

WHAT IS CLAIMED IS:**CLAIMS**

1.- A device for video decoding an encoded current block of video data, the device comprising:

_a memory configured to store a reconstructed current block of video data; and

_one or more processors configured to:

~~receive a process~~ one or more bits, in a bitstream, that represent a flag, and indicate the encoded current block of video data was encoded based on a unified candidate list, ~~wherein;~~ construct the unified candidate list ~~includes motion vector candidates based on~~ for the current block, including: determine one or more adjacent neighboring blocks to the current block that are predicted using one or more translational motion vectors; ~~and motion vector candidates based on~~ and add the one or more adjacent neighboring blocks as normal merge candidates to the unified candidate list; and determine one or more adjacent neighboring blocks to the current block that are predicted using one or more affine motion vectors;

~~receive and adding the one or more adjacent neighboring blocks as affine merge candidates to the unified candidate list such that all of the affine merge candidates follow all of the normal merge candidates of adjacent neighboring blocks in the unified candidate list;~~ process a merge index in the bitstream, wherein the merge index indicates which a candidate in the unified candidate list;

to be used to form a predictor block for the current block, the candidate being one of the normal merge candidates or the affine merge candidates; select one or more motion vectors of the candidate from the unified candidate list, based on the merge index, wherein the candidate has one or more of the motion vectors corresponding to the translational motion vectors or affine motion vectors within the unified candidate list;

~~determine~~ merge index; form, based on the selected one or more motion vectors from the candidate list, ~~a, the predictor block of video data~~ for the current block; and

reconstruct the current block of video data based on the predictor block of video data.

2. The device of claim 1, wherein the ~~candidate is a normal merge candidate in the unified candidate list when it includes one or more motion vectors corresponding to the translational motion vectors;~~ at most seven normal merge candidates and ~~wherein the candidate is an~~ at most four ~~affine motion merge candidate when it includes one or more motion vectors corresponding to affine motion candidates.~~

3. -The device of claim 21, wherein to form the predictor block for the current block, the one or more processors are configured to: select when the motion vectors candidate is one of the normal merge candidate from candidates, perform temporal motion compensation using the unified list and select the translational motion vectors of the candidate to form the predictor block; or when the candidate is one of the affine merge candidate from the unified list, based on a prioritization order

~~of candidates, perform affine motion compensation using the normal merge candidate and affine merge candidate in motion vectors of the unified candidate list, to form the predictor block.~~

4.- The device of claim 3, wherein when the normal merge candidate in the unified candidate list is prioritized, based on the prioritization order, over the affine merge candidate in the unified candidate list.

5. The device of claim 4, wherein the normal merge candidate from a left block relative to the current block of video data is prioritized, based on the prioritization order, over the affine merge candidate from a left block relative to the current block of video data.

6. The device one of claim 4, wherein the normal merge candidate from a neighboring block in one frame, in the unified candidate list, is prioritized, based on the prioritization order, over the affine merge candidate from a neighboring block in a different frame, in the unified candidate list.

7. The device of claim 4, wherein the normal merge candidate from a non-adjacent spatial neighboring block, is prioritized, based on the prioritization order, over the affine merge candidate from the non-adjacent spatial neighboring block.

8. The device of claim 3, wherein the affine merge candidate in the unified candidate list is prioritized, based on the prioritization order, over the normal merge candidate in the unified candidate list.

9. The device of claim 8, wherein the affine merge candidate from a left block relative to the current block of video data is prioritized, based on the prioritized order, over the normal merge candidate from a left block relative to the current block of video data.

10. The device of claim 8, wherein the affine merge candidate from a neighboring block in one frame, in the unified candidate list, is prioritized, based on the prioritization order, over the normal merge candidate from a neighboring block in a different frame, in the unified candidate list.

11. The device of claim 8, wherein the affine merge candidate from a non-adjacent spatial neighboring block, is prioritized, based on the prioritization order, over the normal merge candidate from the non-adjacent spatial neighboring block.

~~12. The device of claim 3, wherein the unified candidate list includes at most seven normal merge candidates in the unified candidate list, and at most four affine merge candidates in the unified candidate list.~~

~~13. The device of claim 3, wherein the unified candidate list includes one or more normal merge candidates in the unified candidate list, and one or more affine merge candidates in the unified candidate list.~~

~~14. The device of claim 3, wherein the one or more processors are configured to apply affine motion compensation with the selected one or more motion vectors of the affine merge candidate.~~

~~15. The device of claim 14, wherein the selected one or more motion vectors of the affine merge candidate corresponds to a four-parameter affine model or a six-parameter affine model.~~

~~16.~~ 5. A method for video decoding an encoded current block of video data, comprising:

~~receiving~~ processing one or more bits, in a bitstream, that represent a flag, and indicate the encoded current block of video data was encoded based on a unified candidate list; ~~wherein;~~ constructing the unified candidate list ~~includes motion vector candidates based on~~ for the current block, including: determining one or more adjacent neighboring blocks to the current block that are predicted using one or more translational motion vectors; and ~~motion vector candidates based on one or more affine motion vectors;~~

~~receiving~~ adding the one or more adjacent neighboring blocks as normal merge candidates to the unified candidate list and determining one or more adjacent neighboring blocks to the current block that are predicted using one or more affine motion vectors and adding the one or more adjacent neighboring blocks as affine merge candidates to the unified candidate list such that all of the affine merge candidates follow all of the normal merge candidates of adjacent neighboring blocks in the unified candidate list; processing a merge index in the bitstream, wherein the merge index indicates ~~which a~~ candidate in the unified candidate list;

to be used to form a predictor block for the current block, the candidate being one of the normal merge candidates or the affine merge candidates; selecting one or more motion vectors of the candidate from the unified candidate list, based on the merge index, wherein the candidate has one or more of the motion vectors corresponding to the translational motion vectors or affine motion vectors within the unified candidate list;

~~determining~~ merge index; forming, based on the selected one or more motion vectors from the candidate list, a, the predictor block of video data for the current block; and

reconstructing the current block of video data based on the predictor block of video data.

~~17.6. The method of claim 16, wherein the candidate is a normal merge candidate in the unified candidate list when it includes one or more motion vectors corresponding to the translational motion vectors, and wherein the candidate is an affine motion merge candidate when it includes one or more motion vectors corresponding to affine motion.~~

~~18. The method of claim 17, wherein the selecting the motion vectors of the normal merge candidate from the unified list and select the motion vectors of the affine merge candidate from the unified list, based on a prioritization order of the normal merge candidate and affine merge candidate in the unified candidate list.~~

~~19. The method of claim 18, wherein the normal merge candidate in the unified candidate list is prioritized, based on the prioritization order, over the affine merge candidate in the unified candidate list.~~

~~20. The method of claim 19, wherein the normal merge candidate from a left block relative to the current block of video data is prioritized, based on the prioritization order, over the affine merge candidate from a left block relative to the current block of video data.~~

~~21. The method of claim 19, wherein the normal merge candidate from a neighboring block in one frame, in the unified candidate list, is prioritized, based on the prioritization order, over the affine merge candidate from a neighboring block in a different frame, in the unified candidate list.~~

~~22. The method of claim 19, wherein the normal merge candidate from a non-adjacent spatial neighboring block, is prioritized, based on the prioritization order, over the affine merge candidate from the non-adjacent spatial neighboring block.~~

~~23. The method of claim 18, wherein the affine merge candidate in the unified candidate list is prioritized, based on the prioritization order, over the normal merge candidate in the unified candidate list.~~

~~24. The method of claim 23, wherein the affine merge candidate from a left block relative to the current block of video data is prioritized, based on the prioritized order, over the normal merge candidate from a left block relative to the current block of video data.~~

~~25. The method of claim 23, wherein the affine merge candidate from a neighboring block in one frame, in the unified candidate list, is prioritized, based on the prioritization order, over the normal merge candidate from a neighboring block in a different frame, in the unified candidate list.~~

26. The method of claim 23, wherein the affine merge candidate from a non-adjacent spatial neighboring block, is prioritized, based on the prioritization order, over the normal merge candidate from the non-adjacent spatial neighboring block.

27. The method of claim 185, wherein the unified candidate list includes at most seven normal merge candidates in the unified candidate list, and at most four affine merge candidates in the unified candidate list.

28. The method of claim 18, wherein the unified candidate list includes one or more normal merge candidates in the unified candidate list, and one or more affine merge candidates in the unified candidate list.

29. 7. An apparatus for video decoding an encoded current block of video data, comprising:

means for receiving processing one or more bits, in a bitstream, that represent a flag, and indicate the encoded current block of video data was encoded based on a unified candidate list, ~~wherein;~~
means for constructing the unified candidate list includes motion vector candidates based on for the current block, including: means for determining one or more adjacent neighboring blocks to the current block that are predicted using one or more translational motion vectors, and ~~motion vector candidates based on~~ adding the one or more affine motion vectors;

~~means for receiving~~ adjacent neighboring blocks as normal merge candidates to the unified candidate list; and means for determining one or more adjacent neighboring blocks to the current block that are predicted using one or more affine motion vectors and adding the one or more adjacent neighboring blocks as affine merge candidates to the unified candidate list such that all of the affine merge candidates follow all of the normal merge candidates of adjacent neighboring blocks in the unified candidate list; means for processing a merge index in the bitstream, wherein the merge index indicates ~~which~~ a candidate in the unified candidate list;

to be used to form a predictor block for the current block, the candidate being one of the normal merge candidates or the affine merge candidates; means for selecting one or more motion vectors of the candidate from the unified candidate list, ~~based on~~ the candidate corresponding to the merge index, ~~wherein the candidate has one or more of the motion vectors corresponding to the translational motion vectors or affine motion vectors within the unified candidate list;~~

means for determining forming, based on the selected one or more motion vectors from the candidate list, a, the predictor block of video data for the current block; and

means for reconstructing the current block of video data based on the predictor block of video data.

30-8. A device for video encoding, the device comprising:

one or more processors configured to:

~~select one or more motion vectors corresponding to translational motion of a normal merge candidate;~~

~~select one or more motion vectors corresponding to affine motion of an affine merge candidate;~~

~~construct a unified candidate list comprising for a current block of video data, including: determine one or more than one normal merge candidate, adjacent neighboring blocks to the current block of video data that are predicted using one or more translational motion vectors and more than one affine merge candidate based on a prioritization order of the add the one or more than one affine adjacent neighboring blocks as normal merge candidates to the unified candidate and the list; determine one or more than one normal merge candidate, wherein the adjacent neighboring blocks to the current block that are predicted using one or more than one normal merge candidate affine motion vectors and add the one or more than one affine merge candidate are associated with an adjacent neighboring blocks as affine merge candidates to the unified candidate list; determine to form a predictor block for the current block using the motion vectors of one of the candidates in the unified candidate list select a merge index corresponding to the one of the unified candidate list;~~

~~select a merge index from the index of the unified candidate list, based on a cost associated with the normal merge candidate, or the affine merge candidate candidates in the unified candidate list; and~~

~~provide the merge index in a bitstream; and~~

~~a memory configured to store the merge index video data.~~

9. The device of claim 1, wherein the one or more processors are further configured to, when the candidate comprises one of the affine merge candidates: determine a first size of the affine merge candidate; determine a second size of the current block; and modify the affine motion vectors of the affine merge candidate according to the first size and the second size.

10. The device of claim 1, wherein the one or more processors are further configured to: determine one or more non-adjacent neighboring blocks to the current block; and add the non-adjacent neighboring blocks as non-adjacent candidates to the unified candidate list following the affine merge candidates.

11. The method of claim 5, further comprising, when the candidate comprises one of the affine merge candidates: determining a first size of the affine merge candidate; determining a second size of the current block; and modifying the affine motion vectors of the affine merge candidate according to the first size and the second size.

12. The method of claim 5, further comprising: determining one or more non-adjacent neighboring blocks to the current block; and adding the non-adjacent neighboring blocks as non-adjacent candidates to the unified candidate list following the affine merge candidates.

13. The device of claim 7, further comprising: means for determining a first size of the affine merge candidate when the candidate comprises one of the affine merge candidates; means for determining a second size of the current block when the candidate comprises one of the affine merge candidates; and means for modifying the affine motion vectors of the affine merge candidate according to the first size and the second size when the candidate comprises one of the affine merge candidates.

14. The device of claim 7, further comprising: means for determining one or more non-adjacent neighboring blocks to the current block; and means for adding the non-adjacent neighboring blocks as non-adjacent candidates to the unified candidate list following the affine merge candidates.

15. The device of claim 8, wherein the one or more processors are further configured to, when the candidate comprises one of the affine merge candidates: determine a first size of the affine merge candidate; determine a second size of the current block; and modify the affine motion vectors of the affine merge candidate according to the first size and the second size.

16. The device of claim 8, wherein the one or more processors are further configured to: determine one or more non-adjacent neighboring blocks to the current block; and add the non-adjacent neighboring blocks as non-adjacent candidates to the unified candidate list following the affine merge candidates.