COEFFICIENTS CODING IN TRANSFORM SKIP MODE

CROSS REFERENCE TO RELATED APPLICATIONS

[001] ~~Under the applicable patent law and/or rules pursuant to the Paris Convention, this application is made to timely claim~~ This application is based on International Patent Application No. PCT/CN2021/084869, filed on April 1, 2021 which claims the priority to and benefits of International Patent Application No. PCT/CN2020/082983, filed on April 2, 2020. ~~For all purposes under the law, the entire disclosure of the aforementioned application is incorporated by reference as part of the disclosure of this application.~~ All the aforementioned patent applications are hereby incorporated by reference in their entireties.

TECHNICAL FIELD

[002] This patent document relates to image and video coding and decoding.

BACKGROUND

[003] Digital video accounts for the largest bandwidth use on the internet and other digital communication networks. As the number of connected user devices capable of receiving and displaying video increases, it is expected that the bandwidth demand for digital video usage will continue to grow.

SUMMARY

[004] The present document discloses techniques that can be used by video encoders and decoders for processing coded representation of video using control information useful for decoding of the coded representation.

[005] In one example aspect, a video processing method is disclosed. The method includes performing a conversion between a current block of a video and a bitstream of the video, wherein the bitstream conforms to a rule that specifies that a context index offset is used for including a first sign flag of a first coefficient in the bitstream, wherein the rule specifies that a value of the context index offset is based on whether a first coding mode is applied on the current block in the bitstream.

[006] In another example aspect, a video processing method is disclosed. The method includes performing a conversion between a current block of a video and a bitstream of the video, wherein the bitstream conforms to a rule that specifies that sign flags of the current block are

CLAIMS

1. A method of video processing, comprising:

performing a conversion between a current block of a video and a bitstream of the video,

wherein the bitstream conforms to a rule that specifies that a context index offset is used for including a first sign flag of a first coefficient in the bitstream,

wherein the rule specifies that a value of the context index offset is based on whether a first coding mode is applied on the current block in the bitstream.

2. The method of claim 1,

wherein the rule specifies that the value of the context index offset is within a first range from N0 to N1, inclusive, in case that the first coding mode is not applied, and

wherein the rule specifies that the value of the context index offset is within a second range from N2 to N3, inclusive, in case that the first coding mode is applied.

3. The method of claim 2, wherein, in the first coding mode, a difference between a quantized residual and a prediction of the quantized residual is represented in the bitstream using a pulse coding modulation representation.

4. The method of claim 2, wherein N0, N1, N2 and N3 are 0, 2, 3 and 5, respectively.

5. The method of any of claims 3 to 4,

wherein the value of the context index offset is equal to 0 or 3 in response to a first neighboring sign value and a second neighboring sign value meeting a first condition, and the first condition is:

(1) the first neighboring sign value is equal to 0 and a second neighboring sign value is equal to 0, or

(2) the first neighboring sign value is equal to a negative of the second neighboring sign value, and

wherein the first neighboring sign value is equal to 0 in response to a horizontal coordinate of the first coefficient being 0 and is equal to a sign value of a left neighboring

1. A method of video processing, comprising:

performing a conversion between a current block of a video and a bitstream of the video,

wherein the bitstream conforms to a first rule that specifies that a context increment is used for including a first sign flag of a first coefficient in the bitstream,

wherein the first rule specifies that a value of the context increment is based on whether a first coding mode is applied on the current block.

2. The method of claim 1,

wherein the first rule specifies that the value of the context increment is within a first range from N0 to N1, inclusive, in case that the first coding mode is not applied, and

wherein the first rule specifies that the value of the context increment is within a second range from N2 to N3, inclusive, in case that the first coding mode is applied.

3. The method of claim 2, wherein, in the first coding mode, a difference between a quantized residual and a prediction of the quantized residual is represented in the bitstream using a pulse coding modulation representation.

4. The method of claim 2 or 3, wherein the first coding mode is a block-based delta pulse code modulation coding mode.

5. The method of any of claims 2-4, wherein N0, N1, N2 and N3 are 0, 2, 3 and 5, respectively.

6. The method of any of claims 1-5,

wherein the value of the context increment is equal to 0 or 3 in response to a first neighboring sign value and a second neighboring sign value meeting a first condition, and the first condition is:

(1) the first neighboring sign value is equal to 0 and the second neighboring sign value is equal to 0, or

coefficient in response to the horizontal coordinate of the first coefficient not being zero, and wherein the second neighboring sign value is equal to 0 in response to a vertical coordinate of the first coefficient being 0 and is equal to a sign value of an above neighboring coefficient in response to the vertical coordinate of the first coefficient not being 0.

6. The method of claim 5, wherein the value of the context index offset is equal to 0 in response to the first coding mode not being applied on the current block, and the value of the context index offset is equal to 3 in response to the first coding mode being applied on the current block.

7. The method of claim 5, wherein the value of the context index offset is equal to 1 or 4 in response to the first neighboring sign value and the second neighboring sign value not meeting the first condition and in response to the first neighboring sign value and the second neighboring sign value meeting a second condition, and the second condition is:
(1) the first neighboring sign value is greater than or equal to 0 and the second neighboring sign value is greater than or equal to 0.

8. The method of any of claim 7, wherein the value of the context index offset is equal to 1 in response to the first coding mode not being applied on the current block, and the value of the context index offset is equal to 4 in response to the first coding mode being applied on the current block.

9. The method of claim 7, wherein the value of the context index offset is equal to 2 or 5 in response to the first neighboring sign value and the second neighboring sign value not meeting the first condition and the second condition.

10. The method of claim 9, wherein the value of the context index offset is equal to 2 in response to the first coding mode not being applied on the current block, and the value of the context index offset is equal to 5 in response to the first coding mode being applied on the current block.

(2) the first neighboring sign value is equal to a negative of the second neighboring sign value.

7. The method of claim 6, wherein the first neighboring sign value is equal to 0 in response to a horizontal coordinate of a current coefficient scan location being 0 and is equal to a sign value of a left neighboring coefficient in response to the horizontal coordinate of the current coefficient scan location not being zero, and

wherein the second neighboring sign value is equal to 0 in response to a vertical coordinate of the current coefficient scan location being 0 and is equal to a sign value of an above neighboring coefficient in response to the vertical coordinate of the first coefficient not being 0.

8. The method of claim 6, wherein the value of the context increment is equal to 0 in response to the first coding mode not being applied on the current block, and the value of the context increment is equal to 3 in response to the first coding mode being applied on the current block.

9. The method of claim 6, wherein the value of the context increment is equal to 1 or 4 in response to the first neighboring sign value and the second neighboring sign value not meeting the first condition and in response to the first neighboring sign value and the second neighboring sign value meeting a second condition, and the second condition is:

(1) the first neighboring sign value is greater than or equal to 0 and the second neighboring sign value is greater than or equal to 0.

10. The method of claim 9, wherein the value of the context increment is equal to 1 in response to the first coding mode not being applied on the current block, and the value of the context increment is equal to 4 in response to the first coding mode being applied on the current block.

11. The method of any of claims 3 to 4,

wherein a transform skip operation is applied to the current block.

12. The method of claim 2, wherein the rule specifies that N0, N1, N3, and N3 are determined based on any one or more of:

(1) an indication included in a sequence parameter set, a video parameter set, a picture parameter set, a picture header, a slice header, a tile group header, a largest coding unit row, group of largest coding unit, a largest coding unit, or a coding unit,

(2) a block dimension of the current block and/or block dimensions of neighboring blocks of the current block,

(3) a block shape of the current block and/or block shapes of neighboring blocks of the current block,

(4) an indication of a color format of the video,

(5) whether a separate or dual coding tree structure is applied to the current block,

(6) a slice type and/or a picture type to which the current block belongs, and

(7) a number of color components of the current block.

13. A method of video processing, comprising:

performing a conversion between a current block of a video and a bitstream of the video,

wherein the bitstream conforms to a rule that specifies that sign flags of the current block are included in the bitstream using any one of context mode and bypass mode based on a number of remaining context coded bins.

14. The method of claim 13, wherein the rule specifies that the bypass mode is used for including the sign flags in the bitstream in response to the number of remaining context coded bins being less than N, where N is an integer.

15. The method of claim 13, wherein the rule specifies that the context mode is used for including the sign flags in the bitstream in response to the number of remaining context coded bins being greater than or equal to N, where N is an integer.

11. The method of claim 9, wherein the value of the context increment is equal to 2 or 5 in response to the first neighboring sign value and the second neighboring sign value not meeting the first condition and the second condition.

12. The method of claim 11, wherein the value of the context increment is equal to 2 in response to the first coding mode not being applied on the current block, and the value of the context increment is equal to 5 in response to the first coding mode being applied on the current block.

13. The method of any of claims 1-12,

wherein a transform skip operation is applied to the current block.

14. The method of any of claims 1-13, wherein the bitstream conforms to a second rule that specifies use of multiple adaptation parameter set network abstraction layer units for the video,

wherein each adaptation parameter set network abstraction layer unit has a corresponding adaptation parameter set type value,

wherein each adaptation parameter set network abstraction layer unit is associated with a corresponding video layer identifier,

wherein each adaptation parameter set network abstraction layer unit is a prefix unit or a suffix unit, and

wherein the second rule specifies that, responsive to the multiple adaptation parameter set network abstraction layer network abstraction layer units having a particular adaptation parameter set type value, adaptation parameter set identifier values of the multiple adaptation parameter set network abstraction layer units belong to a same identifier space.

15. The method of claim 14, wherein the second rule specifies that multiple adaptation parameter set network abstraction layer units have the particular adaptation parameter set type value independent of multiple video layer identifiers of the multiple adaptation parameter set network abstraction layer units, and whether the multiple adaptation parameter set network abstraction layer units are the prefix unit or the suffix unit.

16. The method of claim 13, wherein the rule specifies that the bypass mode is used for including the sign flags in the bitstream in response to the number of remaining context coded bins being equal to N, where N is an integer.

17. The method of claim 13, wherein the rule specifies that the bypass mode is used for including the sign flags in the bitstream in response to the number of remaining context coded bins being greater than N, where N is an integer.

18. The method of any of claims 14 to 17, wherein N is equal to 4.

19. The method of any of claims 14 to 17, wherein N is equal to 0.

20. The method of any of claims 14 to 17, wherein N is an integer based on:

(1) an indication included in a sequence parameter set, a video parameter set, a picture parameter set, a picture header, a slice header, a tile group header, a largest coding unit row, group of largest coding unit, a largest coding unit, or a coding unit,

(2) a block dimension of the current block and/or block dimensions of neighboring blocks of the current block,

(3) a block shape of the current block and/or block shapes of neighboring blocks of the current block,

(4) an indication of a color format of the video,

(5) whether a separate or dual coding tree structure is applied to the current block,

(6) a slice type and/or a picture type to which the current block belongs, and

(7) a number of color components of the current block.

21. The method of any of claims 13 to 20, wherein the rule specifies that the current block is a transform block or a transform skip block.

22. The method of any of claims 13 to 20, wherein the rule specifies that a block differential pulse-coded modulation mode is applied to the current block.

16. The method of any of claims 1-15, wherein the performing the conversion comprising encoding the video into the bitstream.

17. The method of any of claims 1-15, wherein the performing the conversion comprises decoding the video from the bitstream.

18. An apparatus for processing video data comprising a processor and a non-transitory memory with instructions thereon, wherein the instructions upon execution by the processor, cause the processor to:

perform a conversion between a current block of a video and a bitstream of the video,

wherein the bitstream conforms to a first rule that specifies that a context increment is used for including a first sign flag of a first coefficient in the bitstream,

wherein the first rule specifies that a value of the context increment is based on whether a first coding mode is applied on the current block.

19. A non-transitory computer-readable storage medium storing instructions that cause a processor to:

perform a conversion between a current block of a video and a bitstream of the video,

wherein the bitstream conforms to a first rule that specifies that a context increment is used for including a first sign flag of a first coefficient in the bitstream,

wherein the first rule specifies that a value of the context increment is based on whether a first coding mode is applied on the current block.

20. A non-transitory computer-readable recording medium storing a bitstream of a video which is generated by a method performed by a video processing apparatus, wherein the method comprises:

generating the bitstream of the video,

wherein the bitstream conforms to a first rule that specifies that a context increment is used for including a first sign flag of a first coefficient in the bitstream,

wherein the first rule specifies that a value of the context increment is based on whether a first coding mode is applied on the current block.

23. The method of any of claims 13 to 20, wherein the rule specifies that a block differential pulse coded modulation mode is not applied to the current block.

24. A method of video processing, comprising:

performing a conversion between a current block of a video and a bitstream of the video,

wherein the bitstream conforms to a rule that specifies that a context index offset is used for including in the bitstream sign flags of the current block, and

wherein the rule specifies that the context index offset is determined based on an information of the current block.

25. The method of claim 24, wherein the information includes an indication in a sequence parameter set, a video parameter set, a picture parameter set, a picture header, a slice header, a tile group header, a largest coding unit row, group of largest coding unit, a largest coding unit, or a coding unit.

26. The method of claim 24, wherein the information includes a block dimension of the current block and/or block dimensions of neighboring blocks of the current block.

27. The method of claim 24, wherein the information includes a block shape of the current block and/or block shapes of neighboring blocks of the current block.

28. The method of claim 27, wherein context values of the sign flags are separate for different block shapes.

29. The method of claim 24, wherein the information includes one or more prediction modes of one or more neighboring blocks of the current block.

30. The method of claim 29, wherein the one or more prediction modes includes an intra prediction mode.

31. The method of claim 29, wherein the one or more prediction modes includes an inter prediction mode.

32. The method of any of claims 29 to 31, wherein context values of the sign flags are separate for one or more inter coded blocks and for one or more intra coded blocks.

33. The method of claim 24, wherein the information includes indication of block-based delta pulse code modulation modes of neighboring blocks of the current block.

34. The method of claim 24, wherein the information includes an indication of a color format of the video.

35. The method of claim 24, wherein the information includes whether a separate or dual coding tree structure is applied to the current block.

36. The method of claim 24, wherein the information includes a slice type and/or a picture type to which the current block belongs.

37. The method of claim 24, wherein the information includes a number of color components of the current block.

38. The method of claim 37, wherein context values of the sign flags are separate for luma color component of the current block and for chroma color component of the current block.

39. The method of any of claims 1 to 38, wherein the current block is represented in the bitstream using a binary delta pulse code modulation mode.

40. The method of any of claims 1 to 39, wherein the performing the conversion comprising encoding the video into the bitstream.

41. The method of any of claims 1 to 39, wherein the performing the conversion comprises generating the bitstream from the video, and the method further comprises storing the bitstream in a non-transitory computer-readable recording medium.

42. The method of any of claims 1 to 39, wherein the performing the conversion comprises decoding the video from the bitstream.

43. A video decoding apparatus comprising a processor configured to implement a method recited in one or more of claims 1 to 42.

44. A video encoding apparatus comprising a processor configured to implement a method recited in one or more of claims 1 to 42.

45. A computer program product having computer instructions stored thereon, the instructions, when executed by a processor, causes the processor to implement a method recited in any of claims 1 to 42.

46. A non-transitory computer-readable storage medium that stores a bitstream generated according to the method in any one of claims 1 to 42.

47. A non-transitory computer-readable storage medium storing instructions that cause a processor to implement a method recited in any of claims 1 to 42.

48. A method of bitstream generation, comprising:

generating a bitstream of a video according to a method recited in any of claims 1 to 42, and

storing the bitstream on a computer-readable program medium.

49. A method, an apparatus, a bitstream generated according to a disclosed method or a system described in the present document.