CLAIMS

1. A method of processing video data, comprising:

determining, during a conversion between a current chroma video block of a video and a bitstream of the video, that a scaling process is applied on chroma residual samples of the current chroma video block; and

performing the conversion by applying the scaling process on the chroma residual samples,

wherein in the scaling process, the chroma residual samples are scaled based on at least one scaling factor before being used to reconstruct the current chroma video block, and

wherein the at least one scaling factor is derived based on an averaged luma variable computed based on neighboring luma samples of a video unit of the video which is determined based on a luma sample corresponding to a top-left sample of the current chroma video block, and

wherein the at least one scaling factor is derived by:

checking an availability of each of one or more neighboring luma blocks of the video unit, wherein the each of the one or more neighboring luma blocks comprise at least one sample of the neighboring luma samples,

determining, based on the availability of the each of the one or more neighboring luma blocks, whether to retrieve the neighboring luma samples of the video unit,

deriving the at least one scaling factor based on the averaged luma variable computed using the neighboring luma samples by a rounding-based average operation in response to the neighboring luma blocks being available, and

deriving the at least one scaling factor by setting the averaged luma variable equal to a default value in case that the neighboring luma samples being available are absent, wherein the default value equals to 1 << (bitDepth − 1), where bitDepth is a bit depth of the video.

2. The method of claim 1, wherein the neighboring luma samples are located in predefined positions neighboring to the video unit.

3. The method of claim 2, wherein the neighboring luma samples located in the predefined positions neighboring to the video unit include reconstructed luma samples external to the video unit.

4. The method of claim 2, wherein the neighboring luma samples located in the predefined positions neighboring to the video unit include reconstructed luma samples adjacent to the video unit.

5. The method of claim 4, wherein the reconstructed luma samples adjacent to the video unit include at least one of one or more left neighboring luma sample columns or one or more above neighboring luma sample rows of the video unit.

6. The method of any of claims 1-5, wherein a total number of the neighboring luma samples of the video unit is N, where N is an integer greater than 1, and a range of N depends on a size information of the video unit.

7. The method of claim 1, wherein the checking the availability of each of one or more neighboring luma blocks comprises: checking the availability of each of one or more neighboring luma blocks based on at least one of a width and a height of a video region associated with the current chroma video block.

8. The method of claim 7, wherein in response to a top-left sample of the one neighboring block locating outside the video region, the neighboring block is treated as unavailable.

9. The method of claim 7, wherein, for a luma video block of the video region, at least one of the following is performed:

1) a forward mapping process for the luma video block, in which prediction samples of the luma video block are converted from an original domain to a reshaped domain; or

2) an inverse mapping process, which is an inverse operation of the forward mapping process, in which reconstructed samples of the luma video block in the reshaped domain are converted to the original domain, and

wherein the neighboring luma samples include reconstructed samples in the reshaped domain.

10. The method of claim 7, wherein the video region is a picture.

11. The method of any of claims 1-10, wherein the scaling process is based on a piecewise linear model, and wherein an index identifying a piece to which the averaged luma variable belongs, and the at least one scaling factor is derived based on the index.

12. The method of any of claims 1-11, wherein the conversion includes encoding the current chroma video block into the bitstream.

13. The method of any of claims 1-11, wherein the conversion includes decoding the current chroma video block from the bitstream.

14. 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:

determine, during a conversion between a current chroma video block of a video and a bitstream of the video, that a scaling process is applied on chroma residual samples of the current chroma video block; and

perform the conversion by applying the scaling process on the chroma residual samples,

wherein in the scaling process, the chroma residual samples are scaled based on at least one scaling factor before being used to reconstruct the current chroma video block, and

wherein the at least one scaling factor is derived based on an averaged luma variable computed based on neighboring luma samples of a video unit of the video which is determined based on a luma sample corresponding to a top-left sample of the current chroma video block, and

wherein the at least one scaling factor is derived by:

checking an availability of each of one or more neighboring luma blocks of the video unit, wherein the each of the one or more neighboring luma blocks comprise at least one sample of the neighboring luma samples,

determining, based on the availability of the each of the one or more neighboring luma blocks, whether to retrieve the neighboring luma samples of the video unit,

deriving the at least one scaling factor based on the averaged luma variable computed using the neighboring luma samples by a rounding-based average operation in response to the neighboring luma blocks being available, and

deriving the at least one scaling factor by setting the averaged luma variable equal to a default value in case that the neighboring luma samples being available are absent, wherein the default value equals to 1 << (bitDepth − 1), where bitDepth is a bit depth of the video.

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

determine, during a conversion between a current chroma video block of a video and a bitstream of the video, that a scaling process is applied on chroma residual samples of the current chroma video block; and

perform the conversion by applying the scaling process on the chroma residual samples,

wherein in the scaling process, the chroma residual samples are scaled based on at least one scaling factor before being used to reconstruct the current chroma video block, and

wherein the at least one scaling factor is derived based on an averaged luma variable computed based on neighboring luma samples of a video unit of the video which is determined based on a luma sample corresponding to a top-left sample of the current chroma video block, and

wherein the at least one scaling factor is derived by:

checking an availability of each of one or more neighboring luma blocks of the video unit, wherein the each of the one or more neighboring luma blocks comprise at least one sample of the neighboring luma samples,

determining, based on the availability of the each of the one or more neighboring luma blocks, whether to retrieve the neighboring luma samples of the video unit,

deriving the at least one scaling factor based on the averaged luma variable computed using the neighboring luma samples by a rounding-based average operation in response to the neighboring luma blocks being available, and

deriving the at least one scaling factor by setting the averaged luma variable equal to a default value in case that the neighboring luma samples being available are absent, wherein the default value equals to 1 << (bitDepth − 1), where bitDepth is a bit depth of the video..

16. A method for storing a bitstream of a video, comprising:

determining that a scaling process is applied on chroma residual samples of the current chroma video block; and

generating the bitstream by applying the scaling process on the chroma residual samples, and

storing the bitstream in a non-transitory computer-readable recording medium,

wherein in the scaling process, the chroma residual samples are scaled based on at least one scaling factor before being used to reconstruct the current chroma video block, and

wherein the at least one scaling factor is derived based on an averaged luma variable computed based on neighboring luma samples of a video unit of the video which is determined based on a luma sample corresponding to a top-left sample of the current chroma video block, and

wherein the at least one scaling factor is derived by:

checking an availability of each of one or more neighboring luma blocks of the video unit, wherein the each of the one or more neighboring luma blocks comprise at least one sample of the neighboring luma samples,

determining, based on the availability of the each of the one or more neighboring luma blocks, whether to retrieve the neighboring luma samples of the video unit,

deriving the at least one scaling factor based on the averaged luma variable computed using the neighboring luma samples by a rounding-based average operation in response to the neighboring luma blocks being available, and

deriving the at least one scaling factor by setting the averaged luma variable equal to a default value in case that the neighboring luma samples being available are absent, wherein the default value equals to 1 << (bitDepth − 1), where bitDepth is a bit depth of the video.