

CLAIMS

1. A method of video processing, comprising:
 - determining chroma weights used for determining a chroma prediction block of a chroma block of a current block of a video by blending predictions of the chroma block according to a rule, and
 - performing a conversion between the current block and a coded representation of the video according to the determining,
 - wherein the rule specifies that the chroma weights are determined from luma weights of a collocated luma block of the current block;
 - wherein the current block is coded with a geometric partitioning mode.
2. The method of claim 1, wherein the rule specifies that the chroma weights are a subset of the luma weights.
3. The method of claim 1 or 2, wherein a chroma format of the current block is 4:2:0 or 4:2:2 or 4:4:4.
4. The method of claim 1, wherein the rule specifies that the chroma weights applied to the chroma block with a size of $M \times N$ are same as luma weights applied to a luma block of the current block with a size of $M \times N$, whereby M and N are integers greater than 0.
5. The method of claim 4, wherein a chroma format of the current block is 4:4:4.
6. A method of video processing, comprising:
 - determining chroma weights used for determining a chroma prediction block of a chroma block of a current block of a video by blending predictions of the chroma block according to a rule; and
 - performing a conversion between the current block and a coded representation of the video according to the determining,

wherein the rule is dependent on a characteristic of a collocated luma block and/or a characteristic of the current block,

wherein the current block is coded with a geometric partitioning mode.

7. The method of claim 6, wherein the characteristic of the collocated luma block includes a height and/or width of the collocated luma block, and the characteristic of the current block includes a color format of the current block and/or a chroma subsampling ratio of the current block.
8. The method of claim 6, wherein the rule specifies that, for the chroma block having a 4:4:4 chroma format, the chroma weights are same as luma weights applied to the collocated luma block.
9. The method of claim 6, wherein the rule specifies that, for the chroma block having a 4:2:0 or 4:2:2 chroma format, the chroma weights are subsampled from luma weights applied to the collocated luma block.
10. The method of claim 9, wherein a chroma weight, $\text{WeightC}[x][y]$, is calculated by $\text{WeightY}[f(x)][g(y)]$, whereby x is an integer between 0 and $W/\text{subWidthC}-1$, y is an integer between 0 and $H/\text{subHeightC}-1$, W and H are a width and a height of the chroma block, subWidthC and subHeightC denote chroma subsampling ratios in width and height directions, respectively, and $\text{WeightY}[a][b]$ denotes a luma weight, a is an integer between 0 and $W-1$ and b is an integer between 0 and $H-1$.
11. The method of claim 6, wherein the rule specifies that the chroma weights depend on a size of the collocated luma block and are same for different color components.
12. The method of claim 11, wherein a chroma component with a size of $W \times H$ uses same weights as a luma component with the size of $W \times H$, whereby W and H are integers that indicate a width and a height of collocated luma block, respectively.

13. The method of claim 6, wherein the rule specifies that the chroma weights depend on a size of the collocated luma block and a color component of the video unit.
14. The method of claim 13, wherein the chroma weights are different for different color components of the video unit.
15. The method of claim 13, wherein the chroma weights are same for two chroma components.
16. The method of any of claims 6 to 15, wherein the chroma weights and/or the luma weights are integers.
17. The method of claim 16, wherein both odd integer and even integer weights are allowed.
18. The method of any of claims 6 to 17, wherein the chroma weights and/or the luma weights are clipped to a range $[M, N]$, whereby M and N are integers.
19. A method of video processing, comprising:
 - performing a conversion between a current block of a video and a coded representation of the video,
 - wherein, during the conversion, a prediction block for the current block is determined by blending predictions of the current block according to a blending weight mask,
 - wherein the blending weight mask is determined according to a rule;
 - wherein the current block is coded with a geometric partitioning mode.
20. The method of claim 19, wherein the rule specifies that the blending weight mask is derived based on one or more predefined tables which contain N elements, whereby N is an integer greater than 0.
21. The method of claim 19, wherein the rule specifies that the blending weight mask is

calculated from computing equations.

22. The method of any of claims 1 to 21, wherein the current block is coded with the geometric partitioning mode in which two or more sub-regions of the current block are obtained by partitioning the current block along an angular partition, horizontal partition, or vertical partition.

23. The method of any of claims 1 to 22, wherein the conversion includes encoding the video into the coded representation.

24. The method of any of claims 1 to 22, wherein the conversion includes decoding the coded representation to generate the video.

25. A video processing apparatus comprising a processor configured to implement a method recited in any one or more of claims 1 to 24.

26. A computer readable medium storing program code that, when executed, causes a processor to implement a method recited in any one or more of claims 1 to 24.

27. A computer readable medium that stores a coded representation or a bitstream representation generated according to any of the above described methods.