

CLAIMS

1. A method of processing video data, comprising:
 determining, for a conversion between a current video block of a video that is a chroma
 5 block and a bitstream of the video, parameters of cross-component linear model (CCLM) at least
 based on selected chroma samples that are selected from neighboring chroma samples of the
 current video block, wherein the position of the selected chroma samples are derived from a first
 position offset value (F) and a step value (S) value, and wherein the F and S are derived at least
 based on availabilities of the neighboring chroma samples of the current video block and a
 10 dimension of the current video block;
 applying the CCLM on luma samples located in a luma block corresponding to the
 current video block to derive prediction values of the current video block; and
 performing the conversion based on the prediction values.
- 15 2. The method of claim 1, wherein the neighboring chroma samples include left neighboring
 chroma samples, above neighboring chroma samples, above-right neighboring chroma samples,
 or below-left neighboring chroma samples relative to the current video block.
3. The method of claim 1 or 2, wherein $F = \text{Floor}(M/2^i)$, wherein M is a number of the
 20 neighboring chroma samples used to derive the selected chroma samples in horizontal direction,
 or $F = \text{Floor}(N/2^i)$, wherein N is a number of the neighboring chroma samples used to derive the
 selected chroma samples in vertical direction, i is equal to 2 or 3, and the Floor operation is used
 to obtain an integer part of a number.
- 25 4. The method of any one of claims 1 to 3, wherein $S = \text{Max}(1, \text{Floor}(M/2^j))$, or $S = \text{Max}(1, \text{Floor}(N/2^j))$, j is equal to 1 or 2, and the Max operation is used to obtain a maximum of multiple
 numbers.

5. The method of claim 3 or 4, wherein the M and N are both less than or equal to W+H, and determined based on a CCLM mode of the current video block, wherein W and H is a width and a height of the current video block, respectively.

5 6. The method of claim 5, wherein the CCLM mode of the current video block is one of a first CCLM mode that derives the parameters of CCLM based on left neighboring chroma samples and above neighboring chroma samples, a second CCLM mode that derives the parameters of CCLM based on left neighboring chroma samples and below-left neighboring samples, and a third CCLM mode that derives the parameters of CCLM based on above
10 neighboring chroma samples and above-right neighboring chroma sample.

7. The method of any one of claims 3 to 6, wherein in response to selecting two chroma samples in horizontal direction, positions of two selected chroma samples in horizontal direction are $\text{Floor}(M/4)$ and $\text{Floor}(M/4) + \text{Floor}(M/2)$.

15

8. The method of any one of claims 3 to 6, wherein in response to selecting two chroma samples in vertical direction, positions of two selected chroma samples in vertical direction are $\text{Floor}(N/4)$ and $\text{Floor}(N/4) + \text{Floor}(N/2)$.

20 9. The method of any one of claims 3 to 6, wherein in response to selecting four chroma samples in horizontal direction, positions of four selected chroma samples in horizontal direction are $\text{Floor}(M/8)$, $\text{Floor}(M/8) + \text{Floor}(M/4)$, $\text{Floor}(M/8) + 2 * \text{Floor}(M/4)$, $\text{Floor}(M/8) + 3 * \text{Floor}(M/4)$.

25 10. The method of any one of claims 3 to 6, wherein in response to selecting four chroma samples in vertical direction, positions of four selected chroma samples in vertical direction are $\text{Floor}(N/8)$, $\text{Floor}(N/8) + \text{Floor}(N/4)$, $\text{Floor}(N/8) + 2 * \text{Floor}(N/4)$, $\text{Floor}(N/8) + 3 * \text{Floor}(N/4)$.

11. The method of any one of claims 6 to 10, wherein in response to the CCLM mode being
30 the first CCLM mode and the above neighboring chroma samples being available, M is equal to W.

12. The method of any one of claims 6 to 10, wherein in response to the CCLM mode being the third CCLM mode, the above neighboring chroma samples being available, and the above-right neighboring chroma samples being unavailable, M is equal to W.

5

13. The method of any one of claims 6 to 10, wherein in response to the CCLM mode being the first CCLM mode and the left neighboring chroma samples are available, N is equal to H.

10

14. The method of any one of claims 6 to 10, wherein in response to the CCLM mode being the second CCLM mode, the left neighboring chroma samples being available, and the below-left neighboring chroma samples being unavailable, N is equal to H.

15

15. The method of any one of claims 1 to 14, wherein further comprising in response to the neighboring chroma samples of the current video block being unavailable, the prediction values of the current video block are set to a default value,

optionally, the default value is equal to $1 \ll (\text{BitDepth} - 1)$, wherein BitDepth represents the bit-depth of the chroma samples.

20

16. The method of any one of claims 1 to 15, wherein the conversion includes encoding the current video block into the bitstream.

17. The method of any one of claims 1 to 15, wherein the conversion includes decoding the current video block from the bitstream.

25

18. An apparatus for coding 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:

30

determine, for a conversion between a current video block of a video that is a chroma block and a bitstream of the video, parameters of cross-component linear model (CCLM) at least based on selected chroma samples that are selected from neighboring chroma samples of the current video block, wherein the position of the selected chroma samples are derived from a first

position offset value (F) and a step value (S) value, and wherein the F and S are derived at least based on availabilities of the neighboring chroma samples of the current video block and a dimension of the current video block;

5 apply the CCLM on luma samples located in a luma block corresponding to the current video block to derive prediction values of the current video block; and
 perform the conversion based on the prediction values.

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

10 determine, for a conversion between a current video block of a video that is a chroma block and a bitstream of the video, parameters of cross-component linear model (CCLM) at least based on selected chroma samples that are selected from neighboring chroma samples of the current video block, wherein the position of the selected chroma samples are derived from a first position offset value (F) and a step value (S) value, and wherein the F and S are derived at least
 15 based on availabilities of the neighboring chroma samples of the current video block and a dimension of the current video block;

 apply the CCLM on luma samples located in a luma block corresponding to the current video block to derive prediction values of the current video block; and
 perform the conversion based on the prediction values.

20

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:

25 determining, for a conversion between a current video block of a video that is a chroma block and a bitstream of the video, parameters of cross-component linear model (CCLM) at least based on selected chroma samples that are selected from neighboring chroma samples of the current video block, wherein the position of the selected chroma samples are derived from a first position offset value (F) and a step value (S) value, and wherein the F and S are derived at least based on availabilities of the neighboring chroma samples of the current video block and a
 30 dimension of the current video block;

applying the CCLM on luma samples located in a luma block corresponding to the current video block to derive prediction values of the current video block; and generating the bitstream based on the prediction values.