CLAIMS

1. A method of video processing, comprising:

performing a conversion between a video comprising one or more video pictures comprising one or more subpictures and a bitstream of the video,

wherein the bitstream conforms to a rule that specifies that, during a subpicture sub-bitstream extraction process, one or more parameters for a scaling window applicable to a subpicture are determined by rewriting values of a first set of syntax elements included in one or more picture parameter set (PPS);

wherein the rewritten values of the first set of syntax elements are equal to calculation results of original values of the first set of syntax elements from an original one or more PPS and original values of a second set of syntax elements from an original one or more sequence parameter set (SPS).

2. The method of claim 1, wherein the one or more parameters comprises at least one of a left offset, a right offset, a top offset, or a bottom offset of the scaling window applicable to the subpicture.

3. The method of claim 1, wherein the first set of syntax elements comprises at least one of :

pps_scaling_win_left_offset, pps_scaling_win_right_offset, pps_scaling_win_top_offset, pps_scaling_win_bottom_offset; and

wherein the second set of syntax elements comprises at least one of:

sps_subpic_ctu_top_left_x[spIdx], sps_subpic_width_minus1[spIdx], sps_subpic_ctu_top_left_y[spIdx], sps_subpic_height_minus1[spIdx], sps_pic_width_max_in_luma_samples, sps_pic_height_max_in_luma_samples.

4. The method of claim 1, wherein the one or more parameters comprises a left offset of the scaling window applicable to the subpicture derived as follows:

subpicScalWinLeftOffset = pps_scaling_win_left_offset − sps_subpic_ctu_top_left_x[spIdx] * CtbSizeY / SubWidthC,

wherein subpicScalWinLeftOffset denotes the left offset of the scaling window applicable to the subpicture,

wherein pps_scaling_win_left_offset indicates original left offset applied to a picture size for scaling ratio calculation,

wherein sps_subpic_ctu_top_left_x[spIdx] indicates an original x-coordinate of a coding tree unit located at a top-left corner of the spIdx-th subpicture,

wherein CtbSizeY is a width or a height of a luma coding tree block or coding tree unit, and

wherein SubWidthC indicates a width of a video block and is obtained from a table according to a chroma format of a video picture including the video block.

5. The method of claim 1, wherein the one or more parameters comprises a right offset of the scaling window derived as follows:

subpicScalWinRightOffset = (rightSubpicBd >= sps_pic_width_max_in_luma_samples) ?

pps_scaling_win_right_offset : pps_scaling_win_right_offset −

(sps_pic_width_max_in_luma_samples − rightSubpicBd) / SubWidthC, and

rightSubpicBd = (sps_subpic_ctu_top_left_x[spIdx] + sps_subpic_width_minus1[spIdx] + 1) * CtbSizeY,

wherein subpicScalWinRightOffset denotes the right offset of the scaling window applicable to the subpicture,

wherein sps_pic_width_max_in_luma_samples specifies an original maximum width, in units of luma samples, of each decoded picture referring to the sequence parameter set,

wherein pps_scaling_win_right_offset indicates an original right offset applied to a picture size for scaling ratio calculation,

wherein SubWidthC indicates a width of a video block and is obtained from a table according to a chroma format of a picture including the video block, and

wherein sps_subpic_ctu_top_left_x[spIdx]indicates an original x-coordinate of a coding tree unit located at a top-left corner of the spIdx-th subpicture,

wherein sps_subpic_width_minus1[spIdx] indicates an original width of the spIdx-th subpicture, and

wherein CtbSizeY is a width or a height of a luma coding tree block or coding tree unit.

6. The method of claim 1, wherein the one or more parameters comprises a top offset of the scaling window derived as follows:

subpicScalWinTopOffset = pps_scaling_win_top_offset −

sps_subpic_ctu_top_left_y[spIdx] * CtbSizeY / SubHeightC,

wherein subpicScalWinTopOffset denotes the top offset of the scaling window applicable to the subpicture,

wherein pps_scaling_win_top_offset indicates an original top offset applied to a picture size for scaling ratio calculation,

wherein sps_subpic_ctu_top_left_y[spIdx]indicates an original y-coordinate of a coding tree unit located at a top-left corner of the spIdx-th subpicture,

wherein CtbSizeY is a width or a height of a luma coding tree block or coding tree unit, and

wherein SubHightC indicates a height of a video block and is obtained from a table according to a chroma format of a picture including the video block.

7. The method of claim 1, wherein the one or more parameters comprises a bottom offset of the scaling window applicable to the subpicture derived as follows:

subpicScalWinBotOffset = (botSubpicBd >= sps_pic_height_max_in_luma_samples) ? pps_scaling_win_bottom_offset : pps_scaling_win_bottom_offset − (sps_pic_height_max_in_luma_samples − botSubpicBd) / SubHeightC, and botSubpicBd = (sps_subpic_ctu_top_left_y[spIdx] +

sps_subpic_height_minus1[spIdx] + 1) * CtbSizeY,

wherein subpicScalWinBotOffset denotes the bottom offset of the scaling window applicable to the subpicture,

wherein sps_pic_height_max_in_luma_samples indicates an original maximum height, in units of luma samples, of each decoded picture referring to the sequence parameter set,

wherein pps_scaling_win_bottom_offset indicates an original bottom offset applied to a picture size for scaling ratio calculation,

wherein SubHeightC indicates a height of a video block and is obtained from a table according to a chroma format of a picture including the video block,

wherein sps_subpic_ctu_top_left_y[spIdx] indicates an original y-coordinate of a coding tree unit located at a top-left corner of the spIdx-th subpicture;

wherein sps_subpic_height_minus1[spIdx] indicates an original height of the spIdx-th subpicture, and

wherein CtbSizeY is a width or a height of a luma coding tree block or coding tree unit.

8. The method of claim 1, wherein the conversion includes encoding the video into the bitstream.

9. The method of claim 1, wherein the conversion includes decoding the video from the bitstream.

10. 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 video comprising one or more video pictures comprising one or more subpictures and a bitstream of the video,

wherein the bitstream conforms to a rule that specifies that, during a subpicture sub-bitstream extraction process, one or more parameters for a scaling window applicable to a subpicture are determined by rewriting values of a first set of syntax elements included in one or more picture parameter set (PPS);

wherein the rewritten values of the first set of syntax elements are equal to calculation results of original values of the first set of syntax elements from an original one or more PPS and original values of a second set of syntax elements from an original one or more sequence parameter set (SPS).

11. The apparatus of claim 10, wherein the one or more parameters comprises at least one of a left offset, a right offset, a top offset, or a bottom offset of the scaling window applicable to the subpicture.

12. The apparatus of claim 10, wherein the first set of syntax elements comprises at least one of :

pps_scaling_win_left_offset, pps_scaling_win_right_offset, pps_scaling_win_top_offset, pps_scaling_win_bottom_offset; and

wherein the second set of syntax elements comprises at least one of:

sps_subpic_ctu_top_left_x[spIdx], sps_subpic_width_minus1[spIdx], sps_subpic_ctu_top_left_y[spIdx], sps_subpic_height_minus1[spIdx], sps_pic_width_max_in_luma_samples, sps_pic_height_max_in_luma_samples.

13. The apparatus of claim 10, wherein the one or more parameters comprises a left offset of the scaling window applicable to the subpicture derived as follows:

subpicScalWinLeftOffset = pps_scaling_win_left_offset − sps_subpic_ctu_top_left_x[spIdx] * CtbSizeY / SubWidthC,

wherein subpicScalWinLeftOffset denotes the left offset of the scaling window applicable to the subpicture,

wherein pps_scaling_win_left_offset indicates original left offset applied to a picture size for scaling ratio calculation,

wherein sps_subpic_ctu_top_left_x[spIdx] indicates an original x-coordinate of a coding tree unit located at a top-left corner of the spIdx-th subpicture,

wherein CtbSizeY is a width or a height of a luma coding tree block or coding tree unit, and

wherein SubWidthC indicates a width of a video block and is obtained from a table according to a chroma format of a video picture including the video block.

14. The apparatus of claim 10, wherein the one or more parameters comprises a right offset of the scaling window derived as follows:

subpicScalWinRightOffset = (rightSubpicBd >= sps_pic_width_max_in_luma_samples) ?

pps_scaling_win_right_offset : pps_scaling_win_right_offset −

(sps_pic_width_max_in_luma_samples − rightSubpicBd) / SubWidthC, and

rightSubpicBd = (sps_subpic_ctu_top_left_x[spIdx] + sps_subpic_width_minus1[spIdx] + 1) * CtbSizeY,

wherein subpicScalWinRightOffset denotes the right offset of the scaling window applicable to the subpicture,

wherein sps_pic_width_max_in_luma_samples specifies an original maximum width, in units of luma samples, of each decoded picture referring to the sequence parameter set,

wherein pps_scaling_win_right_offset indicates an original right offset applied to a picture size for scaling ratio calculation,

wherein SubWidthC indicates a width of a video block and is obtained from a table according to a chroma format of a picture including the video block, and

wherein sps_subpic_ctu_top_left_x[spIdx]indicates an original x-coordinate of a coding tree unit located at a top-left corner of the spIdx-th subpicture,

wherein sps_subpic_width_minus1[spIdx] indicates an original width of the spIdx-th subpicture, and

wherein CtbSizeY is a width or a height of a luma coding tree block or coding tree unit.

15. The apparatus of claim 10, wherein the one or more parameters comprises a top offset of the scaling window derived as follows:

subpicScalWinTopOffset = pps_scaling_win_top_offset −

sps_subpic_ctu_top_left_y[spIdx] * CtbSizeY / SubHeightC,

wherein subpicScalWinTopOffset denotes the top offset of the scaling window applicable to the subpicture,

wherein pps_scaling_win_top_offset indicates an original top offset applied to a picture size for scaling ratio calculation,

wherein sps_subpic_ctu_top_left_y[spIdx]indicates an original y-coordinate of a coding tree unit located at a top-left corner of the spIdx-th subpicture,

wherein CtbSizeY is a width or a height of a luma coding tree block or coding tree unit, and

wherein SubHightC indicates a height of a video block and is obtained from a table according to a chroma format of a picture including the video block.

16. The apparatus of claim 10, wherein the one or more parameters comprises a bottom offset of the scaling window applicable to the subpicture derived as follows:

subpicScalWinBotOffset = (botSubpicBd >= sps_pic_height_max_in_luma_samples) ? pps_scaling_win_bottom_offset : pps_scaling_win_bottom_offset − (sps_pic_height_max_in_luma_samples − botSubpicBd) / SubHeightC, and botSubpicBd = (sps_subpic_ctu_top_left_y[spIdx] + sps_subpic_height_minus1[spIdx] + 1) * CtbSizeY,

wherein subpicScalWinBotOffset denotes the bottom offset of the scaling window applicable to the subpicture,

wherein sps_pic_height_max_in_luma_samples indicates an original maximum height, in units of luma samples, of each decoded picture referring to the sequence parameter set,

wherein pps_scaling_win_bottom_offset indicates an original bottom offset applied to a picture size for scaling ratio calculation,

wherein SubHeightC indicates a height of a video block and is obtained from a table according to a chroma format of a picture including the video block,

wherein sps_subpic_ctu_top_left_y[spIdx] indicates an original y-coordinate of a coding tree unit located at a top-left corner of the spIdx-th subpicture;

wherein sps_subpic_height_minus1[spIdx] indicates an original height of the spIdx-th subpicture, and

wherein CtbSizeY is a width or a height of a luma coding tree block or coding tree unit.

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

perform a conversion between a video comprising one or more video pictures comprising one or more subpictures and a bitstream of the video,

wherein the bitstream conforms to a rule that specifies that, during a subpicture sub-bitstream extraction process, one or more parameters for a scaling window applicable to a subpicture are determined by rewriting values of a first set of syntax elements included in one or more picture parameter set (PPS);

wherein the rewritten values of the first set of syntax elements are equal to calculation results of original values of the first set of syntax elements from an original one or more PPS and original values of a second set of syntax elements from an original one or more sequence parameter set (SPS).

18. The non-transitory computer-readable storage medium of claim 17, wherein the one or more parameters comprises at least one of a left offset, a right offset, a top offset, or a bottom offset of the scaling window applicable to the subpicture.