COLOR TRANSFORM FOR VIDEO CODING

[0001] This present disclosure claims the benefit of priority to U.S. Patent Application No. 17/035,218, “Color Transform for Video Coding” filed on September 28, 2020, which claims the benefit of priority to U.S. Provisional Application No. 62/913,486, “Improved Color Transform In VVC” filed on October 10, 2019.

TECHNICAL FIELD

[0002] The present disclosure describes embodiments generally related to video coding.

BACKGROUND

[0003] The background description provided herein is for the purpose of generally presenting the context of the disclosure. Work of the presently named inventors, to the extent the work is described in this background section, as well as aspects of the description that may not otherwise qualify as prior art at the time of filing, are neither expressly nor impliedly admitted as prior art against the present disclosure.

[0004] Video coding and decoding can be performed using inter-picture prediction with motion compensation. Uncompressed digital video can include a series of pictures, each picture having a spatial dimension of, for example, 1920 x 1080 luminance samples and associated chrominance samples. The series of pictures can have a fixed or variable picture rate (informally also known as frame rate), of, for example 60 pictures per second or 60 Hz. Uncompressed video has significant bitrate requirements. For example, 1080p60 4:2:0 video at 8 bit per sample (1920x1080 luminance sample resolution at 60 Hz frame rate) requires close to 1.5 Gbit/s bandwidth. An hour of such video requires more than 600 GBytes of storage space.

[0005] One purpose of video coding and decoding can be the reduction of redundancy in the input video signal, through compression. Compression can help reduce the aforementioned bandwidth or storage space requirements, in some cases by two orders of magnitude or more. Both lossless and lossy compression, as well as a combination thereof can be employed. Lossless compression refers to techniques where an exact copy of the original signal can be reconstructed from the compressed original signal. When using lossy compression, the

Inputs to each option of color space conversion equations include an array of luma residual samples of the luma residual block with elements rY[x][y], an array of chroma residual samples of the first chroma residual block with elements rCb[x][y], and an array of chroma residual samples of the second chroma residual block with elements rCr[x][y]. Outputs from each option of color space conversion equations include a modified array of luma residual samples of the luma residual block with elements rY[x][y], a modified array of chroma residual samples of the first chroma residual block with elements rCb[x][y], and a modified array of chroma residual samples of the second chroma residual block with elements rCr[x][y].

[0026] In an embodiment, the first option of color space conversion equations is selected when a transform skip flag corresponding to each of the residual blocks of the current CU has a value of 1. In an embodiment, the second option of color space conversion equations is selected when at least one of transform skip flags corresponding to the residual blocks of the current CU has a value of 0.

[0027] In an embodiment, the first option of color space conversion equations is selected when a transform skip flag corresponding to each of the residual blocks of the current CU has a value of 1 and a quantization parameter (QP) corresponding to each of the residual blocks of the current CU has a value of 4. In an embodiment, the second option of color space conversion equations is selected when at least one of QPs corresponding to the residual blocks of the current CU has a value not equal to 4.

[0028] In an embodiment, a syntax element that is received indicates only the first option of color space conversion equations is applied. The syntax element is one of a slice level syntax element, a picture level syntax element, or a sequence level syntax element. In an embodiment, a syntax element that is received indicates only the first option of color space conversion equations is applied regardless of a value of a flag indicating whether transform and quantization are bypassed for residuals of a CU. The syntax element is one of a slice level syntax element, a picture level syntax element, or a sequence level syntax element.

[0029] In an embodiment, one of two options of color space conversion equations is selected according to a value of a transform skip flag corresponding to each of the residual blocks of the current CU. In an embodiment, one of two options of color space conversion equations is selected according to a variable indicating whether a block-based differential pulse code modulation (BDPCM) is applied to the current CU.

component, i.e., DM mode. In an embodiment, to compensate a dynamic range change of residuals signals before and after color transform, QP adjustments, for example, of (-5, -5, -3), are applied to the transform residuals.

[0151] As the forward and inverse color transforms need to access the residuals of all three components, in some embodiments, the ACT is disabled for some scenarios where not all residuals of three components are available. For example, when separate-tree partition is employed, luma and chroma samples inside one CTU are partitioned by different tree structures. This results in that the CUs in the luma-tree only contains luma component and the CUs in the chroma-tree only contains two chroma components. When intra sub-partition prediction (ISP) is employed, the ISP sub-partition is applied to luma while chroma signals are coded without splitting. Except the last ISP sub-partitions, the other sub-partitions only contain luma component. Accordingly, the ACT can be disabled for separate-tree (dual tree) partition and ISP.

[0152] VI. Cross-Component Linear Model (CCLM)

[0153] For the chroma component of an intra PU, the encoder may select the best chroma prediction modes among 8 modes including Planar, DC, Horizontal, Vertical, a direct copy of the intra prediction mode (DM) from the luma component, Left and Top Cross-component Linear Mode (LT_CCLM), Left Cross-component Linear Mode (L_CCLM), and Top Cross-component Linear Mode (T_CCLM). LT_CCLM, L_CCLM, and T_CCLM can be categorized into the group of Cross-component Linear Mode (CCLM). The difference between these 3 modes is that different regions of neighboring samples may be used for deriving the parameters α and β. For LT_CCLM, both the left and top neighboring samples may be used to derive the parameters α and β. For L_CCLM, in some examples, only the left neighboring samples are used to derive the parameters α and β. For T_CCLM, in some examples, only the top neighboring samples are used to derive the parameters α and β.

[0154] Cross-Component Linear Model (CCLM) prediction modes may be used to reduce a cross-component redundancy, in which the chroma samples are predicted based on the reconstructed luma samples of the same CU by using an example linear model as follows:

Eq. (9): $pred_C(i,j) = \alpha \cdot rec_L'(i,j) + \beta,$

where $pred_C(i,j)$ represents the predicted chroma samples in a CU and $rec_L'(i,j)$ represents the downsampled reconstructed luma samples of the same CU. Parameters α and β may be derived by a straight line equation, which may also be referred to as a max-min method. Since this

computation process may be performed as part of the decoding process, not just as an encoder search operation, no syntax may be used to convey the α and β values.

[0155] For chroma 4:2:0 format, CCLM prediction may apply a six-tap interpolation filter to get the down-sampled luma sample corresponding to a chroma sample as shown in FIGs. 10B-10D. In a formula way, a down-sampled luma sample Rec'L[x, y] is calculated from reconstructed luma samples. The down-sampled luma samples may be used to find the maximum and minimum sample points. The 2 points (a pair of Luma and Chroma) (A, B) may be the minimum and maximum values inside the set of neighboring Luma samples as depicted in FIG 13.

[0156] In FIG. 13, the linear model parameters α and β may be obtained according to the following equations:

Eq. (10): $\alpha = \frac{y_B - y_A}{x_B - x_A}$

Eq. (11): $\beta = y_A - \alpha x_A$

[0157] A division operation is advantageously avoided by using a multiplication and a shift operation in an example. One Look-up Table (LUT) may be used to store the pre-calculated values, and the absolute difference values between maximum and minimum luma samples may be used to specify the entry index of the LUT. The size of the LUT may be 512 in an example.

[0158] FIGs. 14A and 14B illustrate example locations of the samples used for the derivation of α and β in LT_CCLM. In T_CCLM mode, in some examples, only the above neighboring samples (including 2 * W samples) are used to calculate the linear model coefficients. FIGs. 15A and 15B illustrate example locations of the samples used for the derivation of α and β in T_CCLM.

[0159] In L_CCLM mode, in some examples, only left neighboring samples (including 2 * H samples) are used to calculate the linear model coefficients. FIGs. 16A and 16B illustrate example locations of the samples used for the derivation of α and β in L_CCLM.

[0160] The CCLM prediction mode may also include prediction between the two chroma components (i.e., the Cr component is predicted from the Cb component). Instead of using the reconstructed sample signal, the CCLM Cb-to-Cr prediction may be applied in the residual domain. The CCLM Cb-to-Cr prediction may be implemented by adding a weighted reconstructed Cb residual to the original Cr intra prediction to form the final Cr prediction:

has a bit depth of 10 bits while the chroma component has a bit depth of 8 bits. The color transform by applying the various color transform equations (or color space conversion equations) disclosed herein may be ineffective for reducing statistic dependency among the luma and chroma components. In such a scenario, a decoder can determine that no color transform is used for a current CU when a syntax element(s) indicating different bit depths for different color components is received.

[0177] In an embodiment, when transform skip flags signaled in TU level (or CU level) for each color component index for a current CU are all equal to 1 or 0, a QP for each component of the current CU is equal to 4, and the bit depths of the luma and chroma components are different, color transform (in spatial residual domain or before prediction) is not used for the CU. In an example, the transform skip flags for the luma or the chroma component specifies whether a transform (e.g., DCT transform) is applied to an associated transform block or not. The transform skip flag equal to 1 specifies that no transform is applied to the associated transform block. The transform skip flag equal to 0 specifies that the decision whether transform is applied to the associated transform block or not depends on other syntax elements.

[0178] In an example, when the transform skip flag is not present, it is inferred as follows. If a block-based differential pulse code modulation (BDPCM) flag of the respective color component is equal to 1, the transform skip flag is inferred to be equal to 1. Otherwise (the BDPCM flag is equal to 0), the transform skip flag is inferred to be equal to 0. For example, the BDPCM flag equal to 1 specifies that BDPCM is applied to the current chroma or luma coding blocks i.e. the transform is skipped. The BDPCM equal to 0 specifies that BDPCM is not applied to the current chroma or luma coding blocks. In an example, when the BDPCM flag is not present, the BDPCM flag is inferred to be equal to 0.

[0179] In an example, when the QP for each component of the CU is equal to 4, a corresponding quantization step for quantizing transform coefficients can have a value of 1. As a result, there are no changes to values of the original transform coefficients after the quantization operation. Accordingly, the QP of a value of 4 has the effect of a lossless mode.

[0180] In an embodiment, when transform skip flags signaled in TU level (or CU level) for each color component index for a current CU are all equal to 1 or 0 and a QP for each component of the current CU is equal to 4, color transform (in spatial residual domain or before prediction) is not used for the current CU.

[0185] According to the present disclosure, the color transform corresponding to the first option, Option 1, can be suitable for lossless processing of CUs, while the color transform corresponding to the second option, Option 2, can be suitable for lossy processing of CUs. For example, for CUs with a lossless mode enabled (e.g., transform and quantization skip flag equal to 1 is signaled or interpreted), Option 1 color transform can have a better decorrelation effect than Option 2 color transform.

[0186] In an embodiment, for a current CU, when color transform is applied and signaled on CU level, transform skip flags signaled in CU level (or TU level) for each component index are all equal to 1, and a QP of each component of the current CU is equal to 4, it can be determined that the Option 1 color transform can be applied for the CU. For example, a CU level color transform enabled flag can be received from a bitstream at a decoder. The color transform enabled flag equal to 1 specifies that the decoded residuals of the current coding unit are applied using a color space conversion. The color transform enabled flag equal to 0 specifies that the decoded residuals of the current coding unit are processed without a color space conversion. In an embodiment, when the color transform enabled flag is not present, the color transform enabled flag is inferred to be equal to 0. One or more syntax elements indicating the QP of value 4 can also be received from the bitstream. Based on values of the CU level color transform enabled flag, the transform skip flags, and the QP value, the decoder can determine that Option 1 color transform is applied to the current CU.

[0187] In an embodiment, for a current CU, when color transform is applied and signaled on CU level, and at least one of transform skip flags signaled in CU level (TU level) for each component index is not equal to 1, it can be determined that the Option 2 color transform is applied to the current CU. In an embodiment, for a current CU, when color transform is applied and signaled on CU level, and a QP of one of the components of the current CU is not equal to 4, it can be determined that the Option 2 color transform is applied to the current CU.

[0188] In an embodiment, for a current CU, when color transform is applied and signaled in CU level and transform skip flags signaled in CU level for each component index are all equal to 1, it can be determined that the Option 1 color transform is applied to the current CU.

[0189] In an embodiment, only Option 1 color transform is applied for a CTU, a slice, a video sequence, or the like. Option 2 color transform is not used. In an embodiment, only Option 1 color transform is applied for a CTU, a slice, a video sequence, or the like regardless of values of transform and quantization bypass flags. In an example, the transform and quantization

bypass equal 1 specifies transform and quantization is bypassed for a current CU. The current CU thus can be processed in a lossless mode. The transform and quantization bypass equal 0 specifies transform and quantization is not bypassed for the current CU, or whether the transform and quantization is bypassed for the current CU depends on other information.

[0190] In an embodiment, the selection of color transform (whether Option 1 or Option 2) depends on a transform skip flag(s) and/or a BDPCM flag(s) of a current CU. In an example, the selection of the color transform option can depend on transform skip flags of each color component of the current CU as described above.

[0191] In an example, the selection of a color transform option can depend on a BDPCM flag of the current CU. For example, when the BDPCM flag indicates a BDPCM mode is applied to a luma or chroma component of the current CU, it can be determined that transform processing is skipped for the respective luma or chroma component. Otherwise, when the BDPCM flag indicates a BDPCM mode is not applied to a luma or chroma component of the current CU, it can be determined that transform is applied for the respective luma or chroma component. The transform skip flags can accordingly be interpreted to be zero or one. Accordingly, the selection of the color transform option, as described above, can depend on the transform skip flags of each color component with the interpreted values.

[0192] In an example, a BDPCM mode is used for processing a current CU. In a prediction process, for a current pixel X having pixel A as left neighbor, pixel B as top neighbor and C as top-left neighbor, a prediction P(X) can be determined by

$$P(X) = \begin{cases} min(A,B)\ if\ C \geq max(A,B) \\ max(A,B)\ if\ C \leq min(A,B) \\ A + B - C\ otherwise \end{cases}$$

The prediction process uses unfiltered reference pixels when predicting the top row and left column of the current CU. The prediction process then uses reconstructed pixels for the rest of the current CU. Pixels are processed in raster-scan order inside the CU. The prediction errors can then be quantized in spatial domain without transform followed by entropy coding. In an example, a BDPCM flag can be transmitted in CU level to indicate whether a regular intra coding or a BDPCM is used.

[0193] **X. Examples of Decoding Process with Color Transform**

[0194] FIG. 19 illustrates an embodiment of a process (1900) performed by a decoder such as decoder (710). The process (1900) can start from (S1901), and proceed to (S1910).

[0195] At (S1910), a syntax element can be received from a bitstream of a coded video. For example, the syntax element can be a CU level color transform enabled flag indicating that residual blocks of a current CU are processed with a color space conversion. The residual blocks of the current CU can include a luma residual block, a first chroma residual block, and a second chroma residual block.

[0196] At (S1920), in response to receiving the syntax element indicating that the residual blocks of the current CU are processed with the color space conversion, one of two options of color space conversion equations can be selected. In an example, the two options of color space conversion equations can correspond to the Option 1 and Option 2 color space conversion represented by the equations (17)-(20) and equations (21)-(24). In other examples, the two options of color space conversion can take other forms corresponding to other color transform equations.

[0197] In an example, one of two options of color space conversion equations can be selected according to a value of a transform skip flag corresponding to each of the residual blocks of the current CU. The transform skip flags can be received from the bit stream of the coded video. Or, values of the transform skip flags can be interpreted based on other information if the transform skip flags are not signaled in the bit stream of the coded video.

[0198] In an example, one of two options of color space conversion equations can be selected according a QP of each of the residual blocks of the current CU. For example, when the QP has a value of 4, and the transform skip flags of each residual blocks are equal to 1, the Option 1 color transform can be selected. When the QP has a value not equal to 4, the Option 2 color transform can be selected.

[0199] In an example, the selection of one of the two options of color transform can depends on a BDPCM flag of the current CU. For example, when the BDPCM flag indicate a BDPCM mode is applied to a color component of the current CU, the transform skip flag of this color component of the current CU can be interpreted to have a value of 1. The selection of one of the two options can be based on the value of the transform skip flag in combination with values of transform skip flags of other color components of the current CU.

[0200] At (S1930), an inverse color space conversion using the selected option of color space conversion equations can be applied to the residual blocks of the current CU to generate modified versions of the residual blocks of the current CU. The process (1900) can proceed to (S1999) and terminate at (S1999).

certain CANbus devices), or bi-directional, for example to other computer systems using local or wide area digital networks. Certain protocols and protocol stacks can be used on each of those networks and network interfaces as described above.

[0212] Aforementioned human interface devices, human-accessible storage devices, and network interfaces can be attached to a core (2040) of the computer system (2000).

[0213] The core (2040) can include one or more Central Processing Units (CPU) (2041), Graphics Processing Units (GPU) (2042), specialized programmable processing units in the form of Field Programmable Gate Areas (FPGA) (2043), hardware accelerators for certain tasks (2044), graphics adapters (2050), and so forth. These devices, along with Read-only memory (ROM) (2045), Random-access memory (2046), internal mass storage such as internal non-user accessible hard drives, SSDs, and the like (2047), may be connected through a system bus (2048). In some computer systems, the system bus (2048) can be accessible in the form of one or more physical plugs to enable extensions by additional CPUs, GPU, and the like. The peripheral devices can be attached either directly to the core's system bus (2048), or through a peripheral bus (2049). In an example, the screen 2010) can be connected to the graphics adapter (2050). Architectures for a peripheral bus include PCI, USB, and the like.

[0214] CPUs (2041), GPUs (2042), FPGAs (2043), and accelerators (2044) can execute certain instructions that, in combination, can make up the aforementioned computer code. That computer code can be stored in ROM (2045) or RAM (2046). Transitional data can also be stored in RAM (2046), whereas permanent data can be stored for example, in the internal mass storage (2047). Fast storage and retrieve to any of the memory devices can be enabled through the use of cache memory, that can be closely associated with one or more CPU (2041), GPU (2042), mass storage (2047), ROM (2045), RAM (2046), and the like.

[0215] The computer readable media can have computer code thereon for performing various computer-implemented operations. The media and computer code can be those specially designed and constructed for the purposes of the present disclosure, or they can be of the kind well known and available to those having skill in the computer software arts.

[0216] As an example and not by way of limitation, the computer system having architecture (2000), and specifically the core (2040) can provide functionality as a result of processor(s) (including CPUs, GPUs, FPGA, accelerators, and the like) executing software embodied in one or more tangible, computer-readable media. Such computer-readable media can be media associated with user-accessible mass storage as introduced above, as well as certain

WAIP: Wide-Angle Intra Prediction

[217] While this disclosure has described several exemplary embodiments, there are alterations, permutations, and various substitute equivalents, which fall within the scope of the disclosure.

WHAT IS CLAIMED IS:

1. A method of video decoding, comprising:

receiving a syntax element from a bitstream of a coded video indicating that residual blocks of a current coding unit (CU) are processed with a color space conversion, the residual blocks of the current CU including a luma residual block, a first chroma residual block, and a second chroma residual block;

in response to receiving the syntax element indicating that the residual blocks of the current CU are processed with the color space conversion, selecting one of two options of color space conversion equations; and

applying an inverse color space conversion using the selected option of color space conversion equations to the residual blocks of the current CU to generate modified versions of the residual blocks of the current CU.

2. The method of claim 1, wherein the two options of color space conversion equations include:

a first option of color space conversion equations including:

tmp = rY[x][y] – (rCb[x][y] >> 1),

rY[x][y] = tmp + rCb[x][y],

rCb[x][y] = tmp – (rCr[x][y] >> 1), and

rCr[x][y] = rCb[x][y] + rCr[x][y]; and

a second option of color space conversion equations including:

tmp = rY[x][y] - rCb[x][y],

rY[x][y] = rY[x][y] + rCb[x][y],

rCb[x][y] = tmp – rCr[x][y], and

rCr[x][y] = tmp + rCr[x][y],

wherein inputs to each option of color space conversion equations include:

an array of luma residual samples of the luma residual block with elements rY[x][y],

an array of chroma residual samples of the first chroma residual block with elements rCb[x][y], and

an array of chroma residual samples of the second chroma residual block with elements rCr[x][y]; and

outputs from each option of color space conversion equations include:

a modified array of luma residual samples of the luma residual block with elements rY[x][y],

a modified array of chroma residual samples of the first chroma residual block with elements rCb[x][y], and

a modified array of chroma residual samples of the second chroma residual block with elements rCr[x][y].

3. The method of claim 2, wherein the selecting one of two options of color space conversion equations includes:

selecting the first option of color space conversion equations when a transform skip flag corresponding to each of the residual blocks of the current CU has a value of 1.

4. The method of claim 2, wherein the selecting one of two options of color space conversion equations includes:

selecting the second option of color space conversion equations when at least one of transform skip flags corresponding to the residual blocks of the current CU has a value of 0.

5. The method of claim 2, wherein the selecting one of two options of color space conversion equations includes:

selecting the first option of color space conversion equations when a transform skip flag corresponding to each of the residual blocks of the current has a value of 1 and a quantization parameter (QP) corresponding to each of the residual blocks of the current CU has a value of 4.

6. The method of claim 2, wherein the selecting one of two options of color space conversion equations includes:

selecting the second option of color space conversion equations when at least one of QPs corresponding to the residual blocks of the current CU has a value not equal to 4.

7. The method of claim 2, further comprising:

receiving a syntax element indicating only the first option of color space conversion equations is applied, the syntax element is one of a slice level syntax element, a picture level syntax element, or a sequence level syntax element.

8. The method of claim 2, further comprising:

receiving a syntax element indicating only the first option of color space conversion equations is applied regardless of a value of a flag indicating whether transform and quantization are bypassed for residuals of a CU, the syntax element is one of a slice level syntax element, a picture level syntax element, or a sequence level syntax element.

9. The method of claim 1, wherein the selecting one of two options of color space conversion equations includes:

selecting one of two options of color space conversion equations according to a value of a transform skip flag corresponding to each of the residual blocks of the current CU.

10. The method of claim 1, wherein the selecting one of two options of color space conversion equations includes:

selecting one of two options of color space conversion equations according to a variable indicating whether a block-based differential pulse code modulation (BDPCM) is applied to the current CU.

11. The method of claim 1, further comprising:

determining no color space conversion is applied to the residual blocks of the current CU when a bit depth of luma samples of the current CU is different from a bit depth of chroma samples of the current CU.

12. The method of claim 1, further comprising:

determining no color space conversion is applied to the residual blocks of the current CU when a bit depth of luma samples of the current CU is different from a bit depth of chroma samples of the current CU, and a transform skip flag corresponding to each of the residual blocks of the current CU has a value of 1 or 0.

13. The method of claim 1, further comprising:

determining no color space conversion is applied to the residual blocks of the current CU when a bit depth of luma samples of the current CU is different from a bit depth of chroma samples of the current CU, a QP corresponding to each of the residual blocks of the current CU has a value of 4, and a transform skip flag corresponding to each of the residual blocks of the current CU has a value of 1 or 0.

14. An apparatus of video decoding, comprising circuitry configured to perform the method according to any one of claims 1 to 13.

15. A non-transitory computer-readable medium storing instructions that, when executed by a processor, cause the processor to perform the method according to any one of claims 1 to 13.