオコーデは、予測ブロックの輝度サンプルをサブサンプリングまたはダウンサンプリングすることができ、いくつかの例では、輝度サンプルの2つのダウンサンプリングプロセスが1つに統合され得る。

50

【0277】

さらに、第8の技法のいくつかの例では、境界画素のためのサブサンプリング/アップサンプリングの様々な方法が適用され得る。一例では、サブサンプリング/アップサンプリング方法は、PUサイズに(すなわち、KおよびLの値に)依存する。別の例では、サブサンプリング/アップサンプリングの方法は、シーケンスパラメータセット、ピクチャパラメータセット、スライスヘッダ、または別のシンタックス構造においてシグナリングされ得る。

【0278】

第8の技法のいくつかの例では、アップサンプリング/ダウンサンプリング(またはサブサンプリング)は、暗黙的に実施される。言い換えると、アップサンプリングまたはサブサンプリング技法は暗黙的に決定される。すなわち、左側の境界または/および上側の境界の、式(4)および式(5)における $x_i \cdot y_i$ 、 x_i 、および y_i などの合計値が、係数Sと乗じられ、または係数Sで割られる。Sの値は左側の境界または/および上側の境界における画素数の比に依存し得る。

【0279】

したがって、この例では、LM予測動作を実行して予測色差ブロックを予測することの一部として、ビデオコーダは、予測色差サンプルが、併置輝度サンプルと乗じられた第1のパラメータと、第2のパラメータとを足したものに等しくなるように、予測色差サンプルを取得することができ、ここで第1のLMパラメータは

【0280】

【数25】

$$\alpha = \frac{I \cdot S \sum x_i y_i - S \sum x_i \cdot S \sum y_i}{I \cdot S \sum x_i x_i - S \sum x_i \cdot S \sum x_i},$$

【0281】

に等しく、ここでSは非正方形PUの左側の境界または/および上側の境界における画素数の比に依存し、Iはサブサンプリング方法に従って決定された現在のPUの左の境界および上の境界におけるサンプルのサブセットの中の参照サンプルの数であり、 x_i はサブサンプリングされ再構築された輝度参照サンプルであり、 y_i は再構築された色差参照サンプルである。いくつかの例では、 $K \times L$ 色差ブロックに対して、 $S = \max(K, L) / \min(K, L)$ である。

【0282】

上で説明されたように、DST-VII、DCT-VIII、DST-I、およびDCT-Vを使用する、enhancement multiple transform(EMT)方式が提案されている。さらに、上で論じられたように、EMTが適用されるかどうかは、CU内のすべてのTUに対して、フラグ、すなわちEMTフラグを使用して、CUレベルで制御される。EMT対応CU内の各TUに対して、使用されるべき水平方向または垂直方向の変換は、選択された変換セットに対するインデックス、すなわちEMTインデックスによってシグナリングされる。

【0283】

しかしながら、これまでに提案されたようにEMT方式を制御することは、CUの中の各PUの残差特性が異なる場合、効率的ではないことがある。たとえば、これまでに提案されたようにEMT方式を制御することは、CU内のイントラコーディングされたPUおよびインターコーディングされたPUに対して効率的ではないことがある。したがって、本開示の第9の技法によれば、EMTが1つのスライス、ピクチャ、またはシーケンスに対して有効にされ、1つのCUが垂直方向に2つのPUへと分割される(たとえば、 $N \times 2N$ の区分)とき、EMTフラグのシグナリングは次の方法で修正される。左のPUがイントラコーディングされたPUである場合、変換深度は0であり得る。言い換えると、CUの変換木は0以上の深度を有し得る。この場合、EMTフラグはCUレベルでシグナリングされ得る。変換深度が0ではない場合、EMTフラグはPUレベルでシグナリングされ得る。したがって、EMTは、各PUに対して有効にされ

10

20

30

40

50

ることもされないこともある。

【0284】

さらに、本開示の第9の技法によれば、EMTが1つのスライス、ピクチャ、またはシーケンスに対して有効にされ、1つのCUが水平方向に2つのPUへと分割される(たとえば、 $2N \times N$ の区分)とき、EMTフラグのシグナリングは次の方法で修正される。上のPUがイントラコーディングされたPUである場合、変換深度は0であり得る。言い換えると、CUの変換木は0以上の深度を有し得る。この場合、EMTフラグはCUレベルでシグナリングされ得る。変換深度が0ではない場合、EMTフラグはPUレベルでシグナリングされ得る。すなわち、各PUはEMTが有効であることもないこともある。

【0285】

したがって、第9の技法によれば、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、第1のシンタックス要素を含め得る。第1のシンタックス要素は、境界に沿った丁度2つのPUへと区分される特定のCUに対してEMTが有効にされるかどうかを示す。この例では、第1のシンタックス要素がその特定のCUの中にあるか、2つのPUのうちの特定のPUの中にあるかは、CUのPUの分割方向に依存する。さらに、この例では、EMTがある特定のCUに対して有効にされることに基づいて、その特定のCUの各々のそれぞれのTUに対して、ビデオエンコーダ20は、ビットストリームに、それぞれのTUのそれぞれの選択された変換セットを示すそれぞれのシンタックス要素を含め得る。この例では、EMTがその特定のCUに対して有効にされることに基づいて、ビデオエンコーダ20は、それぞれの選択された変換セットの1つまたは複数の変換をそれぞれのTUの変換係数に適用して、サンプル領域の中のそれぞれのTUのそれぞれの変換ブロックを取得し得る。この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、CUのTUのうちの1つまたは複数を表すデータを含め得る。この例では、境界は水平な境界であることがあり、または境界は垂直な境界であることがある。

【0286】

対応する例では、ビデオデコーダ30は第1のシンタックス要素を取得し得る。第1のシンタックス要素は、境界に沿った丁度2つのPUへと区分される特定のCUに対してEMTが有効にされるかどうかを示す。この例では、第1のシンタックス要素がその特定のCUの中にあるか、2つのPUのうちの特定のPUの中にあるかは、CUのPUの分割方向に依存する。この例では、EMTがある特定のCUに対して有効にされると決定したことに応答して、その特定のCUの各々のそれぞれのTUに対して、ビデオデコーダ30は、それぞれのTUのそれぞれの選択された変換セットを示すそれぞれのシンタックス要素を取得し得る。加えて、EMTがある特定のCUに対して有効にされると決定したことに応答して、その特定のCUの各々のそれぞれのTUに対して、ビデオデコーダ30は、それぞれの選択された変換セットの1つまたは複数の変換の逆をそれぞれのTUの変換係数に適用して、サンプル領域の中のそれぞれのTUのそれぞれの変換ブロックを取得し得る。この例では、ビデオデコーダ30は、CUのTUの変換ブロックに少なくとも一部基づいて、CUのコーディングブロックを再構築し得る。この例では、境界は水平な境界であることがあり、または境界は垂直な境界であることがある。

【0287】

本開示の第10の技法によれば、いくつかの変換木構造が、1つのスライス、ピクチャ、またはシーケンスをコーディングするために適用され得る。たとえば、一例では、変換木構造は事前に定義される。いくつかの例では、各ピクチャ、スライス、最大コーディング単位、CU、またはPUに対して、ビデオエンコーダ20は、選択された変換木構造をシグナリングし得る。代わりに、いくつかの例では、ビデオデコーダ30は、予測モード/区分サイズなどのコーディングされた情報から、選択された変換木を導出し得る。

【0288】

したがって、第10の技法によれば、ビデオエンコーダ20は、複数の事前に定義された木構造のうちの特定の木構造に基づいて、CUのTUへとビデオデータのCUを区分し得る。この例では、木構造のルートノードはCUのコーディングブロックに対応する。さらに、この例では、木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親

ノードに対応するブロックの区分であるそれぞれのブロックに対応する。この例では、木構造のリーフノードはCUのTUに対応する。加えて、この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、CUのTUのうちの1つまたは複数を表すデータを含める。いくつかの例では、ビデオエンコーダ20はさらに、ビットストリームに、特定の木構造を特定する1つまたは複数のシンタックス要素を含め得る。いくつかの例では、CUに適用可能な特定の木構造を示す1つまたは複数のシンタックス要素は、ピクチャ、スライス、LCU、CU、およびPUのうちの1つの中にある。さらに、いくつかの例では、CUがTUへと区分されると決定することの一部として、ビデオエンコーダ20は、特定の木構造の明示的なシグナリングを伴わずに、コーディングされた情報から特定の木構造を決定する。そのような例では、コーディングされた情報は、予測モードと区分サイズのうちの少なくとも1つを備え得る。

10

【0289】

対応する例では、ビデオデコーダ30は、複数の定義された木構造のうちのある特定の木構造に基づいて、CUのTUへとビデオデータのCUが区分されると決定し得る。この例では、木構造のルートノードはCUのコーディングブロックに対応する。木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応する。この例では、木構造のリーフノードはCUのTUに対応する。加えて、この例では、ビデオデコーダ30は、CUのTUのうちの少なくとも1つに対するデータに基づいて、CUのコーディングブロックを再構築し得る。いくつかの例では、ビデオデコーダ30は、符号化されたビデオデータを備えるビットストリームから、特定の木構造を特定する1つまたは複数のシンタックス要素を取得し得る。1つまたは複数のシンタックス要素は、ピクチャ、スライス、LCU、CU、および予測単位のうちの1つにおいて、CUに適用可能な特定の木構造を示す。さらに、いくつかの例では、CUがTUへと区分されると決定することの一部として、ビデオデコーダ30は、特定の木構造の明示的なシグナリングを伴わずに、コーディングされた情報から特定の木構造を決定し得る。コーディングされた情報は、予測モードと区分サイズのうちの少なくとも1つを備え得る。

20

【0290】

本開示の第11の例では、サイズが $1 \times N$ および $N \times 1$ に等しい変換も、インターコーディングされたブロックに適用され得る。たとえば、一例では、そのようなTUは、特定の变换深度、たとえば最高の变换深度だけに許容される。いくつかの例では、そのようなTUは、 8×8 に等しいCUサイズなどの、特定のコーディングブロックだけに許容される。さらに、いくつかの例では、第11の技法は、輝度などの特定の色成分だけに適用可能である。

30

【0291】

したがって、第11の技法によれば、ビデオエンコーダ20は、 $1 \times N$ または $N \times 1$ 変換をインターコーディングされたブロックの残差データに適用することによって、変換領域データ(たとえば、変換係数)を決定し得る。この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、変換領域データを表すデータを含め得る。対応する例では、ビデオデコーダ30は、 $1 \times N$ または $N \times 1$ 変換をインターコーディングされたブロックの変換係数に適用することによって、サンプル領域データを決定し得る。この例では、ビデオデコーダ30は、サンプル領域データの一部に基づいて、ビデオデータのCUのコーディングブロックを再構築し得る。たとえば、ビデオデコーダ30は、残差データの対応するサンプルにサンプル領域データのサンプルを加算して、CUのコーディングブロックを再構築し得る。いくつかの事例では、ビデオエンコーダ20およびビデオデコーダ30を伴う第11の技法の上の例では、 $1 \times N$ および $N \times 1$ の変換は、特定の变换深度だけに許容される。加えて、いくつかの事例では、ビデオエンコーダ20およびビデオデコーダ30を伴う第11の技法の上の例では、 $1 \times N$ および $N \times 1$ の変換は、特定のサイズのCUだけに許容される。

40

【0292】

本開示の第12の技法によれば、インターコーディングされたCUのためのHEVCにおいて定義される非対称動き区分も、イントラコーディングされたCUに適用される。イントラ予測

50

されたCUをPUへと非対称に区分することで、ビデオエンコーダ20は、異なるオブジェクトに対応する領域へとCUをより正確に分割することが可能になることがあり、これは圧縮性能を向上させることがある。したがって、第12の技法の例によれば、ビデオエンコーダ20は、ビデオデータのイントラ予測されたCUをPUへと非対称に区分し得る。この例では、ビデオエンコーダ20は、イントラ予測されたCUの各々のそれぞれのPUのそれぞれの予測ブロックを決定し得る。さらに、この例では、ビデオエンコーダ20は、イントラ予測されたCUのPUの予測ブロックおよびイントラ予測されたCUのコーディングブロックに基づいて残差データを取得し得る。加えて、この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、残差データを表すデータを含め得る。

【0293】

10

第12の技法の対応する例によれば、ビデオデコーダ30は、ビデオデータのイントラ予測されたCUがPUへと非対称に区分されると決定し得る。この例では、ビデオデコーダ30は、イントラ予測されたCUの各々のそれぞれのPUのそれぞれの予測ブロックを決定し得る。加えて、この例では、ビデオデコーダ30は、イントラ予測されたCUのPUの予測ブロックに基づいて、イントラ予測されたCUのコーディングブロックを再構築し得る。

【0294】

本開示の第13の技法によれば、1つのイントラコーディングされたCUが複数のPUを含むとき、各PUは固有の色差予測モードを有し得る。言い換えると、PUは、第1のイントラ予測モード(すなわち、輝度予測モード)および第2のイントラ予測モード(すなわち、色差予測モード)を有し得る。ビデオコーダは、輝度予測モードを使用してPUの輝度予測ブロックを決定することができ、色差予測モードを使用してPUの色差予測ブロックを決定することができる。したがって、第13の技法によれば、ビデオエンコーダ20は、ビデオデータのイントラ予測されたCUが少なくとも第1のPUおよび第2のPUを有すると決定することができる。この例では、第1のPUおよび第2のPUは異なる色差予測モードを有する。さらに、この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、第1のPUおよび第2のPUの予測ブロックならびにCUのコーディングブロックに少なくとも一部基づいて残差データを表すデータを含め得る。

20

【0295】

第13の技法の対応する例では、ビデオデコーダ30は、ビデオデータのイントラ予測されたCUが少なくとも第1のPUおよび第2のPUを有すると決定することができる。この例では、第1のPUおよび第2のPUは異なる色差予測モードを有する。さらに、この例では、ビデオデコーダ30は、第1のPUおよび第2のPUの予測ブロックに少なくとも基づいて、CUのコーディングブロックを再構築し得る。

30

【0296】

さらに、第13の技法の例によれば、前にコーディングされたPUの色差イントラ予測モードが、次のPUをコーディングするために考慮され得る。したがって、ビデオコーダは、コーディング順序において現在のPUの前のPUの色差予測モードに少なくとも一部基づいて、現在のPUの色差予測モードを決定し得る。たとえば、ビデオコーダは、現在のPUの色差イントラ予測モードのためのコンテキストモデリングにおいて、前にコーディングされたPUの色差イントラ予測モードを使用し得る。コンテキストモデリングは、コンテキスト適応エントロピーコーディングのためのコーディングコンテキストの識別を備え得る。コーディングコンテキストは値の確率を示し得る。別の例では、ビデオコーダは、前にコーディングされたPUの色差イントラ予測モードを、色差イントラ予測モードリストのための1つの新しい候補として追加し得る。

40

【0297】

第13の技法のいくつかの例では、すべてのPUが同じ色差イントラ予測モードを共有するかどうかを示すために、1つのフラグがCUレベルで最初にコーディングされ得る。したがって、この例では、ビデオエンコーダ20は、ビットストリームに、イントラ予測されたCUのすべてのPUが同じ色差イントラ予測モードを共有することを示すシンタックス要素を含め得る。同様に、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビッ

50

トストリームから、イントラ予測されたCUのすべてのPUが同じ色差イントラ予測モードを共有するかどうかを示すシンタックス要素を取得し得る。

【0298】

さらに、第13の技法のいくつかの例では、1つのCU内のすべての色差PUは、同じ変換木に従うように制約される。同じ変換木に従うように1つのCU内の色差PUのすべてを制約することによって、ビデオエンコーダ20は、異なる色差PUのための異なる変換木の構造を示すデータをビットストリームに含めることが不要になり得る。したがって、ビデオエンコーダ20は、CUの色差PUが異なる構造の変換木を有するビットストリームをビデオエンコーダが生成するのを制限する、ビデオコーディング規格に準拠するビットストリームを生成し得る。同様に、ビデオデコーダ30は、ビデオデータの符号化された表現を備えるビットストリームを取得し得る。この例では、ビットストリームは、CUの色差PUが異なる構造の変換木を有するビットストリームをビデオエンコーダが生成するのを制限する、ビデオコーディング規格に準拠する。

10

【0299】

本開示の第14の例では、1つのイントラコーディングされたCUが複数の長方形PUを含むとき、ビデオコーデはモード依存の走査を適用し得る。モード依存の走査は、TUのための2次元係数ブロックの中の変換係数をエントロピー符号化のための1次元係数ベクトルへと走査するために使用される走査順序である。ビデオエンコーダ20は、TUに対応するPUのためにどのイントラ予測モードが使用されるかに基づいて、複数の利用可能な走査順序の中から、TUの変換係数を走査するために使用すべきモード依存の走査を選択し得る。TUに対応するPUは、TUと同じ広がりをもつことがあり、またはTUと関連付けられるエリアを含むことがある。モード依存の走査を使用することで、CABACのための変換係数をより良好に並べることができる。HEVCでは、モード依存の走査は、 8×8 および 4×4 のTUだけに許容される。

20

【0300】

したがって、第14の技法の例によれば、ビデオエンコーダ20は、2次元変換係数ブロックに基づいて残差データを取得し得る。この例では、ビデオエンコーダ20は、ビデオデータのイントラ予測されたCUの複数の長方形PUの各々の予測ブロックを取得し得る。さらに、この例では、ビデオエンコーダ20は、モード依存の走査を適用して、変換係数の2次元ブロックを、CUのTUに対応する変換係数の1次元アレイへと並べ得る。この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、変換係数の1次元アレイを表すデータを含め得る。

30

【0301】

同様の例では、ビデオデコーダ30は、モード依存の走査を適用して、変換係数の1次元アレイを、ビデオデータのイントラ予測されたCUのTUに対応する2次元変換係数ブロックへと並べ得る。この例では、イントラ予測されたCUは複数の長方形PUを有する。さらに、この例では、ビデオデコーダ30は、変換係数ブロックに基づいて残差データを取得し得る。加えて、ビデオデコーダ30は、PUの各々の予測ブロックを取得し得る。この例では、ビデオデコーダ30は、残差データおよび予測ブロックに基づいて、CUのコーディングブロックを再構築し得る。

40

【0302】

第14の技法の一例では、モード依存の走査の適用は、 8×4 または 4×8 などのあるTUサイズに制約される。いくつかの例では、モード依存の走査は、 8×8 だけ、または 8×8 および 16×16 などの、あるCUサイズに制約される。さらに、いくつかの例では、HEVCにおいて 8×8 および 4×4 に等しいTUサイズのために使用される、イントラ予測モードと走査パターンとの間のマッピングの規則が再使用され得る。いくつかの例では、長方形TUのサイズに応じて、異なるマッピング関数が適用され得る。

【0303】

本開示の他の箇所では説明されるように、VCEG-AZ07は、HEVCにおいて使用される2タップのイントラ補間フィルタよりも方向性イントラ予測の精度を改善するために、4タップの

50

イントラ補間フィルタを使用することを提案した。しかしながら、VCEG-AZ07は、非正方形のイントラコーディングされたPUのための4タップのイントラ補間フィルタをビデオコーダがどのように選択するかを示していない。むしろ、VCEG-AZ07は、ビデオコーダが、 4×4 および 8×8 ブロックにはキュービック補間フィルタを使用し、 16×16 以上のブロックにはガウシアン補間フィルタを使用することを規定している。本開示の第15の技法では、サイズが $K \times L$ に等しい非正方形のイントラコーディングされたPUに対して、4タップのイントラ補間フィルタに関して本開示の他の箇所で説明されるような4タップフィルタのタイプまたは走査パターンを決定するとき、非正方形のイントラコーディングされたPUは、 $N \times N$ に等しい変換サイズとして扱われ、ここで $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ であり、 \log_2 は2を底とする対数であり、 \gg および \ll はそれぞれ右シフトおよび左シフトである。

10

【0304】

したがって、第15の技法の例では、ビデオエンコーダ20は、ビデオデータのCUの非正方形のイントラコーディングされたPUのための4タップの補間フィルタを決定し得る。さらに、この例では、ビデオエンコーダ20は、非正方形のイントラコーディングされたPUの予測ブロックを取得することの一部として、決定された4タップの補間フィルタを適用し得る。たとえば、ビデオエンコーダ20は、2つの整数場所の参照サンプル(すなわち、ピクチャの左上のサンプルに対して相対的に整数の座標にある参照サンプル)の間にある参照サンプルの値を決定するとき、4タップのフィルタを適用し得る。加えて、この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、非正方形PUの予測ブロックおよびCUのコーディングブロックに少なくとも一部基づいて残差データを表すデータを含め得る。この例では、4タップの補間フィルタを決定することの一部として、ビデオエンコーダ20は、正方形PUのサイズに基づいて4タップの補間フィルタを決定することができ、ここで正方形PUのサイズは非正方形のイントラコーディングされたPUの高さおよび幅に基づく。

20

【0305】

第15の技法の対応する例では、ビデオデコーダ30は、ビデオデータのCUの非正方形のイントラコーディングされたPUのための4タップの補間フィルタを決定し得る。加えて、この例では、ビデオデコーダ30は、非正方形のイントラコーディングされたPUの予測ブロックを取得することの一部として、決定された4タップの補間フィルタを適用し得る。さらに、ビデオデコーダ30は、非正方形PUの予測ブロックに少なくとも一部基づいて、CUのコーディングブロックを再構築し得る。この例では、4タップの補間フィルタを決定することの一部として、ビデオデコーダ30は、正方形PUのサイズに基づいて4タップの補間フィルタを決定することができ、ここで正方形PUのサイズは非正方形のイントラコーディングされたPUの高さおよび幅に基づく。

30

【0306】

第15の技法のいくつかの例では、新しい4タップのフィルタは、非正方形のイントラコーディングされたPUに適用され得る。すなわち、非正方形のイントラコーディングされたPUに対しても、4タップのフィルタが適用されてよく、このフィルタは、正方形PUに対して定義されるものとは異なり得る。さらに、第15の技法のいくつかの例では、イントラ予測モードと走査パターンインデックスとの間の異なるマッピングテーブルが、非正方形のイントラコーディングされたPUに対して適用され得る。

40

【0307】

$K \times L$ 変換ブロックは、 $N \times N$ に等しい変換サイズとして扱われ、ここで $\log_2(N \times N) = (\log_2(K) + \log_2(L) + 1)$ である。したがって、 $K \times L$ 変換ブロックに対する走査パターンの選択は、 $N \times N$ ブロックと同じであり得る。

【0308】

上で論じられたように、HEVCにおける変換ブロックはサイズが $N \times N$ であり、ここで $N = 2^m$ であり m は整数である。さらに、HEVCでは、ビデオエンコーダは、2次元の $N \times N$ 変換を変換ブロックに適用して変換係数を生成する。より具体的には、ビデオエンコーダは、 N 点の1

50

次元変換を変換ブロックの各行および変換ブロックの各列に別々に適用することによって、2次元の $N \times N$ 変換を適用する。このように変換を適用することで、変換係数の $N \times N$ ブロックが得られる。

【 0 3 0 9 】

HEVCでは、ビデオエンコーダは、以下を計算することによって、 N 点の1次元DCT変換を変換ブロックのサンプルの i 番目の行または列 w_i に適用し得る。

【 0 3 1 0 】

【 数 2 6 】

$$w_i = \sum_{j=0}^{N-1} u_j c_{ij} \quad (18)$$

10

【 0 3 1 1 】

上の式において、 $i = 0, \dots, N - 1$ である。DCT変換行列 C の要素 c_{ij} は、次のように定義される。

【 0 3 1 2 】

【 数 2 7 】

$$c_{ij} = \frac{A \cdot \cos\left[\frac{\pi}{N}\left(j+\frac{1}{2}\right)i\right]}{\sqrt{N}} \quad (19)$$

20

【 0 3 1 3 】

上の式において、 $i, j = 0, \dots, N - 1$ であり、ここで A は、 $i=0$ に対しては1に等しく、 $i>0$ に対しては $2^{1/2}$ に等しい。

【 0 3 1 4 】

式(19)において、

【 0 3 1 5 】

【 数 2 8 】

30

$$A \cdot \cos\left[\frac{\pi}{N}\left(j+\frac{1}{2}\right)i\right]$$

【 0 3 1 6 】

を X_{ij} と表記するものとする。したがって、式(18)は次のように書き直され得る。

【 0 3 1 7 】

【 数 2 9 】

40

$$w_i = \sum_{j=0}^{N-1} \frac{u_j X_{ij}}{\sqrt{N}}$$

【 0 3 1 8 】

ビデオエンコーダは、水平方向と垂直方向の両方に1次元DCT変換を適用するので、変換係数 w_i は最終的に次のように書き直され得る。

【 0 3 1 9 】

【数 3 0】

$$w_i = \sum_{k=0}^{N-1} \left(\left(\sum_{j=0}^{N-1} \frac{u_j X_{ij}}{\sqrt{N}} \right) \cdot \frac{X_{ik}}{\sqrt{N}} \right) \quad (20)$$

【0 3 2 0】

これはさらに次のように書き直され得る。

【0 3 2 1】

【数 3 1】

10

$$\frac{\sum_{k=0}^{N-1} \left(\left(\sum_{j=0}^{N-1} u_j X_{ij} \right) \cdot X_{ik} \right)}{\sqrt{N} \cdot \sqrt{N}} \quad (21)$$

【0 3 2 2】

したがって、変換は最終的に、

【0 3 2 3】

【数 3 2】

20

$$\sqrt{N} \cdot \sqrt{N}$$

【0 3 2 4】

という「正規化係数」を有するものと見なすことができる。N=2^mなので、

【0 3 2 5】

【数 3 3】

$$\sqrt{N} \cdot \sqrt{N}$$

30

【0 3 2 6】

も2のべき乗である。したがって、変換係数の値は、除算演算の代わりに右シフト演算によって実装され得る。本開示の他の箇所で論じられるように、除算演算の代わりに右シフト演算を使用することで、複雑さを下げ、コーディング速度を上げることができる。

【0 3 2 7】

しかしながら、TUの非正方形変換ブロックとともに式(19)を再使用すると、問題が生じることがある。2D変換(水平方向の変換と垂直方向の変換の両方を含む)に対して、K*L変換を考えると、正規化係数は

【0 3 2 8】

【数 3 4】

40

$$(\sqrt{K} * \sqrt{L})$$

【0 3 2 9】

となる。Nが式 $\log_2(N*N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ を満たす値として定義される場合、利用される正規化係数

【0 3 3 0】

【数 3 5】

$$\sqrt{N} \cdot \sqrt{N}$$

【0 3 3 1】

と実際の正規化係数

【0 3 3 2】

【数 3 6】

10

$$(\sqrt{K} * \sqrt{L})$$

【0 3 3 3】

との比は、

【0 3 3 4】

【数 3 7】

$$1/\sqrt{2}$$

20

【0 3 3 5】

となる。言い換えると、量子化プロセスにおいて $N \times N$ 変換ブロックから導出された同じ正規化係数を再使用するとき、エネルギー(すなわち、量子化された変換係数の平方根の合計)は、

【0 3 3 6】

【数 3 8】

$$\sqrt{2}$$

30

【0 3 3 7】

だけ変化する。

【0 3 3 8】

本開示の第16の技法は、この問題に対処し得る。たとえば、本開示の第16の技法では、サイズが $K \times L$ に等しい非正方形変換ブロックに対して、 $(\log_2(K) + \log_2(L))$ が奇数であるとき、HEVCにおける変換および量子化プロセスは変更されないままであり、非正方形変換ブロックはサイズが $N \times N$ と等しい変換ブロックとして扱われ、ここで $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ である。言い換えると、 $(\log_2(K) + \log_2(L))$ が奇数であることに基づいて、ビデオエンコーダ20は、 $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ となるような値 N を決定し得る。ビデオエンコーダ20は次いで、「正規化係数」において N の決定された値を使用して式(19)に従って定義される、DCT変換行列 C の要素を使用し得る。したがって、ビデオエンコーダ20は、式(21)の「正規化係数」による除算のために右シフト演算を使用し続け得る。

40

【0 3 3 9】

さらに、本開示の第16の技法によれば、変換プロセスの後で、かつ量子化プロセスの前に、変換係数が修正され、2という係数と乗じられる。言い換えると、変換を非正方形変換ブロックに適用して係数ブロックを生成した後で、ビデオエンコーダ20は、係数ブロックの各変換係数を2という係数と乗じる。これは、使用される正規化係数(すなわち、

【0 3 4 0】

50

【 数 3 9 】

$$(\sqrt{N} * \sqrt{N})$$

【 0 3 4 1 】

) の実際の正規化係数 (すなわち、

【 0 3 4 2 】

【 数 4 0 】

10

$$(\sqrt{K} * \sqrt{L})$$

【 0 3 4 3 】

) に対する比が、

【 0 3 4 4 】

【 数 4 1 】

$$\frac{1}{\sqrt{2}}$$

20

【 0 3 4 5 】

に等しいからである。

【 0 3 4 6 】

たとえば、K=8かつL=4とする。この例では、 $\log_2(4*4) = ((\log_2(8) + \log_2(4)) \gg 1)$
 $\ll 1)$ なので、N=4である。

【 0 3 4 7 】

【 数 4 2 】

$$\frac{\sqrt{8} \times \sqrt{4}}{\sqrt{4} \times \sqrt{4}}$$

30

【 0 3 4 8 】

は

【 0 3 4 9 】

【 数 4 3 】

$$\frac{\sqrt{8 \times 4}}{\sqrt{4 \times 4}}$$

【 0 3 5 0 】

40

に等しく、これは

【 0 3 5 1 】

【 数 4 4 】

$$\frac{\sqrt{32}}{\sqrt{16}}$$

【 0 3 5 2 】

に等しく、これは 2 に等しい。 $(\log_2(K) + \log_2(L))$ が偶数であるKおよびLの値に対して、
 使用される正規化係数 (すなわち、

50

【 0 3 5 3 】

【 数 4 5 】

$$(\sqrt{N} * \sqrt{N})$$

【 0 3 5 4 】

) の実際の正規化係数 (すなわち、

【 0 3 5 5 】

【 数 4 6 】

10

$$(\sqrt{K} * \sqrt{L})$$

【 0 3 5 6 】

) に対する比は1に等しいことに留意されたい。したがって、 $(\log_2(K) + \log_2(L))$ が偶数であるとき、ビデオエンコーダ20が変換係数を 2 という係数と乗じる必要はないことがある。

【 0 3 5 7 】

逆量子化プロセスの後、逆量子化された係数がさらに修正され、 2 という係数で割られる。量子化の前に変換係数を 2 と乗じ、変換係数を 2 で割ることで、そうされなければ量子化プロセスにおいて失われるであろう情報を保つことができる。この情報を保つことで、元の変換ブロックのより正確な再構築が確実になり得る。

20

【 0 3 5 8 】

第16の技法の別の例では、HEVCにおける変換および量子化プロセスは、変化しないままであり、 $N \times N$ に等しい変換サイズとして扱われ、ここで $\log_2(N \times N) = (\log_2(K) + \log_2(L) + 1)$ である。変換の後に、かつ量子化プロセスの前に、変換係数が修正され、 2 という係数で割られる。逆量子化プロセスの後に、逆量子化された係数がさらに修正され、 2 という係数と乗じられる。

【 0 3 5 9 】

30

上の第16の技法の例では、 2 という係数は近似により表され得る。たとえば、 $(x * 2)$ という処理は $(x * 181) \gg 7$ により近似することができ、ここで \gg は右シフト演算を表す。 $(x / 2)$ という処理は $(x * 2/2)$ 、すなわち $(x * 181) \gg 8$ により近似することができ、ここで \gg は右シフト演算を表す。

【 0 3 6 0 】

したがって、上で提示された第16の技法の例では、ビデオエンコーダ20は、CUの非正方形TUの変換ブロックに変換を適用して、変換係数のブロックを生成し得る。加えて、ビデオエンコーダ20は、変換係数のブロックの各々のそれぞれの変換係数が、

【 0 3 6 1 】

【 数 4 7 】

40

$$\sqrt{2}$$

【 0 3 6 2 】

の近似と乗じられたそれぞれの逆量子化された変換係数に基づくように、変換係数を修正し得る。この例では、変換係数を修正した後で、ビデオエンコーダ20は、量子化プロセスをCUの非正方形PUの修正された変換係数に適用し得る。さらに、この例では、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、量子化された変換係数に基づくデータを含め得る。いくつかの例では、非正方形TUの変換ブロックに

50

変換を適用することの一部として、ビデオエンコーダ20は、逆量子化された変換係数に、サイズが $N \times N$ である変換を適用することができ、ここで $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ である。

【0363】

対応する例では、ビデオデコーダ30は、逆量子化プロセスをビデオデータのCUの非正方形PUの変換係数に適用し得る。この例では、逆量子化プロセスを変換係数に適用した後で、ビデオデコーダ30は、逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が、

【0364】

【数48】

10

$$\sqrt{2}$$

【0365】

の近似で割られたそれぞれの逆量子化された変換係数に基づくように、逆量子化された変換係数を修正し得る。いくつかの例では、修正された逆量子化された変換係数に逆変換を適用することの一部として、ビデオデコーダ30は、修正された逆量子化された変換係数に、サイズが $N \times N$ である変換を適用することができ、ここで $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ である。

20

【0366】

HEVCでは、ビデオエンコーダ20は、次の式を使用して量子化された変換係数(すなわち、level)を計算し得る。

【0367】

【数49】

$$level = \left((coeff \times f_{QP \% 6} + offset_Q) \gg \frac{QP}{6} \right) \gg shift2 \quad (22)$$

【0368】

30

ここで、coeffは変換係数であり、 $offset_Q$ はオフセット値であり、QPは量子化パラメータであり、 $shift2 = 29 - M - B$ であり、Bはビット深度であり、 $M = \log_2(N)$ であり、 $f = [f_0, \dots, f_5]^T = [26214, 23302, 20560, 18396, 16384, 14564]^T$ (23)である。

【0369】

さらに、HEVCでは、ビデオデコーダ30は、次の式を使用して量子化された変換係数を逆量子化し得る。

【0370】

【数50】

40

$$coeff_Q = \left(\left(level \times \left(g_{QP \% 6} \ll \frac{QP}{6} \right) + offset_{IQ} \right) \gg shift1 \right) \quad (24)$$

【0371】

式(24)において、 $coeff_Q$ は逆量子化された変換係数であり、levelは量子化された変換係数であり、 $offset_{IQ}$ はオフセット値 $= 1 \ll (M - 10 + B)$ であり、 $shift1 = (M - 9 + B)$ であり、gは以下の式(25)に示されるように定義される。

$$g = [f_0, \dots, f_5]^T = [40, 45, 51, 57, 64, 72]^T \quad (25)$$

【0372】

50

本開示の技法によれば、ビデオエンコーダ20は、 $(\log_2(W)+\log_2(H))$ が奇数か偶数かに応じて、異なる量子化行列(すなわち、fの複数のバージョン)を使用し得る。同様に、ビデオデコーダ30は、 $(\log_2(W)+\log_2(H))$ が奇数か偶数かに応じて、異なる逆量子化行列(すなわち、gの複数のバージョン)を使用し得る。gの例は次のように定義される。

[40,45,51,57,64,72], //合計が偶数のとき

[7240,8145,9231,10317,11584,13032], //合計が奇数のとき

合計が偶数である場合のgの各々の対応する値は、181と乗じられることに留意されたい。この例では、量子化段階の前または後に乗算または除算の処理を実行する必要はなく、それは、

【 0 3 7 3 】

10

【 数 5 1 】

$\sqrt{2}$

【 0 3 7 4 】

の補償がすでにgにおいて考慮されているからである。

【 0 3 7 5 】

さらに、式(22)および(24)において、量子化行列fおよびgにおいて選択される値は、量子化パラメータQPに基づいて選択される。量子化行列fおよびgにおける選択される値は、本明細書では量子化行列係数と呼ばれ得る。本開示のいくつかの例では、ビデオエンコーダ20およびビデオデコーダ30は、量子化パラメータに基づいて、また $(\log_2(W)+\log_2(H))$ が奇数か偶数かに基づいて、量子化行列係数を選択し得る。

20

【 0 3 7 6 】

上で簡単に説明され、図2Aおよび図2Bにおいて示されたように、HEVCでは、ビデオコーダは常に、再帰的なz走査順序でCUのTUを処理する。したがって、図2Aに示されるように、TU「a」の変換係数に対応するデータは、TU「b」の変換係数に対応するデータより前にビットストリームにおいて現れ、以下同様である。本開示の第17の技法は、インターモードまたはcomb_modeまたはイントラモードでコーディングされるが非正方形の区分を伴うCUに対して、変換係数のコーディング順序が、再帰的なz走査を常に使用する代わりにPUコーディング順序に依存することを提案する。第17の技法のいくつかの例では、1つのPU内のすべての係数が、別のPUの中の係数をコーディングする前に一緒にコーディングされるものとする。したがって、ビデオエンコーダ20のために、CUのPUの1つの中のすべての係数が、CUのPUの別の1つの係数を符号化する前に一緒に符号化される。同様に、この例では、ビデオデコーダ30のために、CUの1つのPUの中のすべての係数が、CUのPUの別の1つの係数を復号する前に一緒に復号される。

30

【 0 3 7 7 】

この例では、PUの変換係数は、その変換ブロックがPUの予測ブロックのエリア内にある、TUの変換係数である。たとえば、 16×16 のCUのPUの予測ブロックの左上の角の座標を、そのCUのコーディングブロックの左上の角に対して相対的に(0, 0)とし、そのPUの予測ブロックの右下の角の座標を(7, 15)とする。さらに、この例では、そのCUのTUの変換ブロックの左上の角の座標を(4, 0)とし、そのTUの変換ブロックの右下の角の座標を(7, 15)とする。この例では、TUの変換係数はPUの変換係数である。しかしながら、この例では、TUの変換ブロックの左上の角が(8, 0)であり、TUの変換ブロックの右下の角が(15, 7)である場合、TUの変換係数はPUの変換係数ではない。

40

【 0 3 7 8 】

たとえば、図2Aに関して、CU40が、CU40の中心を通過して垂直にCU40を区分する2つのPUを有すると仮定する。したがって、CU40の第1のPUの変換係数は、TU「a」、「b」、「c」、「d」、および「f」の変換係数を含む。CU40の第2のPUの変換係数は、TU「e」、「g」、「h」、「i」、および「j」の変換係数を含む。この例では、ビデオエンコーダ20は、TU「f」の変換係数を表すデータに続いて、TU「e」の変換係数を表すデータを含め得る。

50

対照的に、HEVCでは、TU「f」の変換係数を表すデータは、CU40のPUの形状およびサイズとは関係なく、TU「e」の変換係数を表すデータの後に来る。言い換えると、1つのPUが複数のTUを含むとき、深度優先の横断を用いた再帰的なZ走査が、PU内のこれらの係数をコーディングするために適用される。図2を例として採用すると、区分サイズが $N \times 2N$ に等しい場合、コーディング順序は、a、b、c、d、f、e、g、h、i、jであり得る。

【0379】

第17の技法のいくつかの例は、変換深度が0に等しくないとき、すなわち変換サイズがPUサイズより大きくないときにだけ適用可能である。上で言及されたAMPは、HEVCにおいて定義される4つの場合だけではなく、他の非対称の区分を含み得ることに留意されたい。

【0380】

上で簡単に言及されたように、HEVCにおけるICの設計は正方形PUのみをサポートする。本開示以前には、非正方形PUのためのICパラメータaおよびbをどのように導出するかは知られていなかった。本開示の第18の技法は、非正方形PUとともにICが使用されることを可能にする。たとえば、ビデオエンコーダ20は、ICを使用してビデオデータのピクチャの現在のPUの非正方形予測ブロックを生成し得る。加えて、ビデオエンコーダ20は、予測ブロックに基づいて残差データを生成し得る。たとえば、ビデオエンコーダ20は、残差データの各々のそれぞれのサンプルが、現在のCUのコーディングブロックのそれぞれのサンプルと予測ブロックの対応するそれぞれのサンプルとの差に等しくなるように、残差データを生成し得る。さらに、ビデオエンコーダ20は、残差データに基づくデータを含むビットストリームを出力し得る。たとえば、ビデオエンコーダ20は、残差データに変換を適用して係数ブロックを生成し、係数ブロックの係数を量子化し、量子化された係数の各々を表す1つまたは複数のシンタックス要素をビットストリームに含め得る。この例では、ビデオエンコーダ20は、各々の量子化された係数のためにシンタックス要素のうちの1つまたは複数をエントロピー符号化し得る。他の例では、ビデオエンコーダ20は変換および/または量子化の適用をスキップし得る。

【0381】

さらに、上で与えられたICに関する例の1つまたは複数によれば、ビデオデコーダ30は、ICを使用して、ビデオデータのピクチャの現在のCUの現在のPUの非正方形予測ブロックを生成し得る。加えて、ビデオデコーダ30は、予測ブロックに基づいて、ピクチャのブロック(たとえば、コーディングブロック)を再構築し得る。たとえば、ビデオデコーダ30は、対応する残差サンプルに予測ブロックのサンプルを加算することによって、ブロックのサンプルを再構築し得る。

【0382】

第18の技法の例では、ICを使用して非正方形予測ブロックを生成することの一部として、ビデオコーダ(たとえば、ビデオエンコーダ20および/またはビデオデコーダ30)は、次のように予測ブロックのサンプルを決定し得る。

$$p(i, j) = a * r(i + dv_x, j + dv_y + b), \text{ただし } (i, j) \in PU_c$$

ここで PU_c は現在のPUであり、 (i, j) は予測ブロックの中の画素の座標であり、 (dv_x, dv_y) は PU_c のベクトル(たとえば、視差ベクトル)である。 $p(i, j)$ は PU_c の予測であり、 r はビュー間参照ピクチャであり、 a は第1のICパラメータであり、 b は第2のICパラメータである。さらに、ICを使用して非正方形予測ブロックを生成することの一部として、ビデオコーダは、次のように第1のICパラメータを計算し得る。

【0383】

10

20

30

40

【数 5 2】

$$a = \frac{2N \cdot \sum_{i=0}^{2N-1} \text{Rec}_{\text{neig}}(i) \cdot \text{Rec}_{\text{refneig}}(i) - \sum_{i=0}^{2N-1} \text{Rec}_{\text{neig}}(i) \cdot \sum_{i=0}^{2N-1} \text{Rec}_{\text{refneig}}(i)}{2N \cdot \sum_{i=0}^{2N-1} \text{Rec}_{\text{refneig}}(i) \cdot \text{Rec}_{\text{refneig}}(i) - \left(\sum_{i=0}^{2N-1} \text{Rec}_{\text{refneig}}(i) \right)^2}$$

【0 3 8 4】

10

加えて、ビデオコードは、次のように第2のICパラメータを計算し得る。

【0 3 8 5】

【数 5 3】

$$b = \frac{\sum_{i=0}^{2N-1} \text{Rec}_{\text{neig}}(i) - a \cdot \sum_{i=0}^{2N-1} \text{Rec}_{\text{refneig}}(i)}{2N}$$

【0 3 8 6】

20

上の式において、 Rec_{neig} および $\text{Rec}_{\text{refneig}}$ はそれぞれ現在のCUおよび参照ブロックの隣接画素セットを表し、 $2N$ は Rec_{neig} および $\text{Rec}_{\text{refneig}}$ における画素数を表し、現在のCUは $N \times N$ に等しいサイズを有する。他の例は、上で示される式に対する変形を使用し得る。

【0 3 8 7】

さらに、第18の技法の例では、サイズが $K \times L$ に等しい(K は L と等しくない)1つの非正方形PUに対してICが有効にされるとき、パラメータは様々な方法で導出され得る。たとえば、式(16)および式(17)を使用して線形モデルパラメータを計算するとき、PUの長辺と短辺の両方に位置する画素は、異なるサブサンプリング比などの、異なる方法でサブサンプリングされ得る。しかしながら、2つの辺を合わせた画素の総数は 2^m に等しくなければならないことが要求され得る(ここで m は整数であり、その値はブロックサイズに依存し得る)。したがって、この例では、 Rec_{neig} は現在のCUのすぐ上かつ現在のCUのすぐ左にある画素のサブセットであり、 $\text{Rec}_{\text{refneig}}$ は参照ブロックのすぐ上かつ参照ブロックのすぐ左にある画素のサブセットであり、 Rec_{neig} と $\text{Rec}_{\text{refneig}}$ の中の画素の総数は 2_m に等しく、ここで m は整数である。このサブサンプリングプロセスは、デシメーションまたは補間されたサンプリングであり得る。

30

【0 3 8 8】

ICのためのパラメータを導出する別の例において、式(16)および式(17)を使用して線形モデルパラメータを計算するとき、非正方形PUの短辺における境界の画素は、アップサンプリングされた境界における画素の数がより長い境界における画素の数(すなわち、 $\max(K, L)$)に等しくなるように、アップサンプリングされる。このアップサンプリングプロセスは、デュプリケータまたは補間されたサンプリングであり得る。したがって、この例では、ICを使用して予測ブロックを生成することの一部として、ビデオコードは、現在のCUの左辺と上辺のうちの短い方の中のアップサンプリングされた画素を含むように、 Rec_{neig} を生成し得る。加えて、この例では、ビデオコードは、参照ブロックの左辺と上辺のうちの短い方の中のアップサンプリングされた画素を含むように、 $\text{Rec}_{\text{refneig}}$ を生成し得る。

40

【0 3 8 9】

代わりに、PUの長辺と短辺の両方に位置する画素はアップサンプリングされることがあり、アップサンプリング比は異なることがある。したがって、この例では、ICを使用して予測ブロックを生成することの一部として、ビデオコードは、現在のCUの左辺と上辺のう

50

ちの長い方の中のアップサンプリングされた画素を含むように、 Rec_{neig} を生成し得る。加えて、ビデオコードは、参照ブロックの左辺と上辺のうちの長い方の中のアップサンプリングされた画素を含むように、 $Rec_{refneig}$ を生成し得る。しかしながら、2つの辺を合わせた画素の総数は 2^m に等しくなければならないことが要求され得る(ここで m は整数であり、 m は輝度成分および色差成分に対して異なり得る)。

【0390】

さらに、ICのためのパラメータを導出するいくつかの例では、境界画素のためのサブサンプリング/アップサンプリングの様々な方法が適用され得る。一例では、サブサンプリング/アップサンプリング方法は、PUサイズに(すなわち、 K および L の値に)依存する。したがって、ビデオコードは、現在のPUのサイズに基づいて、 Rec_{neig} および $Rec_{refneig}$ を生成するために使用するサブサンプリング方法またはアップサンプリング方法を決定し得る。別の例では、サブサンプリング/アップサンプリングの方法は、シーケンスパラメータセット、ピクチャパラメータセット、および/またはスライスヘッダにおいてシグナリングされ得る。したがって、いくつかの例では、 Rec_{neig} および $Rec_{refneig}$ を生成するために使用するサブサンプリング方法を示すシンタックス要素を、ビデオエンコーダ20はビットストリームに含めることができ、ビデオデコーダ30はビットストリームから取得することができる。いくつかの例では、アップサンプリングされた画素を生成するために使用するアップサンプリング方法を示すシンタックス要素を、ビデオエンコーダ20はビットストリームに含めることができ、ビデオデコーダ30はビットストリームから取得することができる。

10

20

【0391】

ICのパラメータを導出するいくつかの例では、アップサンプリング/ダウンサンプリング(またはサブサンプリング)は、暗黙的に実施される。たとえば、左側の境界または/および上側の境界の、式(16)および式(17)における合計値が、係数 S と乗じられ、または係数 S で割られ得る。 S の値は左側の境界または/および上側の境界における画素数の比に依存し得る。

【0392】

第18の技法のいくつかの例では、同じサブサンプリング/アップサンプリング方法は、参照ブロックの境界画素(すなわち、 $Rec_{refneig}$)にも適用され得る。たとえば、 Rec_{neig} および $Rec_{refneig}$ の両方をサブサンプリングするために、デシメーションが使用され得る。

30

【0393】

さらに、本開示の特定の技法によれば、LMが正方形PUに対して有効にされるとき、輝度境界画素および色差境界画素がまず、たとえば式(16)および(17)を使用してパラメータを導出するためにサブサンプリングされ得る。サブサンプリング方法は、シーケンスパラメータセット、ピクチャパラメータセット、またはスライスヘッダにおいて、事前に定義され、またはシグナリングされ得る。サブサンプリング方法は、予測単位サイズに依存し得る。

【0394】

したがって、ビデオコード(たとえば、ビデオエンコーダ20またはビデオデコーダ30)は、線形モデル予測演算を実行して、PUのサブサンプリングされた再構築された輝度サンプルから、現在のPUの予測色差ブロックを予測することができる。加えて、ビデオコードは、予測色差ブロックに一部基づいて、ピクチャのブロックを再構築し得る。線形モデル予測動作を実行することの一部として、ビデオコードは、予測色差サンプルが、併置輝度サンプルと乗じられた第1のパラメータと、第2のパラメータとを足したものに等しくなるように、予測色差サンプルを取得することができ、ここで第1のパラメータは

40

【0395】

【数 5 4】

$$\alpha = \frac{I \sum x_i \cdot y_i - \sum x_i \cdot \sum y_i}{I \sum x_i \cdot x_i - \sum x_i \cdot \sum x_i}$$

【0396】

に等しく、第2のパラメータは

$$= (\sum y_i - \sum x_i) / I$$

10

に等しい。上の式において、Iはサブサンプリング方法に従って決定された現在のPUの左の境界および上の境界におけるサンプルのサブセットの中の参照サンプルの数であり、 x_i はサンプリングされ再構築された輝度参照サンプルであり、 y_i は再構築された色差参照サンプルである。この例のいくつかの事例では、サブサンプリング方法を示すシンタックス要素を、ビデオエンコーダ20はビットストリームに含めることができ、ビデオデコーダ30はビットストリームから取得することができる。この例のいくつかの事例では、ビデオエンコーダ20およびビデオデコーダ30は、現在のPUのサイズに基づいて、サブサンプリング方法を決定し得る。

【0397】

様々な例が説明された。本開示の具体的な例は、別々に、または互いに組み合わせて使用され得る。

20

【0398】

図24は、本開示の技法を実装し得る例示的なビデオエンコーダ20を示すブロック図である。図24は説明のために提供され、広く例示されるとともに本開示で説明されるような技法の限定と見なされるべきでない。説明のために、本開示は、HEVCコーディングの文脈でビデオエンコーダ20を説明する。しかしながら、本開示の技法は、他のコーディング規格または方法に適用可能であり得る。

【0399】

ビデオエンコーダ20は処理回路を含み、ビデオエンコーダ20は、本開示において説明される例示的な技法のうちの1つまたは複数を実行するように構成される。そのような処理回路は、固定された機能および/またはプログラム可能回路を含み得る。たとえば、ビデオエンコーダ20は集積回路を含み、図24に示される様々なユニットは、回路バスと相互接続されるハードウェア回路ブロックとして形成され得る。これらのハードウェア回路ブロックは別個の回路ブロックであってよく、またはユニットのうちの2つ以上が共通のハードウェア回路ブロックへと組み合わされてよい。ハードウェア回路ブロックは、算術論理演算装置(ALU)、初等関数演算装置(EFU)、ならびに、AND、OR、NAND、NOR、XOR、XNOR、および他の同様の論理ブロックなどの論理ブロックなどの、演算ブロックを形成する電氣的構成要素の組合せとして形成され得る。

30

【0400】

図24の例において、ビデオエンコーダ20は、予測処理ユニット200、ビデオデータメモリ201、残差生成ユニット202、変換処理ユニット204、量子化ユニット206、逆量子化ユニット208、逆変換処理ユニット210、再構築ユニット212、フィルタユニット214、復号ピクチャバッファ216、およびエントロピー符号化ユニット218を含む。予測処理ユニット200は、インター予測処理ユニット220およびイントラ予測処理ユニット226を含む。インター予測処理ユニット220は、動き推定ユニットおよび動き補償ユニット(図示せず)を含み得る。いくつかの例では、予測処理ユニット200は、本開示の照明補償技法を実行する。いくつかの例では、予測処理ユニット200は、本開示のLM技法を実行して非正方形色差予測ブロックを生成する。さらに、いくつかの例では、予測処理ユニット200は、本開示のIC技法を実行して非正方形予測ブロックを生成する。

40

【0401】

50

ビデオデータメモリ201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するように構成され得る。ビデオデータメモリ201に記憶されるビデオデータは、たとえば、ビデオソース18(図1)から取得され得る。復号ピクチャバッファ216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201および復号ピクチャバッファ216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)、磁気抵抗性RAM(MRAM)、抵抗性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201および復号ピクチャバッファ216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201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ってよく、または、これらのコンポーネントに対してオフチップであってよい。

【0402】

ビデオエンコーダ20は、ビデオデータを受信する。ビデオエンコーダ20は、ビデオデータのピクチャのスライスの中の各CTUを符号化し得る。CTUの各々は、ピクチャの、等しいサイズの輝度コーディングツリーブロック(CTB)、および対応するCTBと関連付けられ得る。CTUを符号化することの一部として、予測処理ユニット200は、四分木区分を実行して、CTUのCTBを次第に小さくなるブロックに分割し得る。より小さいブロックは、CUのコーディングブロックであり得る。たとえば、予測処理ユニット200は、CTUと関連付けられるCTBを4つの等しいサイズのサブブロックに区分することができ、サブブロックの1つまたは複数を4つの等しいサイズのサブサブブロックに区分することができ、以下同様である。

【0403】

ビデオエンコーダ20は、CTUのCUを符号化して、CUの符号化された表現(すなわち、コーディングされたCU)を生成し得る。CUを符号化することの一部として、予測処理ユニット200は、CUの1つまたは複数のPUの間でCUと関連付けられるコーディングブロックを区分し得る。したがって、各PUは、輝度予測ブロックおよび対応する色差予測ブロックと関連付けられ得る。ビデオエンコーダ20およびビデオデコーダ30は、様々なサイズを有するPUをサポートし得る。上で示されたように、CUのサイズは、CUの輝度コーディングブロックのサイズを指すことがあり、PUのサイズは、PUの輝度予測ブロックのサイズを指すことがある。特定のCUのサイズが $2N \times 2N$ であると仮定すると、ビデオエンコーダ20およびビデオデコーダ30は、イントラ予測に対して $2N \times 2N$ または $N \times N$ というPUサイズ、およびインター予測に対して $2N \times 2N$ 、 $2N \times N$ 、 $N \times 2N$ 、 $N \times N$ 、または同様の対称的なPUサイズをサポートし得る。ビデオエンコーダ20およびビデオデコーダ30はまた、インター予測に対して、 $2N \times nU$ 、 $2N \times nD$ 、 $nL \times 2N$ 、および $nR \times 2N$ というPUサイズのための非対称区分をサポートし得る。

【0404】

インター予測処理ユニット220は、インター予測をCUの各PUに対して実行することによって、PUの予測データを生成し得る。PUの予測データは、PUの予測ブロックおよびPUの動き情報を含み得る。インター予測処理ユニット220は、PUがIスライスの中にあるか、Pスライスの中にあるか、またはBスライスの中にあるかに応じて、CUのPUに対して異なる動作を実行し得る。Iスライスの中では、すべてのPUがイントラ予測される。したがって、PUがIスライスの中にある場合、インター予測処理ユニット220は、インター予測をPUに対して実行しない。したがって、Iモードで符号化されるブロックの場合、予測されるブロックは、同じフレーム内の以前符号化された隣接ブロックからの空間予測を使用して形成される。PUがPスライスの中にある場合、インター予測処理ユニット220は、単方向インター予測を使用してPUの予測ブロックを生成することができる。PUがBスライスの中にある場合、インター予測処理ユニット220は、単方向または双方向インター予測を使用してPUの予測ブロックを生成することができる。

【0405】

イントラ予測処理ユニット226は、イントラ予測をPUに対して実行することによって、P

10

20

30

40

50

Uの予測データを生成し得る。PUの予測データは、PUの予測ブロックおよび様々なシンタックス要素を含み得る。イントラ予測処理ユニット226は、Iスライス、Pスライス、およびBスライスの中のPUに対して、イントラ予測を実行し得る。

【0406】

イントラ予測をPUに対して実行するために、イントラ予測処理ユニット226は、複数のイントラ予測モードを使用して、PUの予測データの複数のセットを生成し得る。イントラ予測処理ユニット226は、隣接PUのサンプルブロックからのサンプルを使用して、PUの予測ブロックを生成し得る。PU、CU、およびCTUに対して左から右、上から下への符号化順序を仮定すると、隣接PUは、PUの上、右上、左上、または左であり得る。イントラ予測処理ユニット226は、様々な数のイントラ予測モード、たとえば、33個の方向性イントラ予測モードを使用し得る。いくつかの例では、イントラ予測モードの数は、PUと関連付けられる領域のサイズに依存し得る。

10

【0407】

予測処理ユニット200は、PUのためにインター予測処理ユニット220によって生成される予測データ、またはPUのためにイントラ予測処理ユニット226によって生成される予測データの中から、CUのPUの予測データを選択し得る。いくつかの例では、予測処理ユニット200は、予測データのセットのレート/ひずみの尺度に基づいて、CUのPUの予測データを選択する。選択される予測データの予測ブロックは、選択予測ブロックと本明細書で呼ばれることがある。

【0408】

20

残差生成ユニット202は、CUのコーディングブロック(たとえば、輝度コーディングブロック、Cbコーディングブロック、およびCrコーディングブロック)およびCUのPUの選択された予測ブロック(たとえば、予測輝度ブロック、予測Cbブロック、および予測Crブロック)に基づいて、CUの残差ブロック(たとえば、輝度残差ブロック、Cb残差ブロック、およびCr残差ブロック)を生成し得る。たとえば、残差生成ユニット202は、残差ブロックの中の各サンプルがCUのコーディングブロックの中のサンプルとCUのPUの対応する選択された予測ブロックの中の対応するサンプルとの間の差に等しい値を有するように、CUの残差ブロックを生成し得る。

【0409】

変換処理ユニット204は、区分(たとえば、四分木区分)を実行して、CUと関連付けられた残差ブロックをCUのTUと関連付けられた変換ブロックに区分し得る。したがって、TUは、輝度変換ブロックおよび2つの色差変換ブロックと関連付けられ得る。CUのTUの輝度変換ブロックおよび色差変換ブロックのサイズおよび位置は、CUのPUの予測ブロックのサイズおよび位置に基づいても基づかなくてもよい。「残差四分木」(RQT)と呼ばれる四分木構造が、領域の各々と関連付けられたノードを含み得る。CUのTUは、RQTのリーフノードに相当し得る。

30

【0410】

いくつかの例では、変換処理ユニット204は、2つ(および/または4つ)の子ノードを有するノードを含む残差木構造を決定するための、本開示の技法を実行し得る。たとえば、ビデオデータメモリ201はビデオデータを受信することができ、変換処理ユニット204は木構造に基づいてビデオデータのCUをCUのTUへと区分し得る。この例では、木構造に基づいてCUをCUのTUへと区分することの一部として、変換処理ユニット204は、木構造の中のノードが木構造の中で丁度2つの子ノードを有すると決定し得る。いくつかの事例では、変換処理ユニット204はさらに、木構造の中の第2のノードが木構造の中で丁度4つの子ノードを有すると決定し得る。CUのTUのうちの少なくとも1つに対して、変換処理ユニット204は、変換をTUの残差ブロックに適用して、TUの変換係数のブロックを生成し得る。

40

【0411】

変換処理ユニット204は、1つまたは複数の変換をTUの変換ブロックに適用することによって、CUのTUごとに変換係数ブロックを生成し得る。変換処理ユニット204は、TUと関連付けられた変換ブロックに様々な変換を適用し得る。たとえば、変換処理ユニット204は

50

、離散コサイン変換(DCT)、方向変換、または概念的に類似の変換を、変換ブロックに適用し得る。いくつかの例では、変換処理ユニット204は、変換ブロックに変換を適用しない。そのような例では、変換ブロックは変換係数ブロックとして扱われてよい。いくつかの例では、変換処理ユニット204は、本開示のEMT技法を実行する。

【0412】

量子化ユニット206は、係数ブロックの中の変換係数を量子化し得る。量子化プロセスは、変換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。たとえば、nビットの変換係数は、量子化の間にmビットの変換係数に切り捨てられてよく、nはmよりも大きい。量子化ユニット206は、CUと関連付けられる量子化パラメータ(QP)値に基づいて、CUのTUと関連付けられる係数ブロックを量子化し得る。ビデオエンコーダ20は、CUと関連付けられるQP値を調整することによって、CUと関連付けられた係数ブロックに適用される量子化の程度を調整し得る。量子化が情報の損失をもたらすことがあり、したがって、量子化された変換係数は精度が元の変換係数よりも低いことがある。

10

【0413】

いくつかの例において、量子化ユニット206は、変換係数のブロックの各々のそれぞれの変換係数が、

【0414】

【数55】

$$\sqrt{2}$$

20

【0415】

の近似と乗じられたそれぞれの逆量子化された変換係数に基づくように、変換係数を修正する。この例では、変換係数を修正した後で、量子化ユニット206は、量子化プロセスをCUの非正方形PUの修正された変換係数に適用する。

【0416】

逆量子化ユニット208および逆変換処理ユニット210は、それぞれ、逆量子化および逆変換を係数ブロックに適用して、係数ブロックから残差ブロックを再構築し得る。再構築ユニット212は、予測処理ユニット200によって生成された1つまたは複数の予測ブロックからの対応するサンプルに、再構築された残差ブロックを加算して、TUと関連付けられる再構築された変換ブロックを生成し得る。このようにしてCUのTUごとに変換ブロックを再構築することによって、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築し得る。

30

【0417】

フィルタユニット214は、1つまたは複数のデブロッキング動作を実行して、CUと関連付けられるコーディングブロックにおけるブロッキングアーティファクトを低減し得る。フィルタユニット214が1つまたは複数のデブロッキング動作を再構築されたコーディングブロックに対して実行した後、復号ピクチャバッファ216は、再構築されたコーディングブロックを記憶し得る。インター予測処理ユニット220は、インター予測を他のピクチャのPUに対して実行するために、再構築されたコーディングブロックを含む参照ピクチャを使用し得る。加えて、イントラ予測処理ユニット226は、CUと同じピクチャの中の他のPUに対してイントラ予測を実行するために、復号ピクチャバッファ216の中の再構築されたコーディングブロックを使用し得る。

40

【0418】

エントロピー符号化ユニット218は、ビデオエンコーダ20の他の機能構成要素からデータを受け取り得る。たとえば、エントロピー符号化ユニット218は、係数ブロックを量子化ユニット206から受け取ることができ、シンタックス要素を予測処理ユニット200から受け取ることができる。エントロピー符号化ユニット218は、エントロピー符号化されたデータを生成するために、1つまたは複数のエントロピー符号化動作をデータに対して実行

50

し得る。たとえば、エントロピー符号化ユニット218は、CABAC動作、コンテキスト適応型可変長コーディング(CAVLC)動作、可変長-可変長(V2V)コーディング動作、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)動作、確率区間区分エントロピー(PIPE)コーディング動作、指数ゴロム符号化動作、または別のタイプのエントロピー符号化動作を、データに対して実行し得る。ビデオエンコーダ20は、エントロピー符号化ユニット218によって生成された、エントロピー符号化されたデータを含むビットストリームを出力し得る。たとえば、ビットストリームは、CUのためのRQTを表すデータを含み得る。

【0419】

図25は、本開示の技法を実施するように構成される例示的なビデオデコーダ30を示すブロック図である。図25は説明のために提供され、広く例示され本開示で説明されるような技法を限定するものではない。説明のために、本開示は、HEVCコーディングの文脈においてビデオデコーダ30を説明する。しかしながら、本開示の技法は、他のコーディング規格または方法に適用可能であり得る。

【0420】

ビデオデコーダ30は処理回路を含み、ビデオデコーダ30は、本開示において説明される例示的な技法のうちの1つまたは複数を実行するように構成される。たとえば、ビデオデコーダ30は集積回路を含み、図25に示される様々なユニットは、回路バスと相互接続されるハードウェア回路ブロックとして形成され得る。これらのハードウェア回路ブロックは別個の回路ブロックであってよく、または2つ以上のユニットが共通のハードウェア回路ブロックへと組み合わせられてよい。ハードウェア回路ブロックは、算術論理演算装置(ALU)、初等関数演算装置(EFU)、ならびに、AND、OR、NAND、NOR、XOR、XNOR、および他の同様の論理ブロックなどの論理ブロックなどの、演算ブロックを形成する電氣的構成要素の組合せとして形成され得る。

【0421】

いくつかの例では、図25に示されるユニットのうちの1つまたは複数は、処理回路上で実行されるソフトウェアユニットによって提供され得る。そのような例では、これらのソフトウェアのためのオブジェクトコードがメモリに記憶される。オペレーティングシステムは、ビデオデコーダ30に、オブジェクトコードを取り出させ、オブジェクトコードを実行させることができ、オブジェクトコードは、ビデオデコーダ30に、例示的な技法を実施するための動作を実行させる。いくつかの例では、ソフトウェアユニットは、ビデオデコーダ30が起動時に実行するファームウェアであり得る。したがって、ビデオデコーダ30は、例示的な技法を実行するハードウェアを有する構造的構成要素であり、または、ハードウェアを例示的な技法の実行専用にするためのハードウェア上で実行されるソフトウェア/ファームウェアを有する。

【0422】

図25の例において、ビデオデコーダ30は、エントロピー復号ユニット250、ビデオデータメモリ251、予測処理ユニット252、逆量子化ユニット254、逆変換処理ユニット256、再構築ユニット258、フィルタユニット260、および復号ピクチャバッファ262を含む。予測処理ユニット252は、動き補償ユニット264およびイントラ予測処理ユニット266を含む。他の例では、ビデオデコーダ30は、より多数の、より少数の、または異なる機能構成要素を含み得る。いくつかの例では、予測処理ユニット266は、本開示の照明補償技法を実行する。いくつかの例では、予測処理ユニット266は、本開示のLM技法を実行する。

【0423】

ビデオデータメモリ251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符号化されたビデオデータを記憶し得る。ビデオデータメモリ251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251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262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、または出力のための、参照ビデオデータを記憶する参照ピクチャメモリであってもよい。ビデオデータメモリ251および復号ピクチャバッファ262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)、磁気抵抗性RAM(MRAM)、抵抗性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251および復号ピクチャバッファ262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251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。

10

【0424】

ビデオデータメモリ251は、ビットストリームの符号化されたビデオデータ(たとえば、NALユニット)を受信し記憶する。エントロピー復号ユニット250は、符号化されたビデオデータ(たとえば、NALユニット)をビデオデータメモリ251から受信することができ、NALユニットを構文解析して、シンタックス要素を取得することができる。エントロピー復号ユニット250は、NALユニットの中のエントロピー符号化されたシンタックス要素をエントロピー復号し得る。予測処理ユニット252、逆量子化ユニット254、逆変換処理ユニット256、再構築ユニット258、およびフィルタユニット260は、ビットストリームから抽出されたシンタックス要素に基づいて、復号されたビデオデータを生成し得る。エントロピー復号ユニット250は、エントロピー符号化ユニット218のプロセスとは全般に逆のプロセスを実行し得る。

20

【0425】

シンタックス要素をビットストリームから取得することに加えて、ビデオデコーダ30は、区分されていないCUに対して再構築動作を実行し得る。CUに対して再構築動作を実行するために、ビデオデコーダ30は、CUの各TUに対して再構築動作を実行し得る。CUのTUごとに再構築動作を実行することによって、ビデオデコーダ30は、CUの残差ブロックを再構築し得る。

【0426】

CUのTUに対して再構築動作を実行することの一部として、逆量子化ユニット254は、TUと関連付けられる係数ブロックを逆量子化(inverse quantize)、すなわち、逆量子化(de-quantize)し得る。逆量子化ユニット254が係数ブロックを逆量子化した後、逆変換処理ユニット256は、TUと関連付けられる残差ブロックを生成するために、1つまたは複数の逆変換を係数ブロックに適用し得る。たとえば、逆変換処理ユニット256は、逆DCT、逆整数変換、逆カルーネンレーベ変換(KLT)、逆回転変換、逆方向変換、または別の逆変換を係数ブロックに適用し得る。いくつかの例では、逆変換処理ユニット256は、本開示のEMT技法を実行する。

30

【0427】

本開示のいくつかの例によれば、逆量子化ユニット254は、逆量子化プロセスをビデオデータのCUの非正方形TUの変換係数に適用し得る。さらに、逆量子化プロセスを変換係数に適用した後で、逆量子化ユニット254は、逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が、

40

【0428】

【数56】

$$\sqrt{2}$$

【0429】

50

の近似で割られたそれぞれの逆量子化された変換係数に基づくように、逆量子化された変換係数を修正し得る。

【0430】

いくつかの例では、逆変換処理ユニット256は、2つ(および/または4つ)の子ノードを有するノードを含む残差木構造を決定するための、本開示の技法を適用し得る。たとえば、逆変換処理ユニット256は、木構造に基づいてCUのTUへとビデオデータのCUが区分されると決定し得る。この例では、木構造に基づいてCUがCUのTUへと区分されると決定することの一部として、逆変換処理ユニット256は、木構造の中のノードが木構造の中で丁度2つの子ノードを有すると決定し得る。いくつかの例では、逆変換処理ユニット256は、木構造の中の第2のノードが木構造の中で丁度4つの子ノードを有すると決定し得る。さらに、この例では、CUのTUのうちの少なくとも1つに対して、逆変換処理ユニット256は、変換をTUの係数ブロックに適用して、TUの残差ブロックを生成し得る。

10

【0431】

PUがイントラ予測を使用して符号化されている場合、イントラ予測処理ユニット266は、イントラ予測を実行して、PUの予測ブロックを生成し得る。イントラ予測処理ユニット266は、イントラ予測モードを使用して、空間的に隣接するブロックのサンプルに基づいて、PUの予測ブロックを生成し得る。イントラ予測処理ユニット266は、ビットストリームから取得された1つまたは複数のシンタックス要素に基づいて、PUのイントラ予測モードを決定し得る。

【0432】

20

PUがインター予測を使用して符号化される場合、エンターピー復号ユニット250は、PUの動き情報を決定し得る。動き補償ユニット264は、PUの動き情報に基づいて、1つまたは複数の参照ブロックを決定し得る。動き補償ユニット264は、1つまたは複数の参照ブロックに基づいて、PUの予測ブロック(たとえば、予測輝度ブロック、予測Cbブロック、および予測Crブロック)を生成し得る。

【0433】

再構築ユニット258は、CUのTUのための変換ブロック(たとえば、輝度変換ブロック、Cb変換ブロック、およびCr変換ブロック)、ならびにCUのPUの予測ブロック(たとえば、輝度ブロック、Cbブロック、およびCrブロック)、すなわちイントラ予測データまたはインター予測データのいずれかを適宜使用して、CUのコーディングブロック(たとえば、輝度コーディングブロック、Cbコーディングブロック、およびCrコーディングブロック)を再構築し得る。たとえば、再構築ユニット258は、予測ブロック(たとえば、輝度予測ブロック、Cb予測ブロック、およびCr予測ブロック)の対応するサンプルに、変換ブロック(たとえば、輝度変換ブロック、Cb変換ブロック、およびCr変換ブロック)のサンプルを加算して、CUのコーディングブロック(たとえば、輝度コーディングブロック、Cbコーディングブロック、およびCrコーディングブロック)を再構築し得る。

30

【0434】

フィルタユニット260は、CUのコーディングブロックに1つまたは複数のフィルタを適用し得る。たとえば、フィルタユニット260は、デブロック動作を実行して、CUのコーディングブロックと関連付けられるブロックアーティファクトを低減し得る。ビデオデコーダ30は、CUのコーディングブロックを復号ピクチャバッファ262に記憶し得る。したがって、復号ピクチャバッファ262は、ビデオデータの復号されたブロックを記憶し得る。復号ピクチャバッファ262は、その後の動き補償、イントラ予測、および図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提示のために、参照ピクチャを提供し得る。たとえば、ビデオデコーダ30は、復号ピクチャバッファ262の中のブロックに基づいて、他のCUのPUのためにイントラ予測動作またはインター予測動作を実行し得る。

40

【0435】

図26は、本開示の技法による、LMベースの符号化をサポートする例示的なビデオエンコーダ20を示すブロック図である。図26の例では、ビデオエンコーダ20の構成要素は、図24のビデオエンコーダ20の対応する構成要素と同じ方式で動作する。しかしながら、図26の

50

ビデオエンコーダ20は、LMベースの符号化ユニット222も含む。

【0436】

LMベースの符号化ユニット222は、本開示の他の箇所で説明される例に従って、LM予測符号化を実行し得る。たとえば、逆量子化ユニット208、逆変換処理ユニット210、再構築ユニット212、およびフィルタユニット214は、輝度参照サンプルのセット、色差参照サンプルのセットを再構築することができ、非正方形PUの輝度サンプルも再構築することができる。LMベースの符号化ユニット222は、非正方形PUの長辺に隣接する輝度参照サンプルのセットの中の輝度参照サンプルの総数が非正方形PUの短辺に隣接する輝度参照サンプルのセットの輝度参照サンプルの総数と同じになるように、輝度参照サンプルのセットをダウンサンプリングまたはサブサンプリングし得る。加えて、LMベースの符号化ユニット222は、第1のパラメータが

$$(y_i - x_i) / l$$

に等しいように第1のパラメータを決定することができ、ここで l は輝度参照サンプルのセットの中の参照サンプルの総数であり、 x_i は輝度参照サンプルのセットの中の i 番目の輝度参照サンプルであり、 y_i は色差参照サンプルのセットの中の i 番目の色差参照サンプルである。非正方形PUの予測色差ブロックの各々のそれぞれの色差サンプルに対して、LMベースの符号化ユニット222は、それぞれの色差サンプルの値が、それぞれの色差サンプルに対応するそれぞれの再構築された輝度サンプルと乗じられた第2のパラメータと、第1のパラメータとを足したものに等しくなるように、それぞれの色差サンプルの値を決定することができ、それぞれの色差サンプルに対応する再構築された輝度サンプルは、非正方形PUの再構築された輝度サンプルのうちの1つである。LMベースの符号化ユニット222は、第2のパラメータが

【0437】

【数57】

$$\frac{l \sum x_i \cdot y_i - \sum x_i \cdot \sum y_i}{l \sum x_i \cdot x_i - \sum x_i \cdot \sum x_i}$$

【0438】

に等しいように第2のパラメータを決定することができる。

【0439】

LMベースの符号化ユニット222は、予測ブロックを残差生成ユニット202に出力し得る。残差生成ユニット202は、予測ブロックおよび色差ブロックから残差ブロックを生成する。得られる残差ブロックは、変換処理ユニット103によって変換され、量子化ユニット206によって量子化され、エントロピー符号化ユニット218によってエントロピー符号化される。この結果が次いでビットストリームを介してシグナリングされ、ビデオデコーダ30がビットストリームの中の情報を使用して色差ブロックを再構築し得る。

【0440】

図27は、本開示の技法による、LMベースの復号をサポートする例示的なビデオデコーダ30を示すブロック図である。図27の例では、ビデオデコーダ30の構成要素は、図27のビデオデコーダ30の対応する構成要素と同じ方式で動作する。しかしながら、図27のビデオデコーダ30は、LMベースの復号ユニット265も含む。

【0441】

本開示の様々な例によれば、ビデオデコーダ30は、本開示の他の箇所で与えられる例に従ってLMベースのコーディングを実行するように構成され得る。たとえば、逆量子化ユニット254、逆変換処理ユニット256、再構築ユニット258、およびフィルタユニット260は、輝度参照サンプルのセット、色差参照サンプルのセットを再構築することができ、非正方形PUの輝度サンプルも再構築することができる。LMベースの復号ユニット265は、非正方

10

20

30

40

50

形PUの長辺に隣接する輝度参照サンプルのセットの中の輝度参照サンプルの総数が非正方形PUの短辺に隣接する輝度参照サンプルのセットの輝度参照サンプルの総数と同じになるように、輝度参照サンプルのセットをダウンサンプリングまたはサブサンプリングし得る。加えて、LMベースの復号ユニット265は、第1のパラメータが

$$(y_i - x_i) / l$$

に等しいように第1のパラメータを決定することができ、ここで l は輝度参照サンプルのセットの中の参照サンプルの総数であり、 x_i は輝度参照サンプルのセットの中の i 番目の輝度参照サンプルであり、 y_i は色差参照サンプルのセットの中の i 番目の色差参照サンプルである。非正方形PUの予測色差ブロックの各々のそれぞれの色差サンプルに対して、LMベースの復号ユニット266は、それぞれの色差サンプルの値が、それぞれの色差サンプルに対応するそれぞれの再構築された輝度サンプルと乗じられた第2のパラメータと、第1のパラメータとを足したものに等しくなるように、それぞれの色差サンプルの値を決定することができ、それぞれの色差サンプルに対応する再構築された輝度サンプルは、非正方形PUの再構築された輝度サンプルのうちの1つである。LMベースの復号ユニット266は、第2のパラメータが

【0 4 4 2】

【数 5 8】

$$\frac{l \sum x_i \cdot y_i - \sum x_i \cdot \sum y_i}{l \sum x_i \cdot x_i - \sum x_i \cdot \sum x_i}$$

【0 4 4 3】

に等しいように第2のパラメータを決定することができる。

【0 4 4 4】

LMベースの復号ユニット265は、予測ブロックを再構築ユニット258に出力し得る。再構築ユニット258はまた、残差ブロックを受信する(たとえば、残差ブロックのためのビットストリームの中の情報がエントロピー復号ユニット250によりエントロピー復号され、逆量子化ユニット254により逆量子化され、逆変換処理ユニット256により逆変換された後で)。再構築ユニット258は、残差ブロックを予測ブロックと加算して色差ブロックを再構築する。

【0 4 4 5】

図28は、本開示のLMベースのコーディング技法による、ビデオエンコーダ20の例示的な動作を示すフローチャートである。本開示のフローチャートは、例として与えられる。本開示の範囲内の他の例は、より多数の、より少数の、または異なる活動を含み得る。本開示の範囲内の他の例は、異なる順序の、または並列に実行される活動を含み得る。

【0 4 4 6】

図28の例では、ビデオエンコーダ20がビデオデータを受信し得る(300)。たとえば、ビデオエンコーダ20は、ビデオソース18(図1)、または別のソースからビデオデータを受信し得る。

【0 4 4 7】

加えて、図28の例では、ビデオエンコーダ20は、輝度参照サンプルのセットおよび色差参照サンプルのセットを再構築することができる(302)。輝度参照サンプルのセットは、ビデオデータの現在のピクチャの非正方形輝度ブロックの上辺に隣接する上側輝度サンプルと、非正方形輝度ブロックの左辺に隣接する左側輝度サンプルとを備え得る。いくつかの例では、非正方形輝度ブロックは非正方形PUの輝度予測ブロックである。色差参照サンプルのセットは、非正方形色差ブロックの上辺に隣接する色差サンプルと、非正方形色差ブロックの左辺に隣接する色差サンプルとを備え得る。いくつかの例では、非正方形輝度ブロックは非正方形PUの輝度予測ブロックである。

【0448】

さらに、ビデオエンコーダ20は、非正方形輝度ブロックの輝度サンプルを再構築することができる(304)。たとえば、ビデオエンコーダ20は、本開示の他の箇所で説明されたようにCUの輝度残差サンプルを生成し得る。この例では、ビデオエンコーダ20は、輝度残差サンプルの対応するサンプルに非正方形輝度ブロックの輝度予測ブロックのサンプルを加算して、非正方形輝度ブロックの輝度サンプルを再構築し得る。

【0449】

いくつかの例では、ビデオエンコーダ20は、非正方形輝度ブロックの輝度サンプルをダウンサンプリングまたはサブサンプリングし得る。非正方形輝度ブロックの輝度サンプルをダウンサンプリングまたはサブサンプリングすることによって、ビデオエンコーダ20は、色差予測ブロック(たとえば、輝度ブロックと同じPUの色差予測ブロック)の各色差サンプルに対して1つの輝度サンプルを有する、輝度サンプルのダウンサンプリングまたはサブサンプリングされたセットを取得し得る。ビデオエンコーダ20は、現在のピクチャのカラーフォーマットが4:4:4ではないと決定したことに応答して、非正方形輝度ブロックの輝度サンプルをダウンサンプリングまたはサブサンプリングし得る。

【0450】

加えて、ビデオエンコーダ20は、非正方形輝度ブロックの長辺に隣接する輝度参照サンプルのセットの中の輝度参照サンプルの総数が非正方形輝度ブロックの短辺に隣接する輝度参照サンプルのセットの輝度参照サンプルの総数と同じになるように、輝度参照サンプルのセットをダウンサンプリングまたはサブサンプリングすることができる(306)。ビデオエンコーダ20は、本開示の他の箇所で説明された技法に従って、輝度参照サンプルのセットをダウンサンプリングまたはサブサンプリングし得る。たとえば、ビデオエンコーダ20は、非正方形輝度ブロックの長辺に隣接する輝度参照サンプルのセットが非正方形輝度ブロックの短辺に隣接する輝度参照サンプルのセットの輝度参照サンプルの総数と同じになるように、輝度参照サンプルのセットをデシメーションすることができる。いくつかの例では、ビデオエンコーダ20は、左の参照サンプルまたは上の参照サンプルのうちで、輝度の左の境界と輝度の上の境界のうちの長い方に相当するものをダウンサンプリングまたはサブサンプリングしてよいが、輝度ブロックの左の境界と輝度ブロックの上の境界のうちの短い方はダウンサンプリングまたはサブサンプリングしなくてよい。いくつかの例では、輝度参照サンプルのセットの中の参照サンプルの総数は 2^m に等しく、ここで m は非正方形輝度ブロックの高さおよび/または幅に依存する整数である。

【0451】

いくつかの例では、ビデオエンコーダ20はまた、非正方形色差ブロックの長辺に隣接する色差参照サンプルのセットの中の色差参照サンプルの総数が非正方形色差ブロックの短辺に隣接する色差参照サンプルのセットの色差参照サンプルの総数と同じになるように、色差参照サンプルのセットをダウンサンプリングまたはサブサンプリングすることができる。

【0452】

図28の活動(308)において、ビデオエンコーダ20は、第1のパラメータが

$$(y_i - x_i)/l$$

に基づくように第1のパラメータ()を決定することができる。上の式において、 l は輝度参照サンプルのセットの中の参照サンプルの総数であり、 x_i は輝度参照サンプルのセットの中の i 番目の輝度参照サンプルであり、 y_i は色差参照サンプルのセットの中の i 番目の色差参照サンプルである。ビデオエンコーダ20は、上の式を直接使用する、または追加の定数もしくは係数を含むものなどの上の式の変形を使用するという意味で、上の式に基づいて第1のパラメータを決定し得る。

【0453】

いくつかの例では、ビデオエンコーダ20はまた、第2のパラメータが

【0454】

10

20

30

40

【数 5 9】

$$\frac{I \sum x_i \cdot y_i - \sum x_i \cdot \sum y_i}{I \sum x_i \cdot x_i - \sum x_i \cdot \sum x_i}$$

【0 4 5 5】

に基づくように第2のパラメータ()を決定し得る。ビデオエンコーダ20は、上の式を直接使用する、または追加の定数または係数を含むものなどの上の式の変形を使用するという意味で、上の式に基づいて第2のパラメータを決定し得る。

10

【0 4 5 6】

加えて、図28の例では、予測色差ブロックの各々のそれぞれの色差サンプルに対して、ビデオエンコーダ20は、それぞれの色差サンプルの値が、それぞれの色差サンプルに対応するそれぞれの再構築された輝度サンプルと乗じられた第2のパラメータと、第1のパラメータとを足したものに等しくなるように、それぞれの色差サンプルの値を決定することができる(310)。それぞれの色差サンプルに対応する再構築された輝度サンプルは、非正方形輝度ブロックの再構築された輝度サンプルのうちの1つである。

【0 4 5 7】

さらに、ビデオエンコーダ20は、予測色差ブロックに基づいて、残差データを取得することができる(312)。たとえば、ビデオエンコーダ20は、非正方形予測ブロックの色差ブロックのサンプルとCUの色差コーディングブロックのサンプルとの差に等しい、残差データの色差サンプルの値を決定し得る。

20

【0 4 5 8】

加えて、ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、残差データを表すデータを含めることができる(314)。たとえば、ビデオエンコーダ20は、残差データに1つまたは複数の変換を適用して1つまたは複数の係数ブロックを生成し、係数ブロックを量子化し、変換係数が0ではないかどうか、変換係数が1より大きいかどうか、変換係数が2より大きいかどうか、変換係数の符号、および変換係数の剰余を示すシンタックス要素を生成し得る。この例では、ビデオエンコーダ20は、これらのシンタックス要素のうちの1つまたは複数のCABACコーディングを適用し、得られる値をビットストリームに含め得る。

30

【0 4 5 9】

図29は、本開示のLMベースのコーディング技法による、ビデオデコーダ30の例示的な動作を示すフローチャートである。図29の例では、ビデオデコーダ30は、ビデオデータの符号化された表現を備えるビットストリームを受信する(350)。

【0 4 6 0】

さらに、図29の例では、ビデオデコーダ30は、輝度参照サンプルのセットおよび色差参照サンプルのセットを再構築する(352)。輝度参照サンプルのセットは、ビデオデータの現在のピクチャの非正方形輝度ブロックの上辺に隣接する上側輝度サンプルと、非正方形輝度ブロックの左辺に隣接する左側輝度サンプルとを備える。いくつかの例では、非正方形輝度ブロックは非正方形PUの輝度予測ブロックである。色差参照サンプルのセットは、非正方形色差ブロックの上辺に隣接する色差サンプルと、非正方形色差ブロックの左辺に隣接する色差サンプルとを備える。いくつかの例では、非正方形輝度ブロックは非正方形PUの輝度予測ブロックである。

40

【0 4 6 1】

ビデオデコーダ30は、非正方形輝度ブロックの輝度サンプルを再構築することができる(354)。たとえば、非正方形輝度ブロックの輝度サンプルを再構築することの一部として、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、非正方形輝度ブロックの輝度予測ブロックを生成し得る。加えて、この例では、ビデオデコーダ30は、非正

50

方形予測ブロックの輝度予測ブロックのサンプルを対応する残差サンプルに加算して、輝度サンプルを再構築し得る。

【0462】

いくつかの例では、ビデオデコーダ30は、非正方形輝度ブロックの輝度サンプルをダウンサンプリングまたはサブサンプリングし得る。非正方形輝度ブロックの輝度サンプルをダウンサンプリングまたはサブサンプリングすることによって、ビデオデコーダ30は、色差予測ブロック(たとえば、輝度ブロックと同じPUの色差予測ブロック)の各色差サンプルに対して1つの輝度サンプルを有する、輝度サンプルのダウンサンプリングまたはサブサンプリングされたセットを取得し得る。ビデオデコーダ30は、現在のピクチャのカラーフォーマットが4:4:4ではないと決定したことに応答して、非正方形輝度ブロックの輝度サ

10

【0463】

さらに、図29の例では、ビデオデコーダ30は、非正方形予測ブロックの長辺に隣接する輝度参照サンプルのセットの中の輝度参照サンプルの総数が非正方形予測ブロックの短辺に隣接する輝度参照サンプルのセットの輝度参照サンプルの総数と同じになるように、輝度参照サンプルのセットをダウンサンプリングまたはサブサンプリングすることができる(356)。ビデオデコーダ30は、本開示の他の箇所で説明された技法に従って、輝度参照サンプルのセットをダウンサンプリングまたはサブサンプリングし得る。たとえば、ビデオデコーダ30は、非正方形輝度ブロックの長辺に隣接する輝度参照サンプルのセットが非正方形輝度ブロックの短辺に隣接する輝度参照サンプルのセットの輝度参照サンプルの総数と同じになるように、輝度参照サンプルのセットをデシメーションすることができる。いくつかの例では、ビデオデコーダ30は、左の参照サンプルまたは上の参照サンプルのうちで、輝度ブロックの左の境界と輝度ブロックの上の境界のうちの長い方に相当するものをダウンサンプリングまたはサブサンプリングしてよいが、輝度ブロックの左の境界と輝度ブロックの上の境界のうちの短い方はダウンサンプリングまたはサブサンプリングしなくてよい。いくつかの例では、輝度参照サンプルのセットの中の参照サンプルの総数は 2^m に等しく、ここで m は非正方形輝度ブロックの高さおよび/または幅に依存する整数である。

20

【0464】

いくつかの例では、ビデオデコーダ30はまた、非正方形色差ブロックの長辺に隣接する色差参照サンプルのセットの中の色差参照サンプルの総数が非正方形色差ブロックの短辺に隣接する色差参照サンプルのセットの色差参照サンプルの総数と同じになるように、色差参照サンプルのセットをダウンサンプリングまたはサブサンプリングすることができる。

30

【0465】

加えて、図29の活動(358)において、ビデオデコーダ30は、第1のパラメータが

$$(y_i - \cdot x_i) / l$$

に基づくように第1のパラメータ()を決定することができる。上の式において、 l は輝度参照サンプルのセットの中の参照サンプルの総数であり、 x_i は輝度参照サンプルのセットの中の i 番目の輝度参照サンプルであり、 y_i は色差参照サンプルのセットの中の i 番目の色差参照サンプルである。本開示では、ビデオエンコーダ20および/またはビデオデコーダ30は、式を直接使用する、または追加の定数もしくは係数を含むものなどの式の変形を使用するという意味で、式に基づいて値を決定し得る。

40

【0466】

いくつかの例では、ビデオデコーダ30はまた、第2のパラメータが

【0467】

【数 6 0】

$$\frac{I \sum x_i \cdot y_i - \sum x_i \cdot \sum y_i}{I \sum x_i \cdot x_i - \sum x_i \cdot \sum x_i}$$

【0 4 6 8】

に基づくように第2のパラメータ()を決定し得る。

【0 4 6 9】

10

図29の例では、予測色差ブロックの各々のそれぞれの色差サンプルに対して、ビデオデコード30は、それぞれの色差サンプルの値が、それぞれの色差サンプルに対応するそれぞれの再構築された輝度サンプルと乗じられた第2のパラメータと、第1のパラメータとを足したものに等しくなるように、それぞれの色差サンプルの値を決定することができる(360)。この例では、それぞれの色差サンプルに対応する再構築された輝度サンプルは、非正方形輝度ブロックの再構築された輝度サンプルのうちの1つである。

【0 4 7 0】

さらに、ビデオデコード30は、予測色差ブロックに一部基づいて、コーディングブロックを再構築することができる(362)。たとえば、ビデオデコード30は、予測色差ブロックのサンプルをCUの対応する残差色差サンプルに加算して、CUのコーディングブロックのサンプルを決定し得る。

20

【0 4 7 1】

図30は、本開示の量子化技法による、ビデオエンコーダ20の例示的な動作を示すフローチャートである。図30の例では、ビデオエンコーダ20がビデオデータを受信する(400)。さらに、ビデオエンコーダ20は、残差ブロックの各残差サンプルがCUのコーディングブロックの対応するサンプルとCUのPUの予測ブロックとの差を示すように、CUの非正方形TUの残差ブロックを生成することができる(402)。

【0 4 7 2】

ビデオエンコーダ20は、変換を残差ブロックに適用して変換係数のブロックを生成することができる(404)。たとえば、ビデオエンコーダ20は、DCT変換を残差ブロックに適用し得る。加えて、ビデオエンコーダ20は、変換係数のブロックの各々のそれぞれの変換係数が、

30

【0 4 7 3】

【数 6 1】

$$\sqrt{2}$$

【0 4 7 4】

の近似と乗じられたそれぞれの逆量子化された変換係数に基づくように、変換係数を修正することができる(406)。たとえば、ビデオエンコーダ20は、各々のそれぞれの変換係数が、

40

【0 4 7 5】

【数 6 2】

$$\sqrt{2}$$

【0 4 7 6】

の近似と乗じられたそれぞれの変換係数の元の値に等しくなるように、変換係数を修正し得る。本開示では、

50

【 0 4 7 7 】

【 数 6 3 】

$$\sqrt{2}$$

【 0 4 7 8 】

の近似は、

【 0 4 7 9 】

【 数 6 4 】

10

$$\sqrt{2}$$

【 0 4 8 0 】

の表現(たとえば、

【 0 4 8 1 】

【 数 6 5 】

$$\sqrt{2}$$

20

【 0 4 8 2 】

の浮動小数点数表現)であり得る。いくつかの例では、各々のそれぞれの変換係数が、

【 0 4 8 3 】

【 数 6 6 】

$$\sqrt{2}$$

【 0 4 8 4 】

の近似と乗じられたそれぞれの変換係数の元の値に等しくなるように変換係数を修正することは、1つまたは複数の数学的演算を実行して、

【 0 4 8 5 】

【 数 6 7 】

30

$$\sqrt{2}$$

【 0 4 8 6 】

と乗じられた変換係数を近似する値を決定することを備え得る。

【 0 4 8 7 】

40

非正方形TUのサイズが $K \times L$ であるいくつかの例では、変換を残差ブロックに適用することの一部として、ビデオエンコーダ20は、残差ブロックにサイズが $N \times N$ の変換を適用することができ、ここで $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$ であり、 $((\log_2(K) + \log_2(L))$ は奇数である。たとえば、ビデオエンコーダ20は、上の式(18)に示されるように、 N 点の1次元DCT変換を残差ブロックの行および列に適用し得る。

【 0 4 8 8 】

さらに、図30の例では、変換係数を修正した後で、ビデオエンコーダ20は、量子化プロセスをCUの非正方形予測ブロックの修正された変換係数に適用することができる(408)。たとえば、ビデオエンコーダ20は、上の式(22)に記載されたように、修正された変換係数を量子化し得る。

50

【 0 4 8 9 】

ビデオエンコーダ20は、ビデオデータの符号化された表現を備えるビットストリームに、量子化された変換係数に基づくデータを含めることができる(410)。たとえば、ビデオエンコーダ20は、量子化された変換係数が0ではないかどうか、量子化された変換係数が1より大きいかどうか、量子化された変換係数が2より大きいかどうか、量子化された変換係数の符号、および量子化された変換係数の剰余を示すシンタックス要素を生成し得る。この例では、ビデオエンコーダ20は、これらのシンタックス要素のうちの1つまたは複数にCABACコーディングを適用し、得られる値をビットストリームに含め得る。

【 0 4 9 0 】

非正方形TUのサイズが $K \times L$ であるいくつかの例では、ビデオエンコーダ20は、 $((\log_2(K) + \log_2(L)))$ が奇数であることに基づいて、逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が、

【 0 4 9 1 】

【 数 6 8 】

$$\sqrt{2}$$

【 0 4 9 2 】

の近似と乗じられたそれぞれの逆量子化された変換係数に基づくように、逆量子化された変換係数を修正し得る。そのような例では、 $((\log_2(K) + \log_2(L)))$ が偶数であるとき、ビデオエンコーダ20は、逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が、

【 0 4 9 3 】

【 数 6 9 】

$$\sqrt{2}$$

【 0 4 9 4 】

の近似と乗じられたそれぞれの逆量子化された変換係数に基づくように、逆量子化された変換係数を修正しない。

【 0 4 9 5 】

図31は、本開示の量子化技法による、ビデオデコーダ30の例示的な動作を示すフローチャートである。図31の例では、ビデオデコーダ30は、ビデオデータの符号化された表現を備えるビットストリームを受信することができる(450)。さらに、ビデオデコーダ30は、逆量子化プロセスをビデオデータのCUの非正方形TUの変換係数に適用することができる(452)。たとえば、ビデオデコーダ30は、上の式(24)を適用することによって、変換係数を逆量子化(dequantize)(すなわち、逆量子化(inverse quantize))し得る。

【 0 4 9 6 】

逆量子化プロセスを変換係数に適用した後で、ビデオデコーダ30は、逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が、

【 0 4 9 7 】

【 数 7 0 】

$$\sqrt{2}$$

【 0 4 9 8 】

の近似で割られたそれぞれの逆量子化された変換係数に基づくように、逆量子化された変

換係数を修正することができる(454)。たとえば、ビデオデコーダ30は、各々のそれぞれの修正された変換係数が、

【 0 4 9 9 】

【 数 7 1 】

$$\sqrt{2}$$

【 0 5 0 0 】

の近似で割られた変換係数と等しいと決定し得る。本開示では、

【 0 5 0 1 】

【 数 7 2 】

$$\sqrt{2}$$

【 0 5 0 2 】

の近似は、

【 0 5 0 3 】

【 数 7 3 】

$$\sqrt{2}$$

【 0 5 0 4 】

の表現(たとえば、

【 0 5 0 5 】

【 数 7 4 】

$$\sqrt{2}$$

【 0 5 0 6 】

の浮動小数点数表現)であり得る。いくつかの例では、各々のそれぞれの変換係数が、

【 0 5 0 7 】

【 数 7 5 】

$$\sqrt{2}$$

【 0 5 0 8 】

の近似と乗じられたそれぞれの変換係数の元の値に等しくなるように変換係数を修正することは、1つまたは複数の数学的演算を実行して、

【 0 5 0 9 】

【 数 7 6 】

$$\sqrt{2}$$

【 0 5 1 0 】

10

20

30

40

50

で割られた変換係数を近似する値を決定することを備え得る。

【0511】

さらに、ビデオデコーダ30は、逆変換を修正された逆量子化された変換係数に適用して、残差ブロックを再構築することができる(456)。たとえば、ビデオデコーダ30は、修正された逆量子化された変換係数に逆変換を適用するために、変換行列Cの転置(または整数精度で表されるその近似)とともに式(18)を適用し得る。非正方形TUのサイズが $K \times L$ であるいくつかの例では、変換を残差ブロックに適用することの一部として、ビデオデコーダ30は、残差ブロックにサイズが $N \times N$ の変換を適用することができ、ここで $\log_2(N \times N) = ((\log_2(K) + \log_2(L)) \gg 1) \ll 1$)であり、 $((\log_2(K) + \log_2(L)))$ は奇数である。ビデオデコーダ30は、予測ブロックのサンプルをCUのTUの残差ブロックの対応するサンプルに加算することによって、コーディングブロックのサンプルを再構築することができる(458)。

10

【0512】

非正方形TUのサイズが $K \times L$ であるいくつかの例では、ビデオデコーダ30は、 $((\log_2(K) + \log_2(L)))$ が奇数であることに基づいて、逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が、

【0513】

【数77】

$$\sqrt{2}$$

20

【0514】

の近似で割られたそれぞれの逆量子化された変換係数に基づくように、逆量子化された変換係数を修正し得る。そのような例では、 $((\log_2(K) + \log_2(L)))$ が偶数であるとき、ビデオデコーダ30は、逆量子化された変換係数の各々のそれぞれの逆量子化された変換係数が、

【0515】

【数78】

$$\sqrt{2}$$

30

【0516】

の近似で割られたそれぞれの逆量子化された変換係数に基づくように、逆量子化された変換係数を修正しない。

【0517】

図32は、ICを使用する本開示の技法による、ビデオエンコーダ20の例示的な動作を示すフローチャートである。図32の例では、ビデオエンコーダ20がビデオデータを受信する(500)。たとえば、ビデオエンコーダ20は、ビデオソース18(図1)、または別の箇所からビデオデータを受信し得る。さらに、ビデオエンコーダ20は、ICを使用してビデオデータの現在のピクチャの現在のCUの現在のPUの非正方形予測ブロックを生成することができる(502)。

40

【0518】

ICを使用して非正方形予測ブロックを生成することの一部として、ビデオエンコーダ20は、現在のPUのベクトルに基づいて、参照ピクチャの中の参照ブロックを決定することができる(504)。いくつかの例では、ベクトルは視差ベクトルであり、参照ピクチャはビュー間参照ピクチャである。いくつかの例では、ベクトルは動きベクトルであり、参照ピクチャは時間動きベクトルである。参照ブロックおよび非正方形予測ブロックは、サイズおよび形状が同じであり得る。いくつかの例では、現在のPUのベクトルに基づいて参照ブロックを決定するために、ビデオエンコーダ20は、ベクトルの水平成分を非正方形予測ブロックの左上の角のx座標に加算し、ベクトルの垂直成分を非正方形予測ブロックの左上の角のy座標に加算することによって、参照ブロックの左上の角の参照ピクチャにおける場

50

所を決定し得る。この例では、参照ブロックの左上の角の示される場所が整数画素の参照ピクチャにおける場所を示さない場合、ビデオエンコーダ20は、参照ブロックのサンプルを補間して参照ブロックを決定し得る。

【0519】

さらに、図32の例では、ICを使用して非正方形予測ブロックを生成することの一部として、ビデオエンコーダ20は、参照サンプルの第1のセットをサブサンプリングして、第1のサブサンプリング比を伴う参照サンプルの第1のサブサンプリングされたセットを生成することができる(506)。この例では、参照サンプルの第1のセットの中の参照サンプルの総数は 2^m に等しくなく、参照サンプルの第1のサブサンプリングされたセットの中の参照サンプルの総数は 2^m に等しい。さらに、この例では、参照サンプルの第1のセットは、非正方形予測ブロックの外側の、非正方形予測ブロックの左辺および上辺に沿ったサンプルを備え、 m は整数である。

10

【0520】

加えて、ICを使用して非正方形予測ブロックを生成することの一部として、ビデオエンコーダ20は、参照サンプルの第2のセットをサブサンプリングして、第2のサブサンプリング比を伴う参照サンプルの第2のサブサンプリングされたセットを生成することができる(508)。第1のサブサンプリング比は、第2のサブサンプリング比と同じであることがあり、または異なることがある。この例では、参照サンプルの第2のセットの中の参照サンプルの総数は 2^m に等しくなく、参照サンプルの第2のサブサンプリングされたセットの中の参照サンプルの総数は 2^m に等しい。さらに、この例では、参照サンプルの第2のセットは、参照ブロックの外側の、参照ブロックの左辺および上辺に沿ったサンプルを備える。

20

【0521】

活動(506)および(508)において、ビデオエンコーダ20は、様々な方法でサブサンプリングを実行し得る。たとえば、ビデオエンコーダ20は、デシメーションを使用してサブサンプリングを実行することができる。ビデオエンコーダ20がデシメーションを使用してサブサンプリングを実行する例では、ビデオエンコーダ20は、通常の間隔でサンプルを除去し(たとえば、1つおきにサンプルを除去し)、残りのサンプルの値を変更せずにサンプルの数を減らすことができる。したがって、この例では、ビデオエンコーダ20は、参照サンプルの第1のセットをデシメーションして参照サンプルの第1のサブサンプリングされたセットを生成することと、参照サンプルの第2のセットをデシメーションして参照サンプルの第2のサブサンプリングされたセットを生成することのうちの、少なくとも1つを実行し得る。

30

【0522】

別の例では、ビデオエンコーダ20は、補間を使用してサブサンプリングを実行することができる。ビデオエンコーダ20が補間を使用してサブサンプリングを実行する例では、隣り合うサンプルのそれぞれのペアに対して、ビデオエンコーダ20は、それぞれのペアのサンプルの間の値を補間することができ、サンプルのサブサンプリングされたセットにその補間された値を含めることができる。したがって、この例では、ビデオエンコーダ20は、参照サンプルの第1のセットの補間されたサンプリングを実行して参照サンプルの第1のサブサンプリングされたセットを生成することと、参照サンプルの第2のセットの補間されたサンプリングを実行して参照サンプルの第2のサブサンプリングされたセットを生成することのうちの、少なくとも1つを実行し得る。

40

【0523】

別の例では、ビデオエンコーダ20は、ビットストリームの中のシンタックス要素により示されるサブサンプリング方法を使用してサブサンプリングを実行し得る。したがって、この例では、ビデオエンコーダ20は、ビットストリームに、サブサンプリング方法を示すシンタックス要素を含め得る。この例では、ビデオエンコーダ20は、示されるサブサンプリング方法を使用して参照サンプルの第1のセットをサブサンプリングし参照サンプルの第1のサブサンプリングされたセットを生成することと、示されるサブサンプリング方法を使用して参照サンプルの第2のセットをサブサンプリングし参照サンプルの第2のサブサ

50

ンプリングされたセットを生成することのうちの、少なくとも1つを実行し得る。

【0524】

別の例では、ビデオエンコーダ20は、現在のPUのサイズに基づいて、サブサンプリング方法を決定し得る。この例では、ビデオエンコーダ20は、決定されたサブサンプリング方法を使用して参照サンプルの第1のセットをサブサンプリングし参照サンプルの第1のサブサンプリングされたセットを生成することと、決定されたサブサンプリング方法を使用して参照サンプルの第2のセットをサブサンプリングし参照サンプルの第2のサブサンプリングされたセットを生成することのうちの、少なくとも1つを実行し得る。

【0525】

ICを使用して非正方形予測ブロックを生成することの一部として、図32の活動(510)において、ビデオエンコーダ20は、参照サンプルの第1のサブサンプリングされたセットの中の参照サンプルの総数および参照サンプルの第2のサブサンプリングされたセットの中の参照サンプルの総数、参照サンプルの第1のサブサンプリングされたセット、ならびに参照サンプルの第2のサブサンプリングされたセットに基づいて、第1のICパラメータを決定し得る。たとえば、ビデオエンコーダ20は、第1のICパラメータが

【0526】

【数79】

$$\frac{\sum_{i=0}^{2N-1} \text{Rec}_{\text{neigh}}(i) - a \cdot \sum_{i=0}^{2N-1} \text{Rec}_{\text{refneigh}}(i)}{2N}$$

10

20

【0527】

に基づくように第1のICパラメータを決定し得る。上の式において、2Nは参照サンプルの第1のサブサンプリングされたセットの中の参照サンプルの総数および参照サンプルの第2のサブサンプリングされたセットの中の参照サンプルの総数を表し、 $\text{Rec}_{\text{neigh}}(i)$ は参照サンプルの第1のサブサンプリングされたセットの中のi番目の参照サンプルを表し、 $\text{Rec}_{\text{refneigh}}(i)$ は参照サンプルの第2のサブサンプリングされたセットの中のi番目の参照サンプルを表す。

【0528】

いくつかの例では、ビデオエンコーダ20は、第2のICパラメータが

【0529】

【数80】

$$\frac{2N \cdot \sum_{i=0}^{2N-1} \text{Rec}_{\text{neigh}}(i) \cdot \text{Rec}_{\text{refneigh}}(i) - \sum_{i=0}^{2N-1} \text{Rec}_{\text{neigh}}(i) \cdot \sum_{i=0}^{2N-1} \text{Rec}_{\text{refneigh}}(i)}{2N \cdot \sum_{i=0}^{2N-1} \text{Rec}_{\text{refneigh}}(i) \cdot \text{Rec}_{\text{neigh}}(i) - \left(\sum_{i=0}^{2N-1} \text{Rec}_{\text{refneigh}}(i) \right)^2}$$

30

【0530】

に基づくように第2のICパラメータを決定し得る。

40

【0531】

さらに、ICを使用して非正方形予測ブロックを生成することの一部として、図32の活動(512)において、ビデオエンコーダ20は、第1のICパラメータに基づいて非正方形予測ブロックのサンプルを決定し得る。たとえば、サンプルは現在のピクチャの左上の角に対する相対的な座標(i, j)にあることがあり、ビデオエンコーダ20はサンプルが以下に基づくようにサンプルを決定し得る。

$$a * r(i + dv_x, j + dv_y + b)$$

上の式において、bは第1のICパラメータであり、aは第2のICパラメータであり、rは参照ピクチャであり、 dv_x は現在のPUのベクトル(たとえば、視差ベクトル、動きベクトル)の水平成分であり、 dv_y は現在のPUのベクトルの垂直成分である。

50

【0532】

図32の例では、ビデオエンコーダ20は、非正方形予測ブロックに基づいて残差データを生成することができる(514)。たとえば、ビデオエンコーダ20は、残差データのサンプルが、非正方形予測ブロックのサンプルと現在のCUのコーディングブロックのサンプルとの差に等しくなるように、残差データを生成し得る。加えて、ビデオエンコーダ20は、残差データに基づくデータを含むビットストリームを出力することができる(516)。たとえば、ビデオエンコーダ20は、残差データを示すエントロピー符号化されたシンタックス要素(たとえば、1より大きいこと、2より大きいこと、剰余などを示すシンタックス要素)を含むビットストリームを出力し得る。

【0533】

10

図33は、ICを使用する本開示の技法による、ビデオデータを符号化するためのビデオデコーダ30の例示的な動作を示すフローチャートである。図32の例では、ビデオデコーダ30は、ビデオデータの符号化された表現を備えるビットストリーム受信する(550)。さらに、ビデオデコーダ30は、ICを使用してビデオデータの現在のピクチャの現在のCUの現在のPUの非正方形予測ブロックを生成する(552)。

【0534】

ICを使用して非正方形予測ブロックを生成することの一部として、ビデオデコーダ30は、現在のPUのベクトルに基づいて、参照ピクチャの中の参照ブロックを決定することができる(554)。いくつかの例では、ベクトルは視差ベクトルであり、参照ピクチャはビュー間参照ピクチャである。いくつかの例では、ベクトルは動きベクトルであり、参照ピクチャは時間動きピクチャである。参照ブロックおよび非正方形予測ブロックは、サイズおよび形状が同じである。現在のPUの視差ベクトルに基づいて参照ブロックを決定するために、ビデオデコーダ30は、ベクトルの水平成分を非正方形予測ブロックの左上の角のx座標に加算し、ベクトルの垂直成分を非正方形予測ブロックの左上の角のy座標に加算することによって、参照ブロックの左上の角の参照ピクチャにおける場所を決定し得る。この例では、参照ブロックの左上の角の示される場所が整数画素の参照ピクチャにおける場所を示さない場合、ビデオデコーダ30は、参照ブロックのサンプルを補間して参照ブロックを決定し得る。

20

【0535】

さらに、ICを使用して非正方形予測ブロックを生成することの一部として、ビデオデコーダ30は、参照サンプルの第1のセットをサブサンプリングして、第1のサブサンプリング比を伴う参照サンプルの第1のサブサンプリングされたセットを生成することができる(556)。この例では、参照サンプルの第1のセットの中の参照サンプルの総数は 2^m に等しくなく、参照サンプルの第1のサブサンプリングされたセットの中の参照サンプルの総数は 2^m に等しい。この例では、参照サンプルの第1のセットは、非正方形予測ブロックの外側の、非正方形予測ブロックの左辺および上辺に沿ったサンプルを備えることがあり、mは整数である。

30

【0536】

加えて、ICを使用して非正方形予測ブロックを生成することの一部として、ビデオデコーダ30は、参照サンプルの第2のセットをサブサンプリングして、第2のサブサンプリング比を伴う参照サンプルの第2のサブサンプリングされたセットを生成することができる(558)。第1のサブサンプリング比は、第2のサブサンプリング比と同じであることがあり、または異なることがある。この例では、第2のセットの中の参照サンプルの総数は 2^m に等しくなく、参照サンプルの第2のサブサンプリングされたセットの中の参照サンプルの総数は 2^m に等しい。さらに、この例では、参照サンプルの第2のセットは、参照ブロックの外側の、参照ブロックの左辺および上辺に沿ったサンプルを備え得る。

40

【0537】

活動(556)および(558)において、ビデオデコーダ30は、様々な方法でサブサンプリングを実行し得る。たとえば、ビデオデコーダ30は、デシメーションを使用してサブサンプリングを実行することができる。ビデオデコーダ30がデシメーションを使用してサブサンプリングを実行することができる。

50

リングを実行する例では、ビデオデコーダ30は、通常の間隔でサンプルを除去し(たとえば、1つおきにサンプルを除去し)、残りのサンプルの値を変更せずにサンプルの数を減らすことができる。したがって、この例では、ビデオデコーダ30は、参照サンプルの第1のセットをデシメーションして参照サンプルの第1のサブサンプリングされたセットを生成することと、参照サンプルの第2のセットをデシメーションして参照サンプルの第2のサブサンプリングされたセットを生成することのうちの、少なくとも1つを実行し得る。

【0538】

別の例では、ビデオデコーダ30は、補間を使用してサブサンプリングを実行することができる。ビデオデコーダ30が補間を使用してサブサンプリングを実行する例では、隣り合うサンプルのそれぞれのペアに対して、ビデオデコーダ30は、それぞれのペアのサンプルの間の値を補間することができ、サンプルのサブサンプリングされたセットにその補間された値を含めることができる。したがって、この例では、ビデオデコーダ30は、参照サンプルの第1のセットの補間されたサンプリングを実行して参照サンプルの第1のサブサンプリングされたセットを生成することと、参照サンプルの第2のセットの補間されたサンプリングを実行して参照サンプルの第2のサブサンプリングされたセットを生成することのうちの、少なくとも1つを実行し得る。

【0539】

別の例では、ビデオデコーダ30は、ビットストリームの中のシンタックス要素により示されるサブサンプリング方法を使用してサブサンプリングを実行し得る。したがって、この例では、ビデオデコーダ30は、ビットストリームから、サブサンプリング方法を示すシンタックス要素を取得し得る。この例では、ビデオデコーダ30は、示されるサブサンプリング方法を使用して参照サンプルの第1のセットをサブサンプリングし参照サンプルの第1のサブサンプリングされたセットを生成することと、示されるサブサンプリング方法を使用して参照サンプルの第2のセットをサブサンプリングし参照サンプルの第2のサブサンプリングされたセットを生成することのうちの、少なくとも1つを実行し得る。

【0540】

別の例では、ビデオデコーダ30は、現在のPUのサイズに基づいて、サブサンプリング方法を決定し得る。この例では、ビデオデコーダ30は、決定されたサブサンプリング方法を使用して参照サンプルの第1のセットをサブサンプリングし参照サンプルの第1のサブサンプリングされたセットを生成することと、決定されたサブサンプリング方法を使用して参照サンプルの第2のセットをサブサンプリングし参照サンプルの第2のサブサンプリングされたセットを生成することのうちの、少なくとも1つを実行し得る。

【0541】

さらに、図33の活動(560)において、ビデオデコーダ30は、参照サンプルの第1のサブサンプリングされたセットの中の参照サンプルの総数および参照サンプルの第2のサブサンプリングされたセットの中の参照サンプルの総数、参照サンプルの第1のサブサンプリングされたセット、ならびに参照サンプルの第2のサブサンプリングされたセットに基づいて、第1のICパラメータを決定し得る。たとえば、ビデオデコーダ30は、第1のICパラメータが

【0542】

【数81】

$$\frac{\sum_{i=0}^{2N-1} \text{Rec}_{\text{neigh}}(i) - a \cdot \sum_{i=0}^{2N-1} \text{Rec}_{\text{refneigh}}(i)}{2N}$$

【0543】

に基づくように第1のICパラメータを決定し得る。上の式において、2Nは参照サンプルの第1のサブサンプリングされたセットの中の参照サンプルの総数および参照サンプルの第2のサブサンプリングされたセットの中の参照サンプルの総数を表し、 $\text{Rec}_{\text{neigh}}(i)$ は参照サ

10

20

30

40

50

ンプルの第1のサブサンプリングされたセットの中の*i*番目の参照サンプルを表し、 $Rec_{ref_neigh}(i)$ は参照サンプルの第2のサブサンプリングされたセットの中の*i*番目の参照サンプルを表す。

【0544】

いくつかの例では、ビデオデコーダ30は、第2のICパラメータが

【0545】

【数82】

$$\frac{2N \cdot \sum_{i=0}^{2N-1} Rec_{neigh}(i) \cdot Rec_{refneigh}(i) - \sum_{i=0}^{2N-1} Rec_{neigh}(i) \cdot \sum_{i=0}^{2N-1} Rec_{refneigh}(i)}{2N \cdot \sum_{i=0}^{2N-1} Rec_{refneigh}(i) \cdot Rec_{neigh}(i) - \left(\sum_{i=0}^{2N-1} Rec_{refneigh}(i) \right)^2}$$

10

【0546】

に基づくように第2のICパラメータを決定し得る。

【0547】

加えて、図33の活動(562)において、ビデオデコーダ30は、第1のICパラメータに基づいて非正方形予測ブロックのサンプルを決定し得る。たとえば、サンプルは現在のピクチャの左上の角に対する相対的な座標(*i*, *j*)にあることがあり、ビデオデコーダ30はサンプルが以下に基づくようにサンプルを決定し得る。

20

$$a * r(i + dv_x, j + dv_y + b)$$

上の式において、*b*は第1のICパラメータであり、*a*は第2のICパラメータであり、*r*は参照ピクチャであり、*dv_x*は現在のPUのベクトルの水平成分であり、*dv_y*は現在のPUのベクトルの垂直成分である。

【0548】

ビデオデコーダ30は、非正方形予測ブロックに基づいて、現在のCUのコーディングブロックを再構築することができる(564)。たとえば、ビデオデコーダ30は、非正方形予測ブロックのサンプルを現在のCUのTUの残差ブロックの対応するサンプルに加算することによって、コーディングブロックのサンプルを再構築することができる。

【0549】

30

図34は、柔軟性のある残差木を使用する本開示の技法による、ビデオデータを符号化するためのビデオエンコーダ20の例示的な動作を示すフローチャートである。図34の例では、ビデオエンコーダ20がビデオデータを受信し得る(600)。さらに、ビデオエンコーダ20は、木構造に基づいてCUのTUへとビデオデータのCUを区分することができる(602)。いくつかの例では、ビデオエンコーダ20は、木構造の各々のそれぞれのノードに対して、それぞれのノードのための分割インジケータの値を決定し得る。それぞれのノードのための分割インジケータは、それぞれのノードがどれだけの子ノードを有するかを示し得る。いくつかの事例では、ビデオエンコーダ20は、ビットストリームにおいて、それぞれのノードの分割インジケータの値を明示的に示すシンタックス要素をシグナリングし得る。他の事例では、ビデオデコーダ30は、それぞれのノードのための分割インジケータの値を(たとえば、木構造におけるノードの深度、親ノードの分割ノードの値、それぞれのノードに対応する予測ブロックのサイズおよび/または形状などに基づいて)推測し得る。

40

【0550】

木構造に基づいてCUをCUのTUへと区分することの一部として、ビデオエンコーダ20は、木構造の中のノードが木構造の中で丁度2つの子ノードを有すると決定することができる(604)。この例では、木構造のルートノードはCUのコーディングブロックに対応する。木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応する。木構造のリーフノードはCUのTUに対応する。

【0551】

50

たとえば、ビデオエンコーダ20は、CUのPUの総数に基づいて、木構造が二分木か四分木かを決定し得る。この例では、CUが2つのPUを有することに基づいて、ノードは木構造の中で丁度2つの子ノードを有する。言い換えると、ビデオエンコーダ20は、CUが丁度2つのPUを有することに基づいて、ノードが木構造の中で丁度2つの子ノードを有すると決定し得る。

【0552】

いくつかの例では、ビデオエンコーダ20は、CUが丁度2つのPUを有することに基づいて、ノードが木構造の中で丁度2つの子ノードを有すると決定し得る。

【0553】

さらに、図34の例では、CUのTUのうちの少なくとも1つに対して、ビデオエンコーダ20は、変換をTUの残差ブロックに適用して、TUの変換係数のブロックを生成する(606)。たとえば、ビデオエンコーダ20は、離散コサイン変換(DCT)、離散サイン変換(DST)、または別のタイプの変換をTUの残差ブロックに適用して、変換係数のブロックを生成し得る。加えて、ビデオエンコーダ20は、TUの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することができる(608)。たとえば、ビデオエンコーダ20は、変換係数が0ではないかどうか、変換係数が1より大きいかどうか、変換係数が2より大きいかどうか、変換係数の符号、および変換係数の剰余を示すシンタックス要素を生成し得る。この例では、ビデオエンコーダ20は、これらのシンタックス要素のうちの1つまたは複数にCABACコーディングを適用し得る。

【0554】

図35は、柔軟性のある残差木を使用する本開示の技法による、ビデオデータを復号するためのビデオデコーダ30の例示的な動作を示すフローチャートである。図35の例では、ビデオデコーダ30は、ビデオデータの符号化された表現を備えるビットストリーム受信することができる(650)。加えて、ビデオデコーダ30は、木構造に基づいてCUのTUへとビデオデータのCUが区分されると決定することができる(652)。いくつかの例では、ビデオデコーダ30は、木構造の各々のそれぞれのノードに対して、それぞれのノードのための分割インジケータの値を決定し得る。それぞれのノードのための分割インジケータは、それぞれのノードがどれだけの子ノードを有するかを示し得る。いくつかの事例では、ビデオデコーダ30は、ビットストリームから、それぞれのノードの分割インジケータの値を明示的に示すシンタックス要素を取得し得る。他の事例では、ビデオデコーダ30は、それぞれのノードのための分割インジケータの値を(たとえば、木構造におけるノードの深度、親ノードの分割ノードの値、それぞれのノードに対応する予測ブロックのサイズおよび/または形状などに基づいて)推測し得る。

【0555】

木構造に基づいてCUがCUのTUへと区分されると決定することの一部として、ビデオデコーダ30は、木構造の中のノードが木構造の中で丁度2つの子ノードを有すると決定することができる(654)。この例では、木構造のルートノードはCUのコーディングブロックに対応する。木構造の各々のそれぞれの非ルートノードは、それぞれの非ルートノードの親ノードに対応するブロックの区分であるそれぞれのブロックに対応する。木構造のリーフノードはCUのTUに対応する。本開示の他の箇所では説明されるように、ビデオデコーダ30は、CUの中のPUの数に基づいて、木構造におけるノードの深度に基づいて、シグナリングされるシンタックス要素に基づいて、または他のデータに基づいて、木構造の中のノードが丁度2つの子ノードを有すると決定し得る。

【0556】

たとえば、ビデオデコーダ30は、CUのPUの総数に基づいて、木構造が二分木か四分木かを決定し得る。この例では、CUが2つのPUを有することに基づいて、ノードは木構造の中で丁度2つの子ノードを有する。言い換えると、ビデオデコーダ30は、CUが丁度2つのPUを有することに基づいて、ノードが木構造の中で丁度2つの子ノードを有すると決定し得る。

【0557】

10

20

30

40

50

CUのTUのうちの少なくとも1つに対して、ビデオデコーダ30は、変換をTUの係数ブロックに適用して、TUの残差ブロックを生成することができる(656)。たとえば、ビデオデコーダ30は、逆DCT、逆DST、または別のタイプの変換をTUの係数ブロックに適用して、TUの残差ブロックを生成し得る。加えて、ビデオデコーダ30は、予測ブロックのサンプルをCUのTUの残差ブロックの対応するサンプルに加算することによって、コーディングブロックのサンプルを再構築することができる(658)。

【0558】

本開示のいくつかの態様は、説明を目的にHEVC規格の拡張に関して説明されている。しかしながら、本開示において説明される技法は、まだ開発されていない他の標準的なまたは独自の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にとって有用であり得る。

10

【0559】

ビデオコーダは、本開示において説明されるように、ビデオエンコーダまたはビデオデコーダを指すことがある。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すことがある。同様に、ビデオコーディングは、場合により、ビデオ符号化またはビデオ復号を指すことがある。

【0560】

例によっては、本明細書において説明された技法のうち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てよく、追加され、統合され、または完全に除外されてよい(たとえば、説明されたすべての行為またはイベントが技法の実践にとって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じて並行して実行されてよい。

20

【0561】

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、コンピュータ可読媒体は、一般に、(1)非一時的な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対応し得る。データ記憶媒体は、本開示において説明される技法の実装のための命令、コード、および/またはデータ構造を取り出すために、1つまたは複数のコンピュータまたは1つまたは複数のプロセッサによってアクセスされ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

【0562】

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ストレージデバイス、フラッシュメモリ、または、命令もしくはデータ構造の形態の所望のプログラムコードを記憶するために使用され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、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(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、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、または他の一時的媒体を含まないが、代わりに非一時的有形記憶媒体を対象とすることを理解されたい。本明細書において使用され

40

50

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。上の組合せもコンピュータ可読媒体の範囲に含まれるべきである。

【0563】

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ロジックアレイ(FPGA)、または他の等価な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、本明細書において使用される「プロセッサ」という用語は、上記の構造、または本明細書において説明される技法の実装に適した任意の他の構造のいずれかを指すことがある。加えて、いくつかの態様では、本明細書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ることがあり、または複合コーデックに組み込まれることがある。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装される可能性がある。

【0564】

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において実装され得る。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構成要素、モジュール、またはユニットが説明されたが、それらの構成要素、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つまたは複数のプロセッサを含む相互動作可能なハードウェアユニットの集合によって提供されることがある。

【0565】

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。

【符号の説明】

【0566】

- 10 ビデオ符号化および復号システム
- 12 ソースデバイス
- 14 宛先デバイス
- 16 コンピュータ可読媒体
- 18 ビデオソース
- 19 記憶媒体
- 20 ビデオエンコーダ
- 22 出力インターフェース
- 28 入力インターフェース
- 29 記憶媒体
- 30 ビデオデコーダ
- 32 ディスプレイデバイス
- 50 ブロック
- 52 サンプル
- 54 サンプル
- 60 再構築されたサンプル
- 62 再構築されたサンプル
- 70 サンプル
- 72 予測ブロック
- 74 イントラ予測方向

10

20

30

40

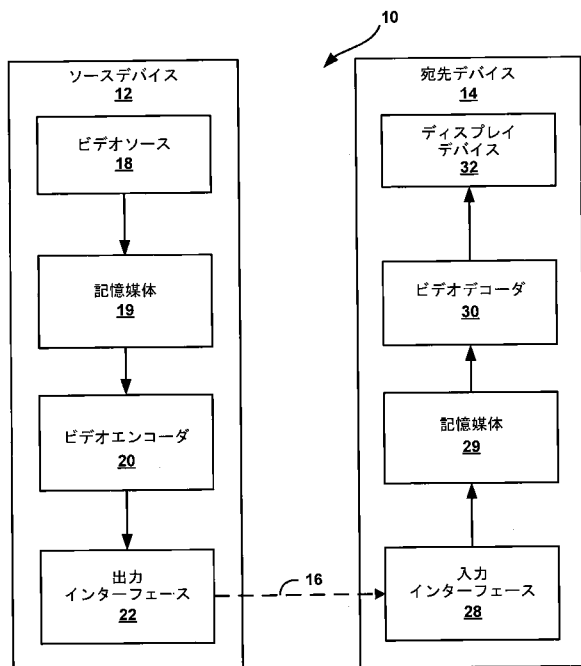
50

78	領域	
80	係数行列	
82	8 × 8行列	
84	1Dバッファ	
90	色差予測ブロック	
92	輝度予測ブロック	
100	コーディングブロック	
110	現在のCU	
112	参照ブロック	
114	参照ブロック	10
116	現在のCU	
130	変換ブロック	
132	変換ブロック	
134	変換ブロック	
136	変換ブロック	
137	変換ブロック	
138	変換ブロック	
139	変換ブロック	
140	変換ブロック	
141	変換ブロック	20
142	変換ブロック	
143	変換ブロック	
150	変換ブロック	
152	変換ブロック	
154	変換ブロック	
156	変換ブロック	
158	変換ブロック	
160	変換ブロック	
162	変換ブロック	
164	変換ブロック	30
166	変換ブロック	
180	TU	
182	TU	
184	TU	
186	TU	
188	TU	
190	TU	
192	TU	
194	TU	
196	TU	40
198	TU	
200	予測処理ユニット	
201	ビデオデータメモリ	
202	残差生成ユニット	
204	変換処理ユニット	
206	量子化ユニット	
208	逆量子化ユニット	
210	逆変換処理ユニット	
212	再構築ユニット	
214	フィルタユニット	50

- 216 復号ピクチャバッファ
- 218 エントロピー符号化ユニット
- 220 インター予測処理ユニット
- 222 LMベースの符号化ユニット
- 226 イントラ予測処理ユニット
- 250 エントロピー復号ユニット
- 251 ビデオデータメモリ
- 252 予測処理ユニット
- 254 逆量子化ユニット
- 256 逆変換処理ユニット
- 258 再構築ユニット
- 260 フィルタユニット
- 262 復号ピクチャバッファ
- 264 動き補償ユニット
- 265 LMベースの復号ユニット
- 266 イントラ予測処理ユニット

10

【図 1】



【図 2 A】

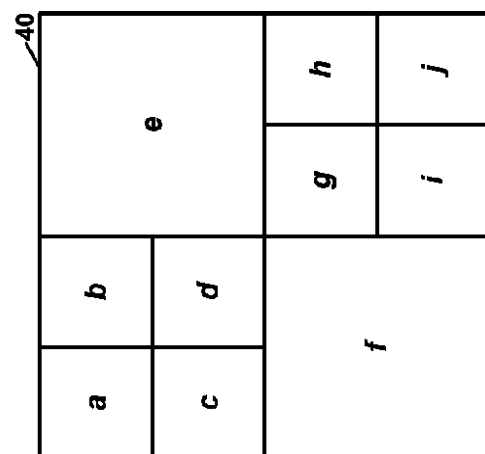


FIG. 2A

【 図 2 B 】

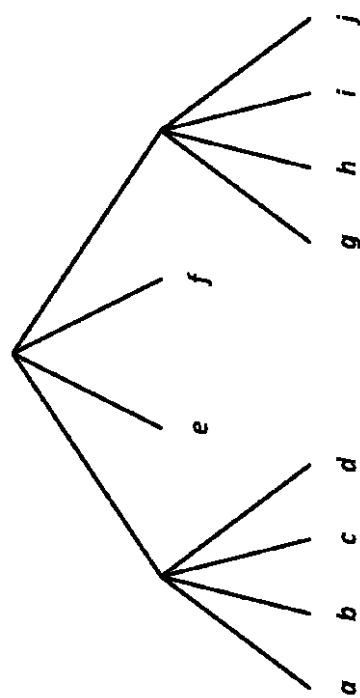


FIG. 2B

【 図 3 】

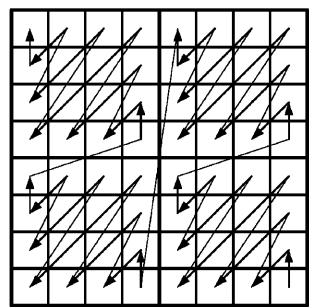


FIG. 3

【 図 4 】

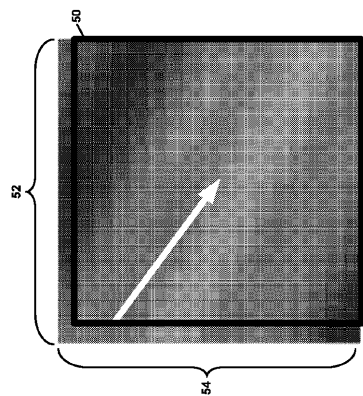
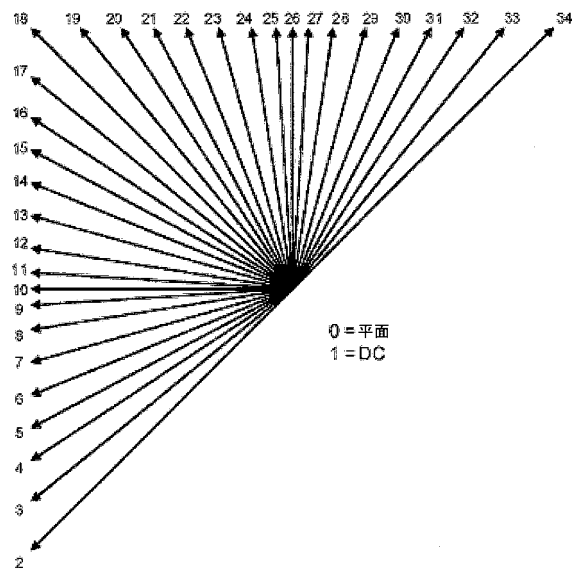


FIG. 4

【 図 5 】



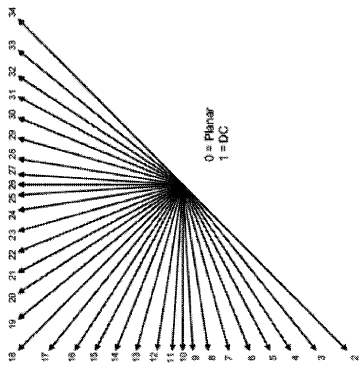


FIG. 5

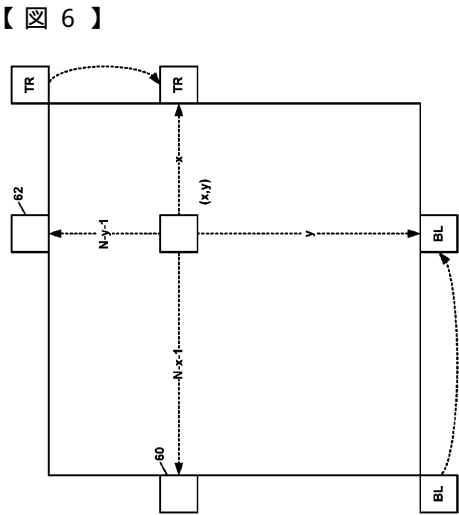


FIG. 6

【 図 7 】

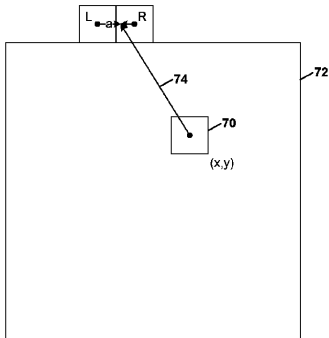


FIG. 7

【 図 8 】

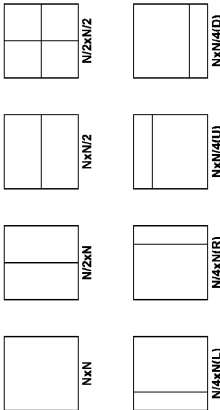
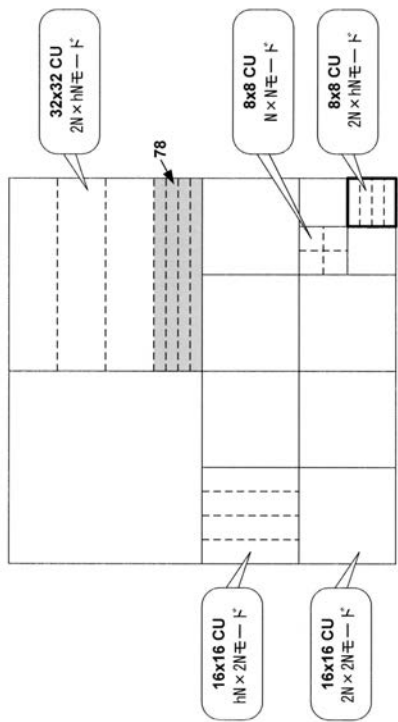


FIG. 8

【 図 9 】



【 図 1 0 】

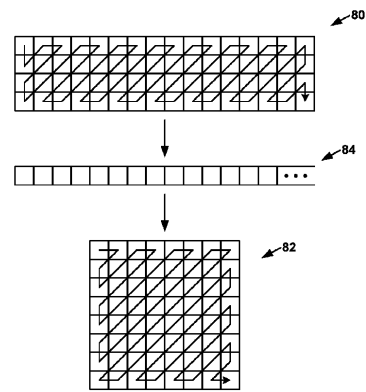


FIG. 10

【 図 1 1 】

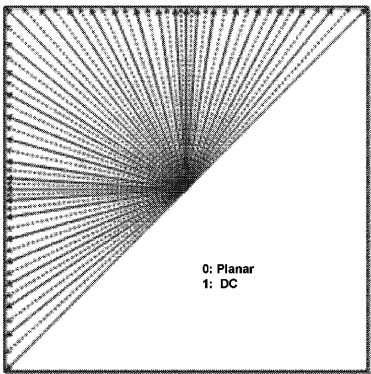
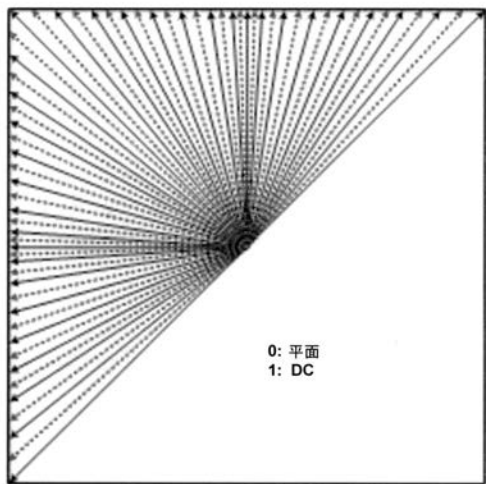
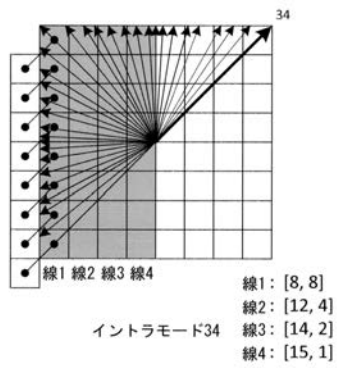
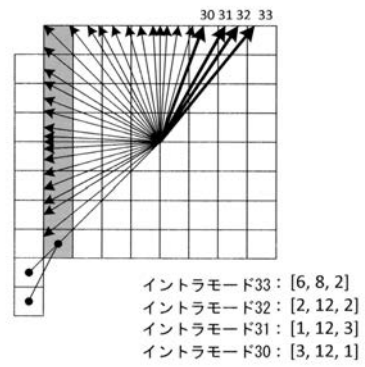


FIG. 11

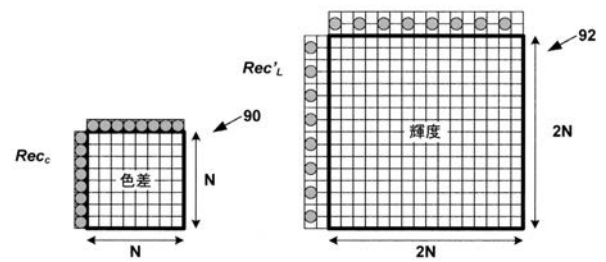
【図 1 2 A】



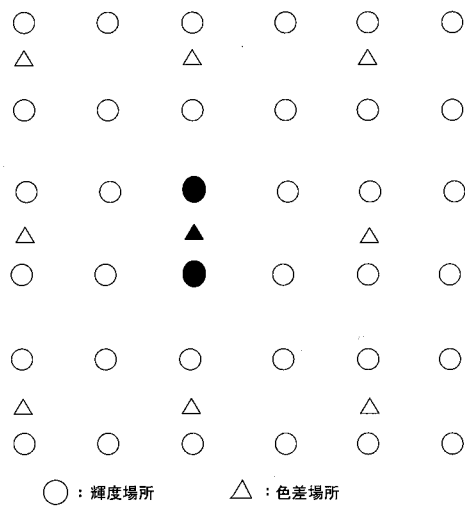
【図 1 2 B】



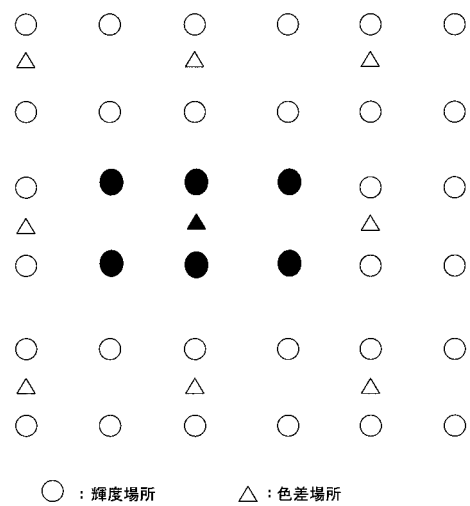
【図 1 3】



【図 1 4】



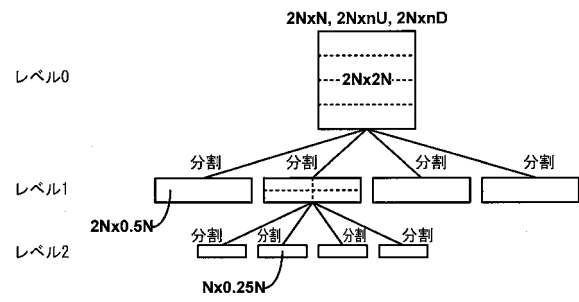
【図 1 5】



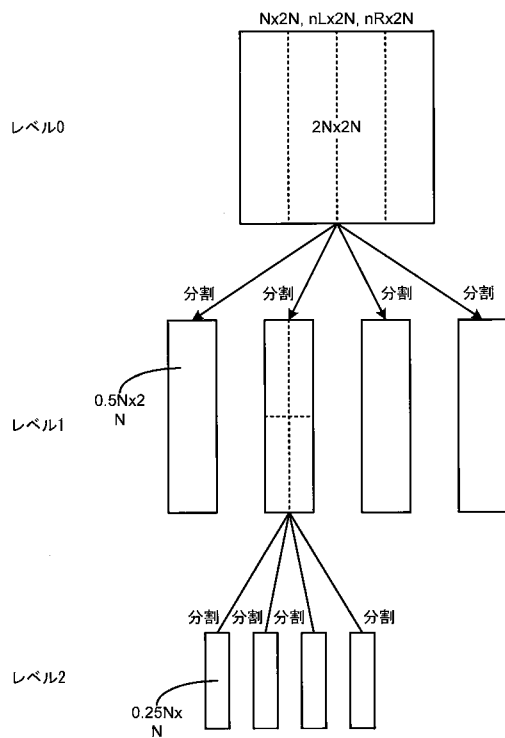
【図 16】



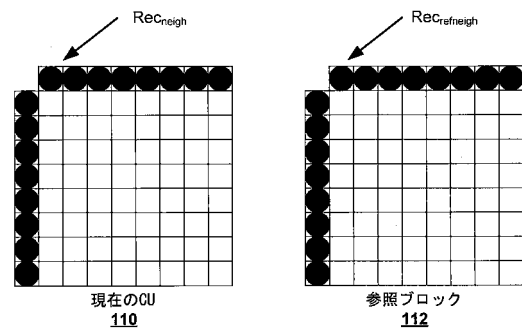
【図 17】



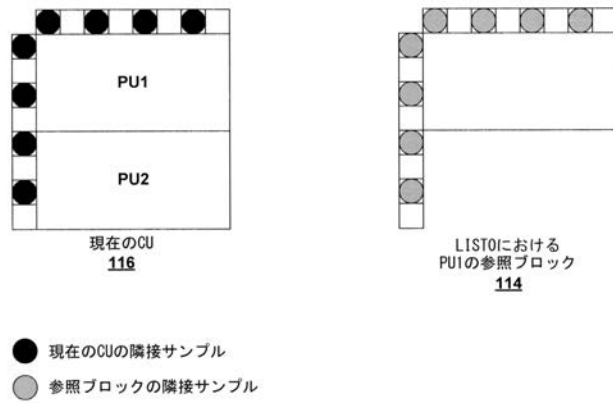
【図 18】



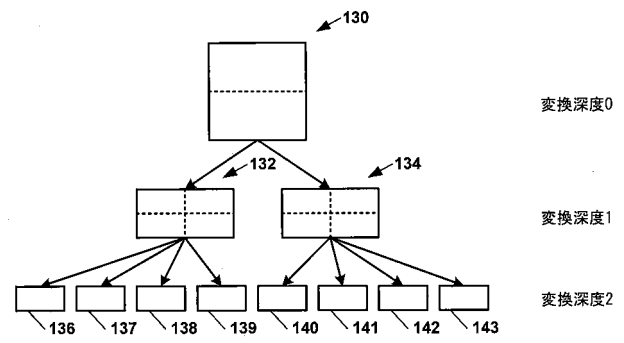
【図 19】



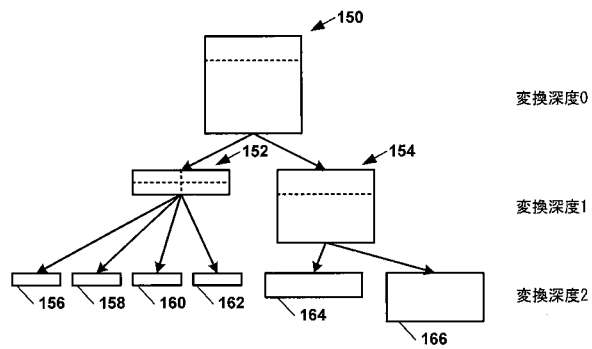
【図 20】



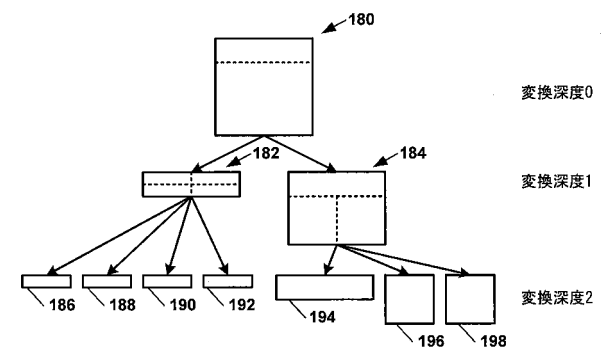
【図 21】



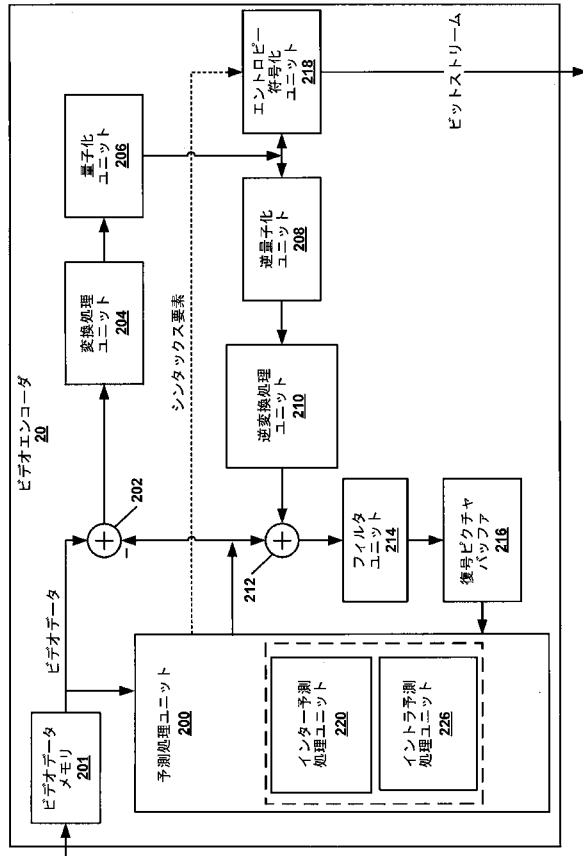
【図 22】



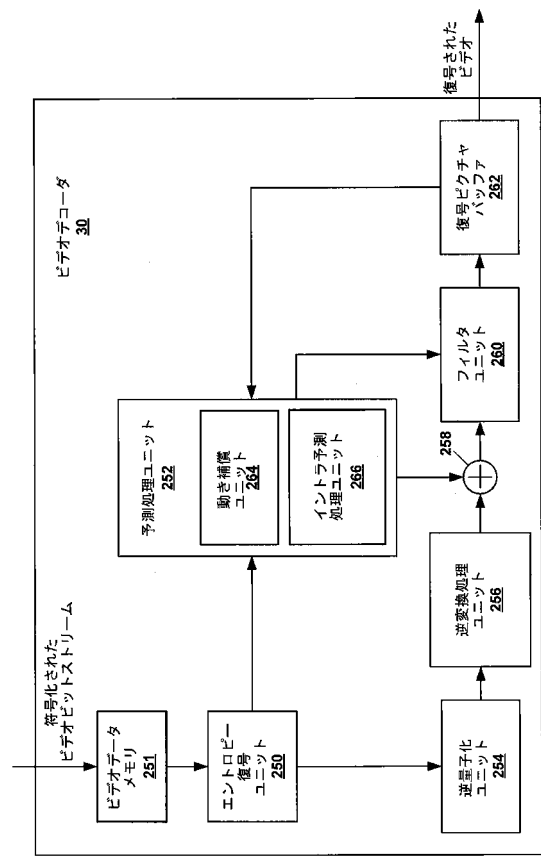
【図 23】



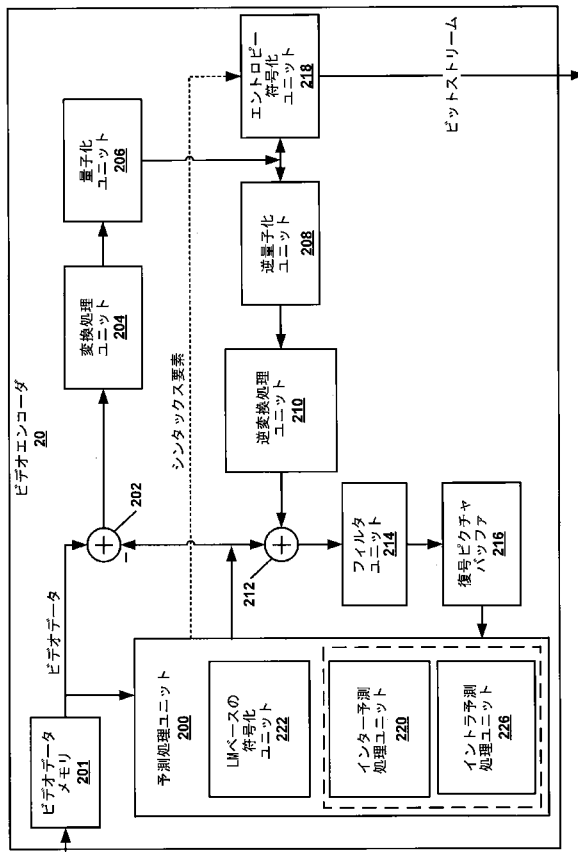
【図 24】



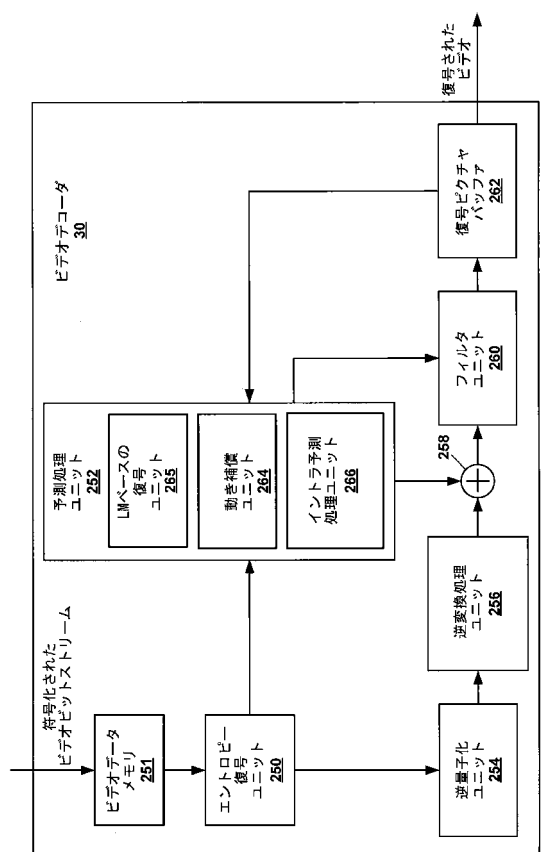
【図 25】



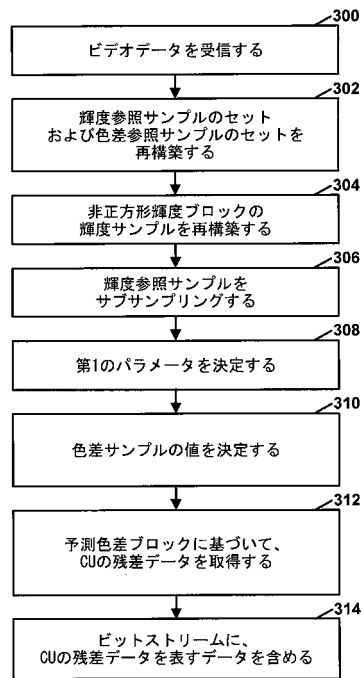
【図 26】



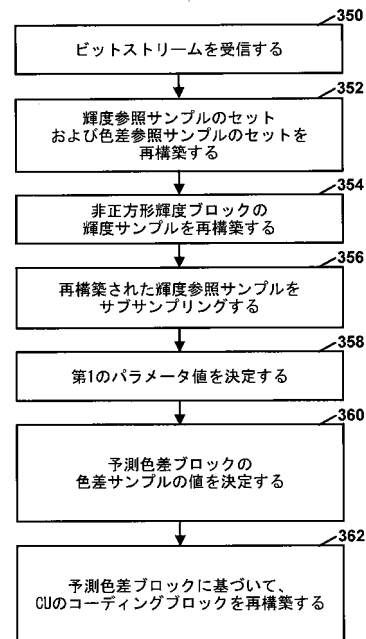
【図 27】



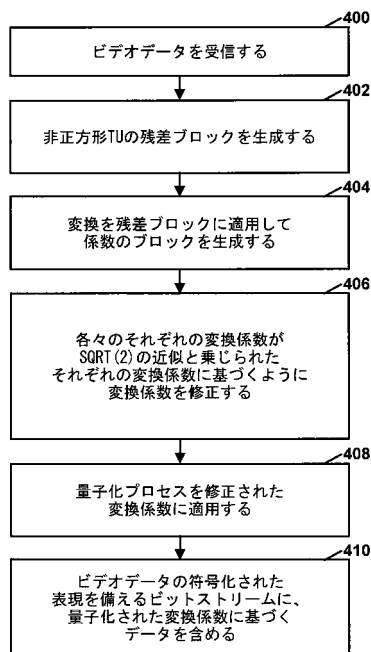
【図 28】



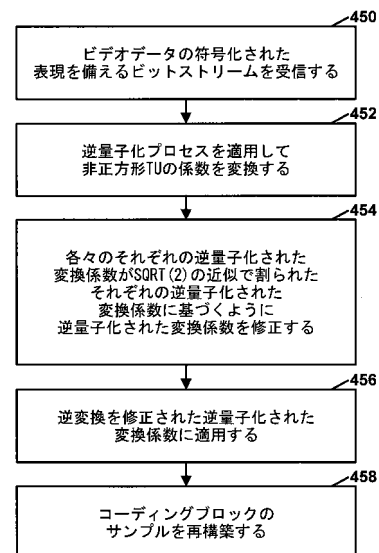
【図 29】



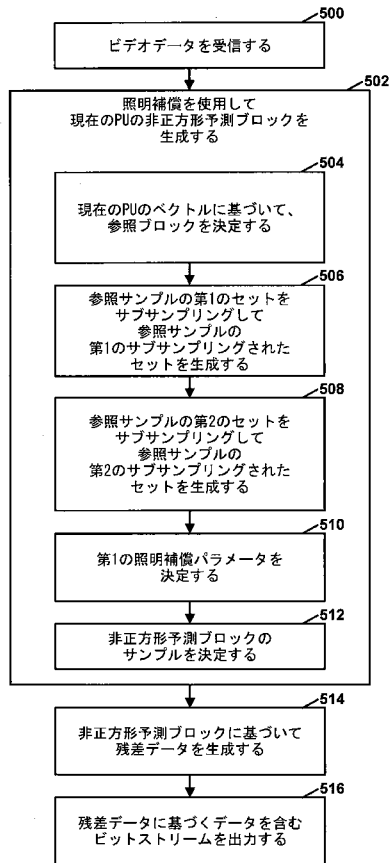
【図 30】



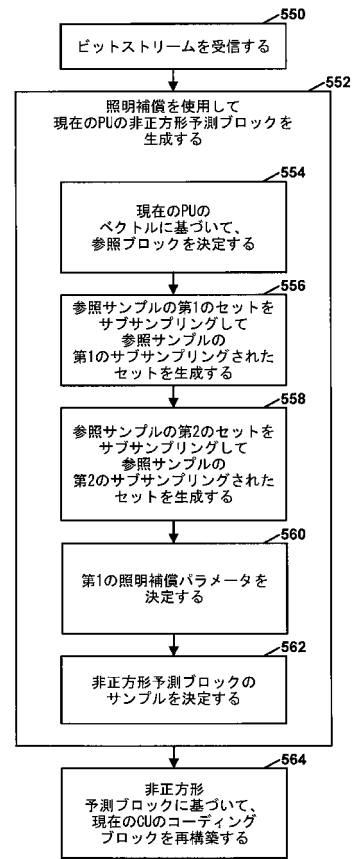
【図 31】



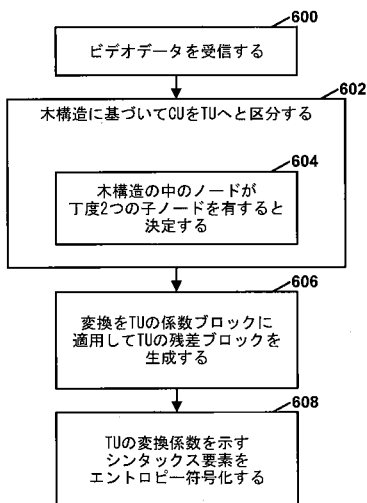
【図 3 2】



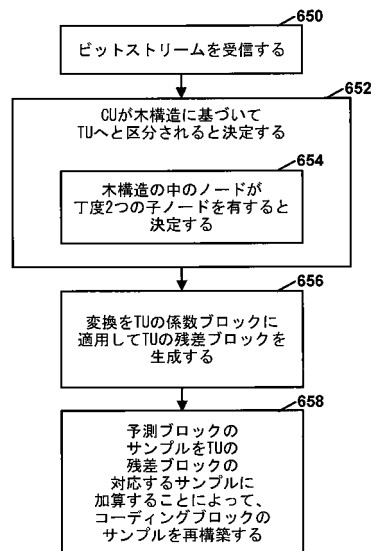
【図 3 3】



【図 3 4】



【図 3 5】



【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2016/063672

A. CLASSIFICATION OF SUBJECT MATTER

INV. H04N19/119 H04N19/625 H04N19/42 G06F17/14 H04N19/60
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04N G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPO-Internal, COMPENDEX, INSPEC, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	MATHIAS WIEN: "Adaptive Block Transforms (ABT)", 13. VCEG MEETING; 02-04-2001 - 04-04-2001; AUSTIN, TEXAS, US; (VIDEOCODING EXPERTS GROUP OF ITU-T SG.16),, no. VCEG-M62, 28 March 2001 (2001-03-28), XP030003232, ISSN: 0000-0461 section "Introduction" -----	1-24
A	US 2013/136175 A1 (WANG XIANGLIN [US] ET AL) 30 May 2013 (2013-05-30) figures 7-10 paragraph [0121] - paragraph [0161] ----- -/--	1-24

☒ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"Z" document member of the same patent family

Date of the actual completion of the international search

7 March 2017

Date of mailing of the international search report

14/03/2017

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel: (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Stoufs, Maryse

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2016/063672

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CAO (TSINGHUA) X ET AL: "CE6.b1 Report on Short Distance Intra Prediction Method", 20110311, no. JCTVC-E278, 11 March 2011 (2011-03-11), XP030008784, ISSN: 0000-0007 cited in the application section 2.2 "SDIP intra prediction and coding" -----	1-24
A	YUAN (TSINGHUA) Y ET AL: "CE2: Non-Square Quadtree Transform for symmetric and asymmetric motion partitions", 97. MPEG MEETING; 18-7-2011 - 22-7-2011; TORINO; (MOTION PICTURE EXPERT GROUP OR ISO/IEC JTC1/SC29/WG11),, no. m20839, 12 July 2011 (2011-07-12), XP030049402, the whole document -----	1-24
A	REZNIK Y A ET AL: "Efficient fixed-point approximations of the 8x8 Inverse Discrete Cosine Transform", OPTOMECHATRONIC MICRO/NANO DEVICES AND COMPONENTS III : 8 - 10 OCTOBER 2007, LAUSANNE, SWITZERLAND; [PROCEEDINGS OF SPIE , ISSN 0277-786X], SPIE, BELLINGHAM, WASH, vol. 6696, 24 September 2007 (2007-09-24), pages 669617-1, XP002489240, DOI: 10.1117/12.740228 ISBN: 978-1-62841-730-2 the whole document -----	1-24
A	MATHIAS WIEN: "Variable Block-Size Transforms for Hybrid Video Coding", DISSERTATION DER RHEINISCH-WESTFAELISCHEN TECHNISCHEN HOCHSCHULE AACHEN, 3 February 2004 (2004-02-03), pages 1-183, XP002481661, [retrieved on 2004-02-03] section 6.3.2 "Normalization and Quantization" -----	1-24

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2016/063672

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2013136175 A1	30-05-2013	CN 103797801 A	14-05-2014
		EP 2756676 A2	23-07-2014
		JP 6022579 B2	09-11-2016
		JP 2014530537 A	17-11-2014
		JP 2016201808 A	01-12-2016
		KR 20140049098 A	24-04-2014
		US 2013136175 A1	30-05-2013
		WO 2013039908 A2	21-03-2013

フロントページの続き

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

(72)発明者 シアン・リ

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 ジエンレ・チェン

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 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

(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

Fターム(参考) 5C159 LC03 MA04 MA05 MA25 MC06 MC11 PP04 UA02 UA05

【要約の続き】

$$\sqrt{2}$$

という係数または

$$\sqrt{2}$$

の近似で割られたそれぞれの逆量子化された変換係数に基づくように逆量子化された変換係数を修正する。