WHAT IS CLAIMED IS:

1. A method of determining an alignment level between motion compensated reference patches, the method comprising:

obtaining, by a decoder, unrounded merge motion vectors between two motion compensated reference patches;

rounding, by the decoder, the unrounded merge motion vectors to obtain rounded motion vectors;

computing, by the decoder, a sum of absolute differences (SAD) between the two motion compensated reference patches by performing motion compensation using the rounded motion vectors;

determining, by the decoder, whether the SAD is less than a coding unit (CU) size-dependent threshold value;

when the SAD is less than the CU size-dependent threshold value:

skipping remaining decoder-side motion vector refinement (DMVR) process steps; and

finalizing the motion compensation; and

when the SAD is not less than the CU size-dependent threshold value:

performing the remaining DMVR process steps using the unrounded merge motion vectors; and

finalizing the motion compensation.

2. The method of claim 1, wherein performing the final motion compensation comprises:

rounding the unrounded merge motion vectors to a nearest integer sample position in a horizontal direction and a vertical direction to obtain integer position samples; and

averaging the integer position samples.

3. The method of claim 1, wherein performing the final motion compensation comprises:

rounding the unrounded merge motion vectors to a nearest half-pixel sample position in a horizontal direction and a vertical direction to obtain half-pixel position samples; and

averaging the half-pixel position samples.

4. The method of claim 1, wherein the motion compensated reference patches each comprise a predetermined size of MxN pixels, M and N being positive integers having a same value.

5. The method of claim 1, wherein the motion compensated reference patches each comprise a predetermined size of MxN pixels, M and N being positive integers having different values.

6. The method of claim 1, wherein the CU size-dependent threshold value is a function of a bit depth of the unrounded merge motion vectors.

7. A method of determining an alignment level between motion compensated reference patches, the method comprising:

obtaining, by a decoder, motion compensated interpolated samples based on sub-pixel accurate merge motion vectors from a bilinear motion compensated interpolation;

computing, by the decoder, a sum of absolute differences (SAD) between two motion compensated reference patches using a subset of the motion compensated interpolated samples;

determining, by the decoder, whether the SAD is less than a coding unit (CU) size-dependent threshold value;

when the SAD is less than the CU size-dependent threshold value:

skipping remaining decoder-side motion vector refinement (DMVR) process steps; and

performing final motion compensation; and

when the SAD is not less than the CU size-dependent threshold value:

performing the remaining DMVR process steps; and

performing the final motion compensation.

8. The method of claim 7, wherein the subset of the motion compensated interpolated samples are unrounded merge motion vectors between the two motion compensated reference patches.

9. The method of claim 7, wherein the bilinear motion compensated interpolation process is performed in parallel on the motion compensated reference patches.

10. The method of claim 7, wherein the motion compensated interpolated samples are derived from the two motion compensated reference patches in an interleaved manner.

11. The method of claim 7, wherein the subset of the motion compensated interpolated samples comprises a block of samples having a dimension equal to a dimension of a coding unit (CU).

12. The method of claim 11, wherein the subset of the motion compensated interpolated samples comprises (CU_width – 2) * (CU-height – 2) samples, wherein CU_width is a width of the coding unit and CU-height is a height of the coding unit.

13. The method of claim 11, wherein the subset of the motion compensated interpolated samples comprises (CU_width * CU_height – (CU_width – 2) * (CU-height – 2)) samples, wherein CU_width is a width of the coding unit and CU-height is a height of the coding unit.

14. The method of claim 7, wherein the CU size-dependent threshold value is a function of a number of samples in the subset of motion compensated interpolated samples.

15. The method of claim 7, wherein the bilinear motion compensated interpolation is performed in parallel on the two motion compensated reference patches.

16. The method of claim 7, wherein the bilinear motion compensated interpolation is performed in an interleaved manner on the two motion compensated reference patches.

17. A method of determining an alignment level between motion compensated reference patches, the method comprising:

obtaining, by a decoder, motion compensated interpolated samples based on sub-pixel accurate merge motion vectors;

computing, by the decoder, a mean-removed sum of absolute differences (MR-SAD) between motion compensated interpolated samples at center positions of respective first and second motion compensated reference patches;

determining, by the decoder, whether the MR-SAD is less than a coding unit (CU) size-dependent threshold value;

when the MR-SAD is less than the CU size-dependent threshold value:

skipping remaining decoder-side motion vector refinement (DMVR) process steps; and

performing final motion compensation using the unrounded merge motion vectors; and

when the MR-SAD is not less than the CU size-dependent threshold value:

performing the remaining DMVR process steps using the unrounded merge motion vectors; and

performing the final motion compensation using the unrounded merge motion vectors.

18. The method of claim 17, wherein computing the MR-SAD comprises:

computing a mean value of the first motion compensated reference patch;

computing a mean value of the second motion compensated reference patch;

determining a first motion compensated interpolated sample at the center of the first motion compensated reference patch

determining a second motion compensated interpolated sample at the center of the first motion compensated reference patch, wherein the MR-SAD is computed using the equation:

MR-SAD = SAD (first_sample – mean(first_patch), second_sample – mean (second_patch)), where first_sample represents the first motion compensated interpolated sample at the center of the first motion compensated reference patch, second_sample represents the second motion compensated interpolated sample at the center of the second motion compensated

reference patch, mean(first_block) represents an averaging operation of sample values of first_block, mean(second_block) represents an averaging operation of sample values of second_block, and SAD is a summing operation of absolute differences between (first_sample – mean(first_patch)) and (second_sample – mean (second_patch)).

19. The method of claim 17, further comprising:

adjusting the CU size-dependent threshold value in response to a bit depth of the motion compensated interpolated samples.

20. The method of claim 19, wherein adjusting the CU size-dependent threshold value comprises bitwise shifting a number of bits to the right.

21. An inter prediction method, comprising:

obtaining a first motion compensated reference patch and a second motion compensated reference patch based on initial motion vectors of Merge Mode;

computing a patch difference value between the first motion compensated reference patch and the second motion compensated reference patch;

obtaining a prediction value of a current block based on the initial motion vectors in the event that the patch difference value is less than a threshold, wherein the threshold is determined based on a size of the current block; and

refining the initial motion vectors to obtain the perdition value of the current block in the event that the patch difference value is greater than or equal to the threshold.

22. The method of claim 21, wherein the initial motion vectors are derived from motion vectors of a neighboring block of the current block.

23. The method of claim 21 or 22, wherein the initial motion vectors comprises a first motion vector and a second motion vector, wherein the first motion compensated reference patch is obtained according to the first motion vector, and the second motion compensated reference patch is obtained according to the second motion vector.

24. The method of any one of claims 21-23, wherein the patch difference value is a SAD value or a mean-reduced SAD value.

25. The method of any one of claims 21-24, wherein computing the patch difference value between the first motion compensated reference patch and the second motion compensated reference patch, comprising:

computing the patch difference value between a subset samples of the first motion compensated reference patch and a corresponding subset samples of the second motion compensated reference patch.

26. The method of any one of claims 21-25, wherein after computing the patch difference value between the first motion compensated reference patch and the second motion compensated reference patch, further comprising:

performing a comparison based on the patch difference value and the threshold.

27. The method of claim 26, wherein before performing the comparison based on the patch difference value and the threshold, further comprising:

adjusting the threshold after determining the threshold according to the size of the current block.

28. The method of any one of claims 21-27, wherein in the event that the patch difference value is less than the threshold, before obtaining the prediction value of the current block based on the initial motion vectors, further comprising:

determining decoder-side motion vector refinement (DMVR) is skipped.

29. The method of any one of claims 21-28, wherein in the event that the patch difference value is less than the threshold, obtaining the prediction value of the current block using motion vectors that are the same as the initial motion vectors.

30. An inter prediction apparatus, comprising:

a non-transitory memory having processor-executable instructions stored thereon; and

a processor, coupled to the memory, configured to execute the processor-executable instructions to facilitate the method according to any one of claims 21-29.

WHAT IS CLAIMED IS:

1. A method of motion compensation for a block of a frame in a video with a decoder-side motion vector refinement (DMVR) process comprising:

obtaining, by a decoder, motion compensated interpolated samples in two reference frames of the block based on sub-pixel accurate merge motion vectors of the block for the two reference frames via a bilinear motion compensated interpolation;

computing, by the decoder, a sum of absolute differences (SAD) between two motion compensated reference patches using a subset of the motion compensated interpolated samples;

determining, by the decoder, whether the SAD is less than a coding unit (CU) size-dependent threshold value, the CU threshold value depending on a size of the CU;

when the SAD is less than the CU size-dependent threshold value:

skipping remaining steps of the DMVR process for the block; and

performing final motion compensation using the sub-pixel accurate merge motion vectors of the block to obtain reconstructed samples for the block; and

when the SAD is not less than the CU size-dependent threshold value:

performing the remaining steps of the DMVR process to determine refined motion vectors for the block; and

performing the final motion compensation using the refined motion vectors to obtain the reconstructed samples for the block.

2. The method of claim 1, wherein the subset of the motion compensated interpolated samples are generated based on unrounded merge motion vectors associated with the two motion compensated reference patches.

3. The method of claim 1, wherein the bilinear motion compensated interpolation for generating the motion compensated interpolated samples in the two reference frames are performed in parallel .

4. The method of claim 1, wherein the motion compensated interpolated samples are generated in an interleaved manner.

5. The method of claim 1, wherein the subset of the motion compensated interpolated samples comprises a block of samples having a dimension equal to a dimension of a coding unit (CU).

6. The method of claim 5, wherein the subset of the motion compensated interpolated samples comprises (CU_width – 2) * (CU-height – 2) samples, wherein CU_width is a width of the coding unit and CU-height is a height of the coding unit.

7. The method of claim 5, wherein the subset of the motion compensated interpolated samples comprises (CU_width * CU_height – (CU_width – 2) * (CU-height – 2)) samples, wherein CU_width is a width of the coding unit and CU-height is a height of the coding unit.

8. The method of claim 1, wherein the CU size-dependent threshold value is a function of a number of samples in the subset of motion compensated interpolated samples.

9. A decoder, performing motion compensation for a block of a frame in a video with a decoder-side motion vector refinement (DMVR) process, the decoder comprising:

one or more processors; and

a non-transitory computer-readable storage medium coupled to the processors and storing programming for execution by the processors, the programming, when executed by the processors, configuring the decoder to:

obtain motion compensated interpolated samples in two reference frames of the block based on sub-pixel accurate merge motion vectors of the block for the two reference frames via a bilinear motion compensated interpolation;

compute a sum of absolute differences (SAD) between two motion compensated reference patches using a subset of the motion compensated interpolated samples;

determine whether the SAD is less than a coding unit (CU) size-dependent threshold value, the CU threshold value depending on a size of the CU;

when the SAD is less than the CU size-dependent threshold value:

skip remaining steps of the DMVR process for the block; and

perform final motion compensation using the sub-pixel accurate merge motion vectors of the block to obtain reconstructed samples for the block; and

when the SAD is not less than the CU size-dependent threshold value:

perform the remaining steps of the DMVR process to determine refined motion vectors for the block; and

perform the final motion compensation using the refined motion vectors to obtain the reconstructed samples for the block.

10. The decoder of claim 9, wherein the subset of the motion compensated interpolated samples are generated based on unrounded merge motion vectors associated with the two motion compensated reference patches.

11. The decoder of claim 9, wherein the bilinear motion compensated interpolation for generating the motion compensated interpolated samples in the two reference frames are performed in parallel.

12. The decoder of claim 9, wherein the motion compensated interpolated samples are generated in an interleaved manner.

13. The decoder of claim 9, wherein the subset of the motion compensated interpolated samples comprises a block of samples having a dimension equal to a dimension of a coding unit (CU).

14. The decoder of claim 13, wherein the subset of the motion compensated interpolated samples comprises (CU_width – 2) * (CU-height – 2) samples, wherein CU_width is a width of the coding unit and CU-height is a height of the coding unit.

15. The decoder of claim 13, wherein the subset of the motion compensated interpolated samples comprises (CU_width * CU_height – (CU_width – 2) * (CU-height – 2)) samples, wherein CU_width is a width of the coding unit and CU-height is a height of the coding unit.

16. The decoder of claim 9, wherein the CU size-dependent threshold value is a function of a number of samples in the subset of motion compensated interpolated samples.

17. A non-transitory computer-readable storage medium comprising instructions which, when the instructions are executed by a computer, cause the computer to carry out motion compensation for a block of a frame in a video with a decoder-side motion vector refinement (DMVR) process comprising:

obtaining motion compensated interpolated samples in two reference frames of the block based on sub-pixel accurate merge motion vectors of the block for the two reference frames via a bilinear motion compensated interpolation;

computing a sum of absolute differences (SAD) between two motion compensated reference patches using a subset of the motion compensated interpolated samples;

determining whether the SAD is less than a coding unit (CU) size-dependent threshold value, the CU threshold value depending on a size of the CU;

when the SAD is less than the CU size-dependent threshold value:

skipping remaining steps of the DMVR process for the block; and

performing final motion compensation using the sub-pixel accurate merge motion vectors of the block to obtain reconstructed samples for the block; and

when the SAD is not less than the CU size-dependent threshold value:

performing the remaining steps of the DMVR process to determine refined motion vectors for the block; and

performing the final motion compensation using the refined motion vectors to obtain the reconstructed samples for the block.

18. The non-transitory computer-readable storage medium of claim 17, wherein the subset of the motion compensated interpolated samples are generated based on unrounded merge motion vectors associated with the two motion compensated reference patches.

19. The non-transitory computer-readable storage medium of claim 17, wherein the bilinear motion compensated interpolation for generating the motion compensated interpolated samples in the two reference frames are performed in parallel.

20. The non-transitory computer-readable storage medium of claim 17, wherein the motion compensated interpolated samples are generated in an interleaved manner.