CLAIMS

What is claimed is:

1. A method of video processing, comprising:

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

wherein the bitstream conforms to a formatting rule specifying that a size of a merge estimation region (MER) is indicated in the bitstream;

wherein the size of the MER is based on a dimension of a video unit; and

wherein the MER comprises a region used for deriving a motion candidate for the conversion.

2. The method of claim 1, wherein the video unit comprises a coding unit or a coding tree unit.

3. The method of claim 1 or 2, wherein the dimension of the video unit comprises at least a width, a height, or an area of the video unit.

4. The method of any of claims 1 to 3, wherein the dimension of the MER is constrained to be smaller than the dimension of the video unit.

5. The method of any of claims 1 to 3, wherein the dimension of the MER is constrained to be smaller than or equal to the dimension of the video unit.

6. The method of any of claims 1 to 5, wherein the dimension of the MER is indicated as an index value in the bitstream.

7. The method of claim 6, wherein the index value has a one-to-one mapping relationship with the dimension of the MER.

8. The method of claim 6, wherein the dimension of the MER or the index value is coded in the bitstream based on an exponential Golomb code.

9. The method of claim 6, wherein the dimension of the MER or the index value is coded in the bitstream based on a unary code, a rice code, or a fixed length code.

10. The method of any of claims 6 to 9, wherein the index indicating the dimension of the MER is represented as S-Δ or M-S in the bitstream representation, wherein S represents the dimension of the MER, and wherein Δ and/or M are integer values.

11. The method of claim 10, where the Δ and/or M are determined based on the dimension of the maximum or minimum video unit.

12. The method of claim 11, wherein Δ is equal to the dimension of the minimum video unit.

13. The method of claim 11, wherein M is equal to the dimension of the maximum video unit.

14. The method of claim 11, wherein Δ is equal to (the dimension of the minimum video unit + offset), offset being an integer.

15. The method of claim 11, wherein M is equal to (the dimension of the maximum video unit + offset), offset being an integer.

16. The method of claim 14 or 15, wherein the offset is equal to 1 or -1.

17. A method of video processing, comprising:

performing a conversion between a block of a video and a bitstream of the video in a palette coding mode in which a palette of representative sample values is used for coding the block of video in the bitstream,

wherein a maximum number of palette size or palette predictor size used in the palette mode is restricted to m×N, and

wherein m and N are positive integers.

18. The method of claim 17, wherein N is equal to 8.

19. The method of claim 17 or 18, wherein a value associated with m is signaled as a syntax element in the bitstream.

20. The method of claim 19, wherein the value comprises m or m+offset, wherein offset is an integer.

21. The method of claim 19, wherein the syntax element is binarized in the bitstream based on unary coding, exponential Golomb coding, rice coding, or fixed length coding.

22. A method of video processing, comprising:
determining, for a conversion between a current block of a video and a bitstream of the video, that a deblocking filtering process is disabled for a boundary of the current block in case the boundary coincides with a boundary of a sub-picture having a sub-picture index X and a loop filtering operation is disabled across boundaries of the subpicture, X being a non-negative integer; and
performing the conversion based on the determining.

23. The method of claim 22, wherein the deblocking filtering process is applicable to vertical boundaries, and wherein the deblocking filtering process is disabled for a left boundary of the current block in case the left boundary coincides with a left or a right boundary of the sub-picture having the sub-picture index X and the loop filtering operation is disabled across boundaries of the subpicture.

24. The method of claim 22, wherein the deblocking filtering process is applicable to horizontal boundaries, and wherein the deblocking filtering process is disabled for a top boundary of the current block in case the top boundary coincides with a top or a bottom boundary of the sub-picture having the sub-picture index X and the loop filtering operation is disabled across boundaries of the subpicture.

~~25. The method of any of claims 1 to 24, wherein the conversion decoding the video from the bitstream.~~

~~26. The method of any of claims 1 to 24, wherein the conversion encoding the bitstream from the video.~~

~~27. A method for storing bitstream of a video, comprising:~~

~~generating a bitstream of the video from a block; and~~

~~storing the bitstream in a non-transitory computer-readable recording medium, wherein the bitstream conforms to a formatting rule that specifies a size of a merge estimation region (MER) is indicated in the bitstream, wherein the size of the MER is based on a dimension of a size of a video unit, and wherein the MER comprises a region used for deriving a motion candidate for the conversion.~~

~~28. A method for storing bitstream of a video, comprising:~~

~~applying, during a conversion between a block of a video and a bitstream of the video, a palette coding mode in which a palette of representative sample values is used for coding the block of video in the bitstream;~~

~~generating the bitstream from the block based on the applying; and~~

~~storing the bitstream in a non-transitory computer-readable recording medium, wherein a maximum number of palette size or palette predictor size used in the palette mode is restricted to m×N, m and N being positive integers.~~

~~29. A method for storing bitstream of a video, comprising:~~

~~determining that a deblocking filtering process is disabled for a boundary of a current block in case the boundary coincides with a boundary of a sub-picture having a sub-picture index X and a loop filtering operation is disabled across boundaries of the subpicture, X being a non-negative integer;~~

~~generating the bitstream from the current block based on the determining; and~~

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

~~30. A video processing apparatus comprising a processor configured to implement a method recited in any one or more of claims 1 to 29.~~

~~31. A computer-readable medium having code stored thereon, the code, when executed, causing a processor to implement a method recited in any one or more of claims 1 to 29.~~

~~32. A computer readable medium that stores a bitstream generated according to any of claims 1 to 29.~~

~~33. A method, an apparatus, a bitstream generated according to a method or a system described in the present document.~~

1. A method of processing video data, comprising:

determining, for a conversion between a current block of a video and a bitstream of the video, that a first prediction mode is applied to the current block,

maintaining a predictor palette table,

constructing, in the first prediction mode, a current palette comprising one or more palette predictors for the current block based on the predictor palette table, and

performing the conversion based on the first prediction mode,

wherein in the first prediction mode, reconstructed samples of the current block are represented by a set of representative color values, and the set of representative color values comprises at least one of 1) palette predictors derived from the current palette, 2) escaped samples, or 3) palette information included in the bitstream,

wherein a maximum number of entries in the current palette is restricted to m×8, and a maximum number of entries in the predictor palette table are restricted to n×8, and wherein m and n are positive integers.

2. The method of claim 1, wherein m or n is equal to 4 or 8.

3. The method of claim 1 or claim 2, wherein a syntax element indicating a parallel merge estimation level is included in the bitstream, and wherein the maximum value of the syntax element depends on a size of a coding tree unit (CTU).

4. The method of any one of claims 1-3, wherein the maximum value of the syntax element is constrained to be smaller than the size of the CTU.

5. The method of any one of claims 1-4, wherein the syntax element is coded by an exponential Golomb code.

6. The method of any one of claims 1-5, wherein the conversion includes encoding the current block into the bitstream.

7. The method of any one of claims 1-5, wherein the conversion includes decoding the current block from the bitstream.

8. 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, for a conversion between a current block of a video and a bitstream of the video, that a first prediction mode is applied to the current block,

maintain a predictor palette table,

construct, in the first prediction mode, a current palette comprising one or more palette predictors for the current block based on the predictor palette table, and

perform the conversion based on the first prediction mode,

wherein in the first prediction mode, reconstructed samples of the current block are represented by a set of representative color values, and the set of representative color values comprises at least one of 1) palette predictors derived from the current palette, 2) escaped samples, or 3) palette information included in the bitstream,

wherein a maximum number of entries in the current palette is restricted to m×8, and a maximum number of entries in the predictor palette table are restricted to n×8, and wherein m and n are positive integers.

9. The apparatus of claim 8, wherein m or n is equal to 4 or 8.

10. The apparatus of claim 8 or claim 9, wherein a syntax element indicating a parallel merge estimation level is included in the bitstream, and wherein the maximum value of the syntax element depends on a size of a coding tree unit (CTU).

11. The apparatus of any one of claims 8-10, wherein the maximum value of the syntax element is constrained to be smaller than the size of the CTU.

12. The apparatus of any one of claims 8-11, wherein the syntax element is coded by an exponential Golomb code.

13. The apparatus of any one of claims 8-12, wherein the conversion includes encoding the current block into the bitstream.

14. The apparatus of any one of claims 8-12, wherein the conversion includes decoding the current block from the bitstream.

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

determine, for a conversion between a current block of a video and a bitstream of the video, that a first prediction mode is applied to the current block,

maintain a predictor palette table,

construct, in the first prediction mode, a current palette comprising one or more palette predictors for the current block based on the predictor palette table, and

perform the conversion based on the first prediction mode,

wherein in the first prediction mode, reconstructed samples of the current block are represented by a set of representative color values, and the set of representative color values comprises at least one of 1) palette predictors derived from the current palette, 2) escaped samples, or 3) palette information included in the bitstream,

wherein a maximum number of entries in the current palette is restricted to m×8, and a maximum number of entries in the predictor palette table are restricted to n×8, and wherein m and n are positive integers.

16. The non-transitory computer-readable storage medium of claim 15, wherein a syntax element indicating a parallel merge estimation level is included in the bitstream, and wherein the maximum value of the syntax element depends on a size of a coding tree unit (CTU).

17. The non-transitory computer-readable storage medium of claim 15 or claim 16, wherein the maximum value of the syntax element is constrained to be smaller than the size of the CTU.

18. Th non-transitory computer-readable storage medium of any one of claims 15-17, wherein the syntax element is coded by an exponential Golomb code.

19. 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:

determining that a first prediction mode is applied to a current block of a video,

maintaining a predictor palette table,

constructing, in the first prediction mode, a current palette comprising one or more palette predictors for the current block based on the predictor palette table, and

generating the bitstream based on the first prediction mode,

wherein in the first prediction mode, reconstructed samples of the current block are represented by a set of representative color values, and the set of representative color values comprises at least one of 1) palette predictors derived from the current palette, 2) escaped samples, or 3) palette information included in the bitstream,

wherein a maximum number of entries in the current palette is restricted to m×8, and a maximum number of entries in the predictor palette table are restricted to n×8, and wherein m and n are positive integers.

20. The non-transitory computer-readable recording medium of claim 19, wherein a syntax element indicating a parallel merge estimation level is included in the bitstream, and wherein the maximum value of the syntax element depends on a size of a coding tree unit (CTU),

wherein the maximum value of the syntax element is constrained to be smaller than the size of the CTU.