METHOD AND APPARATUS FOR CODING IMAGE OF VIDEO SEQUENCE, AND TERMINAL DEVICE

TECHNICAL FIELD

The present disclosure relates to the technical field of image coding, and in particular, to a method and an apparatus for coding an image of a video sequence, and a terminal device.

BACKGROUND

Video coding (video encoding and decoding) is used in a wide range of digital video applications, for example broadcast digital TV, video transmission over internet and mobile networks, real-time conversational applications such as video chat, video conferencing, DVD and Blu-ray discs, video content acquisition and editing systems, and camcorders of security applications.

In video compression, inter prediction is a process of using reconstructed samples of previously decoded reference pictures by specifying motion vectors relative to a current image block. These motion vectors can be coded as a prediction residual by using spatial or temporal motion vector predictors. The motion vectors can be at sub-pixel accuracy. In order to derive the sub-pixel accurate pixel values in the reference images from the reconstructed integer position values, an interpolation filter is applied. Bi-prediction refers to a process where the prediction for the current image block is derived as a weighted combination of two prediction blocks derived using two motion vectors from two reference picture areas. In this case, in addition to the motion vectors, the reference indices for the reference pictures from which the two prediction blocks are derived also need to be coded. The motion vectors for the current image block can also be derived through a merge process where a spatial neighbor's motion vectors and reference indices are inherited without coding any motion vector residuals. In addition to spatial neighbors, motion vectors of previously coded reference pictures are

also stored and used as temporal merge options with appropriate scaling of the motion vectors to take care of the distance to the reference images relative to the distance to the reference pictures for the current image block.

Several methods have been proposed for performing a decoder-side motion vector refinement (DMVR for short) or derivation (DMVD for short) so that the motion vector can be further refined. During both of the DMVR process and the DMVD process, a refinement search needs to be performed around starting points that can be sub-pixel accurate motion vectors, which is usually complex and time-consuming.

This background information is provided to reveal information believed by the applicant to be of possible relevance to the present disclosure. No admission is necessarily intended, nor should be construed, that any of the preceding information constitutes prior art against the present disclosure.

SUMMARY

In view of the above, in order to overcome the above problem, the present disclosure provides a method and an apparatus for coding an image of a video sequence, and a terminal device.

The foregoing and other objects are achieved by the subject matter of the independent claims. Further implementation forms are apparent from the dependent claims, the description and the figures.

According to a first aspect the present disclosure relates to a method for coding an image of a video sequence, where the image comprises a current image block. The method is performed by an apparatus for coding an image of a video sequence. In the method, initial motion information of the current image block is obtained, sample values of at least two candidate reference image blocks corresponding to the current image block are obtained according to the initial motion information, a bit-depth of sample values of each candidate reference image block is constrained to a predefined bit-depth, then refined motion information is obtained based on constrained sample values of at least two constrained

candidate reference image blocks, and a prediction image block of the current image block is determined according to the refined motion information.

With the method for coding an image of a video sequence provided in the present disclosure, the bit-depth of sample values of each candidate reference image block is constrained before the refining of the motion information, thus rendering the computing complexity lower and allowing for less processing time.

In a possible implementation form of the method according to the first aspect as such, the obtaining refined motion information based on constrained sample values of the at least two constrained candidate reference image blocks includes: using a second cost function to determine a position of a constrained candidate reference image block with the least cost; obtaining the refined motion information according to the position of the constrained candidate reference image block with the least cost. Where the second cost function indicates a difference between a constrained candidate reference image block and a template of the current image block, wherein the template is obtained based on the initial motion information and a position of the current image block and a bit-depth of sample values of the template is constrained to the predefined bit-depth. Here the template serves as a comparison reference obtained from a weighted combination of two constrained candidate reference image blocks. The cost is calculated as:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{I0}(i,j) - \boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I1}(i,j) + (\boldsymbol{m0} - \boldsymbol{m1}))))|, \text{ if } \boldsymbol{m0} > \boldsymbol{m1}$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I0}(i,j) + (\boldsymbol{m1} - \boldsymbol{m0}))) - \boldsymbol{I1}(i,j))|, \text{otherwise}$$

Where $\boldsymbol{H}$ is a height of the current image block, $\boldsymbol{W}$ is a width of the current image block, $\boldsymbol{I0(i, j)}$ is a sample value at a position (i, j) of a constrained candidate reference image block, $\boldsymbol{I1(i, j)}$ is a sample value at the position (i, j) of the template, $\boldsymbol{m0}$ is a mean of sample values of the constrained candidate reference image block, $\boldsymbol{m1}$ is a mean of sample values of the template, d is the predefined bit-depth.

The above equation aims to limit the variables therein do not exceed the maximum value that can be expressed with d bits. It also specifies a normative process through which the final absolute difference can be performed between two quantities at the predefined bit-depth, thereby avoiding possible overflowing errors.

In a possible implementation form of the method according to the first aspect as such, using a second cost function to determine a position of a constrained candidate reference image block with the least cost; obtaining the refined motion information according to the position of the constrained candidate reference image block with the least cost. Where the at least two constrained candidate reference image blocks comprises a set of first constrained candidate reference image blocks and a set of second constrained candidate reference image blocks, and the second cost function indicates a difference between a first constrained candidate reference image block and a second constrained candidate reference image block. The cost is calculated as:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{I0}(i,j) - \boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I1}(i,j) + (\boldsymbol{m0} - \boldsymbol{m1}))))|, \text{ if } \boldsymbol{m0} > \boldsymbol{m1}$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I0}(i,j) + (\boldsymbol{m1} - \boldsymbol{m0}))) - \boldsymbol{I1}(i,j))|, \text{otherwise}$$

Where $\boldsymbol{H}$ is a height of the current image block, $\boldsymbol{W}$ is a width of the current image block, $\boldsymbol{I0(i, j)}$ is a sample value at a position (i, j) of a first constrained candidate reference image block, $\boldsymbol{I1(i, j)}$ is a sample value at the position (i, j) of a second constrained candidate reference image block, $\boldsymbol{m0}$ is a mean of sample values of the first constrained candidate reference image block, $\boldsymbol{m1}$ is a mean of sample values of the second constrained candidate reference image block, d is the predefined bit-depth.

The above equation aims to limit the variables therein do not exceed the maximum value that can be expressed with d bits. It also specifies a normative process through which the final absolute difference can be performed between two quantities at the predefined bit-depth, thereby avoiding possible overflowing errors.

According to a second aspect the present disclosure relates to an apparatus for coding an image of a video sequence. The apparatus includes an obtaining module, a constraining module and a determining module, where the obtaining module is configured to obtain initial motion information of the current image block, as well as to obtain sample values of at least two candidate reference image blocks corresponding to the current image block according to the initial motion information, and also obtain refined motion information based on constrained sample values of the at least two constrained candidate reference image blocks. The constraining module is configured to constrain a bit-depth of sample values of each

candidate reference image block to a predefined bit-depth, and the determining module is configured to determine a prediction image block of the current image block according to the refined motion information.

With the apparatus for coding an image of a video sequence provided in the present disclosure, the bit-depth of sample values of each candidate reference image block is constrained before the refining of the motion information, thus rendering the computing complexity lower and allowing for less processing time.

In a possible implementation form of the apparatus according to the second aspect as such, the obtaining module is specifically configured to: use a second cost function to determine a position of a constrained candidate reference image block with the least cost; obtain the refined motion information according to the position of the constrained candidate reference image block with the least cost. Where the second cost function indicates a difference between a constrained candidate reference image block and a template of the current image block, wherein the template is obtained based on the initial motion information and a position of the current image block and a bit-depth of sample values of the template is constrained to the predefined bit-depth. Here the template serves as a comparison reference obtained from a weighted combination of two constrained candidate reference image blocks. The cost is calculated as:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{I0}(i,j) - \boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I1}(i,j) + (\boldsymbol{m0} - \boldsymbol{m1}))))|, \text{ if } \boldsymbol{m0} > \boldsymbol{m1}$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I0}(i,j) + (\boldsymbol{m1} - \boldsymbol{m0}))) - \boldsymbol{I1}(i,j))|, \text{otherwise}$$

Where $\boldsymbol{H}$ is a height of the current image block, $\boldsymbol{W}$ is a width of the current image block, $\boldsymbol{I0(i, j)}$ is a sample value at a position (i, j) of a constrained candidate reference image block, $\boldsymbol{I1(i, j)}$ is a sample value at the position (i, j) of the template, $\boldsymbol{m0}$ is a mean of sample values of the constrained candidate reference image block, $\boldsymbol{m1}$ is a mean of sample values of the template, d is the predefined bit-depth.

The above equation aims to limit the variables therein do not exceed the maximum value that can be expressed with d bits. The above equation specifies a normative process through which the final absolute difference can be performed between two quantities at the predefined bit-depth, thereby avoiding possible overflowing errors.

In a possible implementation form of the apparatus according to the second aspect as such, the obtaining module is specifically configured to: use a second cost function to determine a position of a constrained candidate reference image block with the least cost; obtain the refined motion information according to the position of the constrained candidate reference image block with the least cost. Where the at least two constrained candidate reference image blocks comprises a set of first constrained candidate reference image blocks and a set of second constrained candidate reference image blocks, and the second cost function indicates a difference between a first constrained candidate reference image block and a second constrained candidate reference image block. The cost is calculated as:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{I0}(i,j) - \boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I1}(i,j) + (\boldsymbol{m0} - \boldsymbol{m1}))))|, \text{ if } \boldsymbol{m0} > \boldsymbol{m1}$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I0}(i,j) + (\boldsymbol{m1} - \boldsymbol{m0}))) - \boldsymbol{I1}(i,j))|, \text{otherwise}$$

Where $\boldsymbol{H}$ is a height of the current image block, $\boldsymbol{W}$ is a width of the current image block, $\boldsymbol{I0(i, j)}$ is a sample value at a position (i, j) of a first constrained candidate reference image block, $\boldsymbol{I1(i, j)}$ is a sample value at the position (i, j) of a second constrained candidate reference image block, $\boldsymbol{m0}$ is a mean of sample values of the first constrained candidate reference image block, $\boldsymbol{m1}$ is a mean of sample values of the second constrained candidate reference image block, d is the predefined bit-depth.

The above equation specifies a normative process through which the final absolute difference can be performed between two quantities at the predefined bit-depth, thereby avoiding possible overflowing errors.

The method according to the first aspect of the present disclosure can be performed by the apparatus according to the second aspect of the present disclosure. Further features and implementation forms of the method according to the first aspect of the present disclosure result directly from the functionality of the apparatus according to the third aspect of the present disclosure and its different implementation forms.

According to a third aspect the present disclosure relates to an apparatus for coding an image of a video sequence which includes a processor and a memory. The memory is storing instructions that cause the processor to perform the method according to the first aspect.

According to a fourth aspect, a computer-readable storage medium having stored thereon instructions that when executed cause one or more processors configured to code an image of a video sequence is proposed. The instructions cause the one or more processors to perform a method according to the first aspect or any possible embodiment of the first aspect.

According to a fifth aspect the present disclosure relates to an encoder including an apparatus for coding an image of a video sequence according to the second or the third aspect.

According to a sixth aspect the present disclosure relates to an decoder including an apparatus for coding an image of a video sequence according to the second or the third aspect.

According to a seventh aspect the present disclosure relates to a terminal device including an encoder according to the fifth aspect and a decoder according to the sixth aspect.

With the method or apparatus for coding an image of a video sequence provided in the present disclosure, the bit-depth of sample values of each candidate reference image block is constrained before the refining of the motion information, thus rendering the computing complexity lower and allowing for less processing time.

BRIEF DESCRIPTION OF DRAWINGS

The accompanying drawings are used to provide a further understanding of the present disclosure, constitute a part of the specification, and are used to explain the present disclosure together with the following specific embodiments, but should not be construed as limiting the present disclosure. In the drawings,

FIG. 1 is a schematic view of template matching (TM) based decoder side motion vector derivation (DMVD);

FIG. 2 is a schematic view of a bilateral matching (BM) based DMVD;

FIG. 3 is a schematic view of bilateral template matching based decoder side motion vector refinement (DMVR);

FIG. 4 is a schematic flowchart of a method for coding an image of a video sequence according to an embodiment of the present disclosure;

FIG. 5 is a schematic flowchart of a method for coding an image of a video sequence according to an embodiment of the present disclosure; and

FIG. 6 is a structural view of an apparatus for coding an image of a video sequence according to an embodiment of the present disclosure.

DESCRIPTION OF EMBODIMENTS

In the following description, reference is made to the accompanying figures, which form part of the disclosure, and which show, by way of illustration, specific aspects of embodiments of the present disclosure or specific aspects in which embodiments of the present disclosure may be used. It is understood that embodiments of the present disclosure may be used in other aspects and comprise structural or logical changes not depicted in the figures. The following detailed description, therefore, is not to be taken in a limiting sense, and the scope of the present disclosure is defined by the appended claims.

For instance, it is understood that a disclosure in connection with a described method may also hold true for a corresponding device or system configured to perform the method and vice versa. For example, if one or a plurality of specific method steps are described, a corresponding device may include one or a plurality of units, e.g. functional units, to perform the described one or plurality of method steps (e.g. one unit performing the one or plurality of steps, or a plurality of units each performing one or more of the plurality of steps), even if such one or more units are not explicitly described or illustrated in the figures. On the other hand, for example, if a specific apparatus is described based on one or a plurality of units, e.g. functional units, a corresponding method may include one step to perform the functionality of the one or plurality of units (e.g. one step performing the functionality of the one or plurality of units, or a plurality of steps each performing the functionality of one or more of the plurality of units), even if such one or plurality of steps are not explicitly described or illustrated in the figures. Further, it is understood that the features of the various exemplary embodiments and/or aspects described herein may be combined with each other, unless specifically noted otherwise.

Video coding typically refers to the processing of a sequence of pictures, which form the video or video sequence. Instead of the term "picture" the term "frame" or "image" may be used as synonyms in the field of video coding. Video coding used in the present disclosure

(or present disclosure) indicates either video encoding or video decoding. Video encoding is performed at the source side, typically comprising processing (e.g. by compression) the original video pictures to reduce the amount of data required for representing the video pictures (for more efficient storage and/or transmission). Video decoding is performed at the destination side and typically comprises the inverse processing compared to the encoder to reconstruct the video pictures. Embodiments referring to "coding" of video pictures (or pictures in general, as will be explained later) shall be understood to relate to either "encoding" or "decoding" for video sequence. The combination of the encoding part and the decoding part is also referred to as CODEC (Encoding and Decoding).

In case of lossless video coding, the original video pictures can be reconstructed, i.e. the reconstructed video pictures have the same quality as the original video pictures (assuming no transmission loss or other data loss during storage or transmission). In case of lossy video coding, further compression, e.g. by quantization, is performed, to reduce the amount of data representing the video pictures, which cannot be completely reconstructed at the decoder, i.e. the quality of the reconstructed video pictures is lower or worse compared to the quality of the original video pictures.

Several video coding standards since H.261 belong to the group of "lossy hybrid video codecs" (i.e. combine spatial and temporal prediction in the sample domain and 2D transform coding for applying quantization in the transform domain). Each picture of a video sequence is typically partitioned into a set of non-overlapping blocks and the coding is typically performed on a block level. In other words, at the encoder the video is typically processed, i.e. encoded, on a block (video block) level, e.g. by using spatial (intra picture) prediction and temporal (inter picture) prediction to generate a prediction block, subtracting the prediction block from the current block (block currently processed/to be processed) to obtain a residual block, transforming the residual block and quantizing the residual block in the transform domain to reduce the amount of data to be transmitted (compression), whereas at the decoder the inverse processing compared to the encoder is partially applied to the encoded or compressed block to reconstruct the current block for representation. Furthermore, the encoder duplicates the decoder processing loop such that both will generate identical predictions (e.g. intra- and inter predictions) and/or re-constructions for processing, i.e.

coding, the subsequent blocks.

As used herein, the term "block" may a portion of a picture or a frame. For convenience of description, embodiments of the present disclosure are described herein in reference to High-Efficiency Video Coding (HEVC) or the reference software of Versatile video coding (VVC), developed by the Joint Collaboration Team on Video Coding (JCT-VC) of ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Motion Picture Experts Group (MPEG). One of ordinary skill in the art will understand that embodiments of the present disclosure are not limited to HEVC or VVC. It may refer to a CU, PU, and TU. In HEVC, a CTU is split into CUs by using a quad-tree structure denoted as coding tree. The decision whether to code a picture area using inter-picture (temporal) or intra-picture (spatial) prediction is made at the CU level. Each CU can be further split into one, two or four PUs according to the PU splitting type. Inside one PU, the same prediction process is applied and the relevant information is transmitted to the decoder on a PU basis. After obtaining the residual block by applying the prediction process based on the PU splitting type, a CU can be partitioned into transform units (TUs) according to another quadtree structure similar to the coding tree for the CU. In the newest development of the video compression technical, Qual-tree and binary tree (QTBT) partitioning frame is used to partition a coding block. In the QTBT block structure, a CU can have either a square or rectangular shape. For example, a coding tree unit (CTU) is first partitioned by a quadtree structure. The quadtree leaf nodes are further partitioned by a binary tree structure. The binary tree leaf nodes are called coding units (CUs), and that segmentation is used for prediction and transform processing without any further partitioning. This means that the CU, PU and TU have the same block size in the QTBT coding block structure. In parallel, multiply partition, for example, triple tree partition was also proposed to be used together with the QTBT block structure.

For the sake of clarity, several terms involved in the present disclosure will be explained in the first place.

The term "coding" in the present disclosure may refer to either encoding or decoding, which is not limited in any one of the embodiments of the present disclosure unless otherwise specified.

The term "current image block" in the present disclosure represents an image block

currently under processing, for example, during the encoding process, the current image block may be the image block currently being encoded, while during the decoding process, the current image block may be the image block currently being decoded.

The term "reference image block" in the present disclosure represents an image block which provides reference information for the current image block, during a searching process, a best matching reference image block needs to be found among multiple reference image blocks through traversal operations.

The term "current image" in the present disclosure refers to an image where the current image block is located. Correspondingly, the term "reference image" in the present disclosure refers to an image where the reference image block is located.

The term "prediction image block" in the present disclosure represents an image block which provides predictions to the current image block. For example, a best matching reference image block may be found among the multiple reference image blocks, and this best matching reference image block may serve as the prediction image block of the current image block.

The term "motion vector (MV)" in the present disclosure represents a position offset of a reference image block relative to a current image block.

The "motion information" of an image block in the present disclosure represents information about the movement of the image block, in some embodiments, the motion information may include two kinds of information such as indication of a reference image and a motion vector of a reference image block in said reference image; in some embodiments, the motion information may only include the motion vector of said reference image block. The contents of the motion information are not limited in the present disclosure and may change along with specific application scenarios.

The term "sample value" of an image block refers to the pixel value at a sample point of the image block.

A terminal device may be any one of the following devices: a smartphone, a mobile phone, a cellular phone, a cordless phone, a session initiation protocol (SIP) phone, a wireless local loop (WLL) station, a personal digital assistant (PDA), a handheld device capable of wireless communication, an on-board equipment, a wearable device, a computing device or

other processing devices connecting to a wireless modem.

As can be seen from the above description, the motion information needs to be encoded and transmitted to the decoder side. There are two main techniques for MV encoding, including the merge technique and the advanced motion vector prediction (AMVP). The principles of these two techniques are as same as those in prior art, thus will not be illustrated in detail herein.

Further, several methods have been proposed for performing a DMVD process or a DMVR process so that the motion vector residual coding bits can be further reduced.

One class of methods, called template matching (TM) methods, using an L-shaped region adjacent the current block (as shown in FIG. 1) that has already been reconstructed, called as the template, and identifying a best matching L-shaped region, using cost functions such as sum of absolute differences (SAD) or mean-removed sum of absolute differences (MR-SAD), in each reference image (denoted as Ref0 in FIG. 1) using a plurality of suitably scaled spatial and temporal motion vector candidates. Then, centered on the best matching candidate, further refinement is performed within a certain refinement distance around that center. On the encoder side, rate distortion optimized cost is computed to decide between uni-prediction (i.e. prediction using the best matching reference) and bi-prediction (i.e. prediction derived by averaging the top two best matching references).

Another class of methods, called BM methods. As shown in FIG. 2, the motion information of a current CU is derived by finding the closest match between two blocks along the motion trajectory of the current CU in two different reference images (denoted as Ref0 and Ref1 in FIG. 2). Under the assumption of continuous motion trajectory, the motion vectors MV0 and MV1 respectively pointing to the two reference blocks Ref0 and Ref1 shall be proportional to the temporal distances, i.e., TD0 and TD1, between the current image and the two reference images. When the current image is temporally between the two reference images and the temporal distance from the current image to the two reference images is the same, the bilateral matching becomes mirror based bi-directional MV.

In the BM merge mode, bi-prediction is always applied since the motion information of a CU is derived based on the closest match between two blocks along the motion trajectory of the current CU in two different reference images.

In some embodiments, the temporal distances are ignored and bilateral matching is performed with equal and opposite motion vectors in the past and future reference images respectively.

In a variant of bilateral matching mode, that is, the aforementioned DMVR method, as shown in FIG. 3, a bilaterally averaged template is first created using the prediction blocks referred by the initial MV0 and MV1 (denoted as Step 1 in FIG. 3) in the first candidate reference image block list L0 and the second initial candidate reference image block list L1 obtained from explicitly signaled merge index and BM is performed against this template to find the best matched blocks referred by the updated MV0' and MV1' (denoted as Step 2 in FIG. 3). The template is updated if there is any movement. Also, in some embodiments, the refinement is performed in one reference and the motion vector in the other reference is obtained through mirroring of this refined motion vector. The refinement alternates between the two references till either the center position has the least error or the maximum number of iterations is reached.

In some of the methods of refinement, a CU level refinement is first performed. Then a sub-CU level multi-candidate evaluation is performed along with the CU-level refined MVs as candidates. Optionally, each sub-CU can perform its own refinement with respect to the best matching candidate.

In a default merge mode, no DMVD is required, while in an explicit merge mode, the merge mode (whether TM merge or BM merge) may be signaled. In some embodiments, no merge index is signaled while in other embodiments, to simplify the decoder complexity of performing multiple motion compensations, an explicit merge index is signaled.

Given the implicit DMVD or DMVR process, the encoder needs to perform these steps in exactly the same manner as the decoder in order for the encoder-side reconstruction to match with the decoder-side reconstruction.

In fact, the DMVD requires no merge index while the DMVR requires a merge index to obtain the initial motion information of the current image block.

Whenever an explicit merge index is signaled, the DMVR starts from the motion vectors and reference indices normatively deduced from the signaled merge index. When an explicit merge-mode index is not signaled, that is the DMVD case, a set of initial motion vector

candidates are evaluated at the decoder using a cost function and the candidate with the lowest cost is chosen as the starting point for refinement. Thus, irrespective of whether the DMVD method is based on TM or based on BM through difference minimization between equal and oppositely displaced blocks in a first candidate reference image block list L0 and a second candidate reference image block list L1 (commonly referred to as BM cost) or based on the difference between an averaged version of the corresponding patches in the first candidate reference image block list L0 and the second candidate reference image block list L1 and a displacement in L0/L1 (commonly referred to as DMVR cost), there is a refinement search that needs to be performed around the starting point that may be sub-pixel accurate motion vectors.

In order to evaluate the cost function, sometimes an interpolation needs to be performed to derive the sample values at the sub-pixel accurate centers based on the sample values of the reference images at the integer positions. The interpolating filter can be as simple as a bilinear interpolation filter or can be a longer filter such as the 2-D DCT-based separable interpolation filters. In order to reduce the complexity of deriving the candidate reference image block for a current image block again and again at each position considered during the refinement, an integer-pixel distance grid of refinement points centered at the sub-pixel accurate position(s) in the first candidate reference image block list L0 and/or the second candidate reference image block list L1 had been proposed in another invention. With this, only incremental interpolations need to be performed as a new position close to a current best cost position is considered. After the refinement of integer-pixel distance grid is completed, best integer delta MVs are obtained with respect to the merge MVs.

In order to improve the compression gains further, sub-pixel distance refinement can be performed. A half pixel distance refinement requires the candidate reference image blocks at half pixel distance from the best integer distance MV position(s) in the reference image(s). It is possible to perform the sub-pixel accurate refinement using BM cost or DMVR cost. As we go to quarter-pixel distance and lower sub-pixel accuracies, the interpolation filter required at each position is different as the sub-pixel position phases do not align. In another application, using the costs evaluated at integer distance positions centered on the position with the least cost, a parametric error surface is fitted and the sub-pixel accurate

displacements with the least cost is computed.

However, there may still be complexity and time-consuming issues with the above process. It is known that the bit depths of the reference image typically range from 8 bits to 12 bits. Thus, during the integer distance refinement iterations described above and the evaluations of cost to compute the parameters of an error surface, the candidate reference image blocks of L0 and L1 are maintained in buffers. The total storage requirement for these buffers will depend on the bit depths of the candidate reference image blocks. In software, for instance, a 10-bit value occupies 16-bits of storage to avoid unnecessary packing (into supported data types) and corresponding unpacking operations. Also, the cost calculations for refinement are typically performed using SAD or a variant of it such as MR-SAD where the sum of absolute differences are computed after adjusting for the mean difference between the two blocks (at a sample level). The complexity of the cost calculation and the vector operation (single instruction multiple data (SIMD)) throughput will depend on the bit depths of the candidate reference image blocks. For instance, in software, 8-bit SADs can be performed in nearly half as many cycles as it takes to perform 9-16-bit SADs. Also, many smartphone chipsets contain hardware engines for encoding as well as decoding. As part of encoder hardware engines, they tend to have 8-bit SAD calculation building blocks.

Hence, aiming the above problem, the present disclosure provides a method for coding an image of a video sequence, where the bit-depths of the candidate reference image blocks are constrained, thereby allowing for a lower buffer size requirement as well as a lower gate count in hardware or a higher throughput for the cost evaluations, without impacting the bit-rate savings coming from decoder side motion vector refinement. The present disclosure provides a normative method (that is employed at both of the encoder side and decoder side) which constrains the bit depth of the candidate reference image blocks that are used in the cost function evaluations to a predefined bit-depth.

Embodiments of the present disclosure will be elaborated in detail as follows.

FIG. 4 is a schematic flowchart of a method for coding an image of a video sequence according to an embodiment of the present disclosure, where the image includes a current image block. This method can be implemented by an apparatus for an image of a video sequence at an encoder side or a decoder side. Besides, the method may be combined with

the DMVD or the DMVR process, or any other processes for refining the motion information of a current image block, which is not limited herein.

The method includes the following steps:

S401: obtaining initial motion information of the current image block.

First of all, the initial motion information of the current image block may be obtained so as to determine the candidate reference image blocks corresponding to the current image block.

In a DMVD process, as mentioned above, since there is no merge index, therefore, a merge candidate list may be constructed based on motion information of adjacent blocks of the current image block. Then, a first cost function may be used to determine motion information with the least cost from the merge candidate list as the initial motion information. Here the merge candidate list may be constructed using existing methods, which will not be illustrated in detail herein. The first cost function may be one of the following: a SAD, a MR-SAD or a sum of squared differences (SSD).

In a DMVR process, as mentioned above, a merge index indicating the best motion information in the merge candidate list is available. Therefore, in this case, motion information identified by the merge index may be determined from the merge candidate list as the initial motion information, where the merge candidate list may be constructed using existing methods, as same as in the DMVD process.

S402: obtaining sample values of at least two candidate reference image blocks corresponding to the current image block according to the initial motion information.

After obtaining the initial motion information of the current image block, at least two candidate reference image blocks may be obtained according to the initial motion information, where the candidate reference image blocks serve as candidates for selection.

In fact, before obtaining sample values of at least two candidate reference image blocks corresponding to the current image block, the positions of the at least two candidate reference image blocks should be determined in the first place. As mentioned above, the candidate reference image blocks are located in a reference image, so the reference image needs to be identified at first.

In one possible implementation, the initial motion information includes an initial motion vector of the current image block and indication of a reference image, so that the reference image may be determined based on the indication, in some embodiments, the indication may be the index of the reference image. In another possible implementation, the initial motion information includes an initial motion vector of the current image block, the reference image may be a predefined image or based on a predefined algorithm, that is, an algorithm known to both of the encoder side and the decoder side to determine the reference image, for example, taking the image located most adjacent to the current image as the reference image.

In an embodiment, positions of the at least two candidate reference image blocks may be determined according to the initial motion information and a position of the current image block. For example, an initial candidate reference image block may be determined based on the initial motion vector and the position of the current image block as a starting point. Then, based on a search step, which may be predefined according to actual requirements and may be sub-pixel accurate, the positions of the candidate reference image blocks may be determined. For example, if the search step equals to 1, then the candidate reference image blocks may be obtained by moving the initial candidate reference image block towards different directions by one pixel distance. The at least two candidate reference image blocks may be obtained through iteratively moving in accordance with the search step. Also, a search range may be predefined, therefore, the reference image blocks within the predefined search range may be determined as said candidate reference image blocks.

Then, after determining the position of each candidate reference image block, the sample values of these candidate reference image blocks may be further determined based thereon. It is possible that only the sample values at the integer-pixel positions of the reference image are known. Regarding each candidate reference image block, if the sample values of the candidate reference image block are available, which means that the sample values of this candidate reference image block can be obtained from the sample values of the reference image, this may be a possible case when sample points of said candidate reference image block are located on the integer-pixel positions of the reference image, so the sample values of the candidate reference image block may be obtained simply by copying the sample values of the reference image. Otherwise, i.e., if the sample values of the candidate reference

image block are unavailable, in order to obtain the sample values of the candidate reference image block, an interpolation operation may be performed to determine these unavailable sample values. The interpolation operation may be done using any existing methods and will not be described in detail herein. For example, given the position of the current image block, a bilinear interpolation may be performed to obtain the sample values at the integer distance positions within the predefined search range.

S403: constraining a bit-depth of sample values of each candidate reference image block to a predefined bit-depth.

As described above, since the bit-depths of the candidate reference image blocks may affect the overall processing speed and complexity, therefore, before performing the refinement of the motion information, it would be better to constrain the bit-depth of sample values of each candidate reference image block.

It should be noted that, for each candidate reference image block, the bit-depths of the sample values are the same, and the bit-depth of the reference image, i.e., where the candidate reference image blocks are located, may also be the bit-depth of the sample values of the each candidate reference image block. Therefore, the constraining of the bit-depth of sample values of each candidate reference image block to the predefined bit-depth means to constrain the bit-depth of each sample value of each candidate reference image block to the predefined bit-depth.

In an embodiment, the constraining may be realized by performing a right shift by an amount of (b-d) on the bit-depth of sample values of each candidate reference image block, where b is the bit-depth of the sample values of the candidate reference image block, d is the predefined bit-depth, and b is greater than d. Those skilled in the art will know that the value of d can be chosen according to actual requirements.

Suppose the interpolation typically performs the following operation to derive the sample value after a Q-format adjustment arising from an interpolation filter weights,

$$\boldsymbol{L} = (\hat{\boldsymbol{L}} + (\mathbf{1} \ll (\boldsymbol{Q} - \mathbf{1})) \gg \boldsymbol{Q} \quad (1)$$

Where $\hat{L}$ is a non-normalized sample value of the candidate reference image block, L is a normalized sample value of the candidate reference image block, Q is a normalization factor of the interpolation filter.

Let b be the bit-depth of the sample values of the first candidate reference image block and d be the predefined bit-depth (less than b), then the bit-depth constrained sample value is derived as follows:

$$L' = L \gg (b - d) \quad (2)$$

Where L' is a constrained sample value of the candidate reference image block. Any right shift required by the interpolation process (from an increased internal bit-depth required during the 2-D interpolation) can be combined with this right shift.

Then the process of bit-depth constraining can be combined as,

$$L' = \left(\hat{L} + \left(1 \ll (Q - 1)\right)\right) \gg (Q + b - d) \quad (3)$$

The rest of the steps of the normative motion vector refinement process may be performed using these constrained bit-depth samples to obtain the refined motion vectors using which the normative motion compensated prediction and coding of the prediction residuals are performed.

For example, if b is 10 bits and d is 8 bits, then after constraining the bit-depth to 8-bits, the buffer size for 10-bit data reduces by 2x when 10-bit is stored in 16-bit data-types. Even otherwise, the buffer size requirement comes down by 20%. By using byte-wise SIMD operations on software platforms, the cycle count for evaluating cost functions can be halved. On hardware platforms, the cost evaluation operation can use 20% less bit planes, which results in reduced gate count and area.

S404: obtaining refined motion information based on constrained sample values of the at least two constrained candidate reference image blocks.

After the constraining operation in S403, the refined motion information may be obtained based on the constrained sample values of the at least two constrained candidate reference image blocks, here the term “constrained candidate reference image block” refers to the candidate reference image block with constrained sample values.

In an embodiment, a second cost function may be used to determine a position of a constrained candidate reference image block with the least cost, and then the refined motion information may be obtained according to the position of the constrained candidate reference image block. The second cost function is one of a SAD, a MR-SAD or a SSD.

In a TM based DMVD process, the second cost function may indicate a difference between a template of a constrained candidate reference image block (the L-shaped region of Ref0 shown in FIG. 1) and a template of the current image block (the L-shaped region of the current image shown in FIG. 1). Here the template actually serves as comparison references.

In a BM based DMVD process, there is no template, the second cost function may indicate a difference between two candidate reference image blocks.

Regarding the DMVR case, detailed description may be made in the following part.

As described in S402, in the case where the initial motion information includes both of the initial MV and the indication of the reference image, the refined motion information may also include a refined MV and the indication of the reference image.

S405: determining a prediction image block of the current image block according to the refined motion information.

As long as the refined motion information is obtained, with a determined reference image, the prediction image block of the current image block may be determined.

In an embodiment, the initial motion information includes an initial motion vector of the current image block, in this case, the refined motion information also include a refined motion vector of the current image block. As already described in S402, the reference image may be a reference image predefined or obtained based on a predefined algorithm, therefore, the prediction image block of the current image block may be determined according to the refined motion vector and the reference image.

In another embodiment, the initial motion information includes an initial motion vector and indication of a reference image, in this case, the refined motion information also include a refined motion vector of the current image block and the indication of the reference image, thus the prediction image block of the current image block may be determined according to the refined motion vector and the indication of the reference image.

It should be noted that although the terms herein are expressed herein using singular forms, those skilled in the art would know, without any creative effort, that if the method is combined with a refining process where several reference images are used for prediction, then the corresponding terms may represent a plurality of elements, for example, if the method is combined with a BM based refining process, the initial vector may include two initial MVs, there may be two reference images.

The present disclosure provides a method for coding an image of a video sequence, where initial motion information of the current image block is obtained, sample values of at least two candidate reference image blocks corresponding to the current image block are obtained according to the initial motion information, a bit-depth of sample values of each candidate reference image block is constrained to a predefined bit-depth, then refined motion information is obtained based on sample values of the at least two constrained candidate reference image blocks, and a prediction image block of the current image block is determined according to the refined motion information. Since the bit-depth of sample values of each candidate reference image block is constrained before the refining of the motion information, thus rendering the computing complexity lower and allowing for less processing time.

In order to elaborate the method of the present disclosure more clearly, in the following, taking the BM based DMVR process as an example, the method will be described more detailedly in combination with the BM based DMVR process.

FIG. 5 is a schematic flowchart of a method for coding an image of a video sequence according to an embodiment of the present disclosure. This method includes the following steps:

S501: obtaining initial motion information of the current image block.

Reference may be made to the description in S401.

Further, in this embodiment, the initial motion information includes first initial motion vector and second initial motion vector, and indications of two reference image. The two reference images include a first reference image corresponding to the first initial motion vector and a second reference image corresponding to the second initial motion vector.

S502: obtaining a first initial reference image block and a second initial reference image block based on the initial motion information and a position of the current image block.

For example, the first and second reference images may be obtained based on the indications, which may be respective indexes of the two reference images. Then, the first initial reference image block may be obtained according to the first initial motion vector and the position of the current image block, in a similar way, the second initial reference image block may be obtained according to the second initial motion vector and the position of the current image block.

S503: obtaining a template for the current image block.

For example, the template may be a weighted combination of the first initial reference image block and the second initial reference image block. Here the weights may be chosen according to actual requirements, which will not be limited herein.

This step is optional, since the later cost evaluation with a cost function may be calculated directed based on reference image blocks rather than using the template.

S504: obtaining sample values of a set of first candidate reference image blocks and sample values of a set of second candidate reference image blocks corresponding to the current image block.

The first candidate reference image block may be located in a reference image in a list L0, and may also be referred to as a forward candidate reference image block. The second candidate reference image block be located in a reference image in a list L1, and may also be referred to as a backward candidate reference image block.

For both of the first candidate reference image blocks and the second candidate reference image blocks, reference may be made to the description in S402 to obtain sample values thereof.

An example will be briefly described to shown how to implement the interpolation.

Provided that the position of the current image block is (x, y), with a dimension of W*H, where W is the width of the current image block and H is the height of the current image block. The first reference image is denoted as L0 and the second reference image is denoted as L1. The first initial motion vector and the second initial motion vector are respectively denoted as MV0 and MV1, where MV0 and MV1 are sub-pixel accurate motion vectors.

First, sub-pixel offsets FMV0 and FMV1, and integer parts IMV0 and IMV1 may be derived from the first and the second initial motion vector. FMV0 and FMV1 are respective fractional parts of the first and second initial motion vectors, while IMV0 and IMV1 are respective integer parts of the first and second initial motion vectors.

Then for each integer distance pixel position required in the search range in the two reference images L0 and L1, the interpolation operation may be performed using the sub-pixel phases.

S505: constraining a bit-depth of sample values of each first candidate reference image block to a predefined bit-depth and constraining a bit-depth of sample values of each second candidate reference image block to a predefined bit-depth.

For example, in this embodiment, the bit-depth of each first candidate reference image block may be constrained to a desired bit-depth by performing a right shift (with or without rounding) by the amount equal to the difference between the bit-depth of the first reference image and a desired bit-depth. Similar operations may be performed for constraining the bit-depth of each second candidate reference image block.

Notably, the sample values of the first and second candidate reference image blocks may be constrained simultaneously or in an arbitrary order, which is not limited herein.

S506: using a second cost function to determine a first position of a first constrained candidate reference image block with the least cost.

For example, the second cost function may be used to determine the first position of the first refined motion vector, where the second cost function indicates a difference between a first constrained candidate reference image block and a template of the current image block, where a bit-depth of sample values of the template is also constrained to the predefined bit-depth.

Therefore, the differences between the template and each first constrained candidate reference image block may be calculated and the first position of the first constrained reference image block with the smallest difference may be determined.

Here the cost may be calculated as follows:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(I0(i,j) - MIN((1 \ll d) - 1, (I1(i,j) + (m0 - m1))))|, \text{ if } m0 > m1$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(MIN((1 \ll d) - 1, (I0(i,j) + (m1 - m0))) - I1(i,j))|, \text{otherwise} \quad (4)$$

Where H is a height of the current image block, W is a width of the current image block, $I0(i, j)$ is a sample value at a position (i, j) of a first constrained candidate reference image block, $I1(i, j)$ is a sample value at the position (i, j) of the template, $m0$ is a mean of sample values of the first constrained candidate reference image block, $m1$ is a mean of sample values of the template, d is the predefined bit-depth.

The above equation aims to limit the variables therein do not exceed the maximum value that can be expressed with d bits. The above equation specifies a normative process through which the final absolute difference can be performed between two quantities at the predefined bit-depth, thereby avoiding possible overflowing errors.

S507: using a second cost function to determine a second position of a second constrained candidate reference image block

Similarly, the differences between the template and each second constrained candidate reference image block may be calculated and a second constrained reference image block with the smallest difference may be selected from the set of second constrained candidate reference image blocks. Here the cost may also be calculated in a similar way as in S506.

Notably, S506 and S507 may be executed simultaneously or in an arbitrary order, which is not limited herein.

S508: determining a prediction image block of the current image block according to the refined motion information.

For example, the refined motion information may be obtained according to a first position of the first constrained reference image block and a second position of the second constrained reference image block.

Besides, as described above, S503 may be an optional step. In fact, in the above example, the cost is calculated based on a template, which is actually the aforementioned DMVR cost.

It is also possible to evaluate the BM cost instead of the DMVR cost. In this case, S503 will be omitted. And in S507 and S508, since the second cost function indicates a difference

between a first constrained candidate reference image block and a second constrained candidate reference image block, therefore, the cost may be calculated as:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{I0}(i,j) - \boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I1}(i,j) + (\boldsymbol{m0} - \boldsymbol{m1}))))|, \text{ if } \boldsymbol{m0} > \boldsymbol{m1}$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I0}(i,j) + (\boldsymbol{m1} - \boldsymbol{m0}))) - \boldsymbol{I1}(i,j))|, \text{ otherwise} \quad (5)$$

Where $\boldsymbol{H}$ is a height of the current image block, $\boldsymbol{W}$ is a width of the current image block, $\boldsymbol{I0(i, j)}$ is a sample value at a position (i, j) of a first constrained candidate reference image block, $\boldsymbol{I1(i, j)}$ is a sample value at the position (i, j) of a second constrained candidate reference image block corresponding to the first constrained candidate reference image block, $\boldsymbol{m0}$ is a mean of sample values of the first constrained candidate reference image block, $\boldsymbol{m1}$ is a mean of sample values of the second constrained candidate reference image block, b is the bit-depth of the sample values of the first and second candidate reference image block, d is the predefined bit-depth and b is greater than d. Those skilled in the art will know that the value of d can be chosen according to actual requirements.

The above equation aims to limit the variables therein do not exceed the maximum value that can be expressed with d bits. It can be seen that the above equation actually aims to limit the value of the $(\boldsymbol{I0}(i,j) - \boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I1}(i,j) + (\boldsymbol{m0} - \boldsymbol{m1})$ in case of $\boldsymbol{m0} > \boldsymbol{m1}$, or otherwise $(\boldsymbol{m1} - \boldsymbol{m0}))) - \boldsymbol{I1}(i,j)$. In fact, this serves to prevent its value from overflowing when being stored in a buffer.

The above equation specifies a normative process through which the final absolute difference can be performed between two quantities at the predefined bit-depth, thereby avoiding possible overflowing errors.

Even after constraining the bit-depth of each candidate reference image block to the predefined bit-depth d as in the embodiment shown in FIG. 4, the value on which absolute is performed can exceed d when MR-SAD is employed. This will require 1-bit extra in hardware implementations, but can have a 2x throughput impact in software SIMD (e.g. 9-bit absolute difference will have half the throughput as with 8-bit absolute difference). Also, given that the entire process is normative, the bit-depth in MR-SAD case also needs to be clearly specified to avoid drift between encoder and decoder implementations.

The present disclosure provides a method for coding an image of a video sequence, which is combined with a BM based DMVR process. Where initial motion information of the current image block is obtained, sample values of at least two candidate reference image blocks corresponding to the current image block are obtained according to the initial motion information, a bit-depth of sample values of each candidate reference image block is constrained to a predefined bit-depth, then refined motion information is obtained based on sample values of the at least two constrained candidate reference image blocks, and a prediction image block of the current image block is determined according to the refined motion information. Since the bit-depth of sample values of each candidate reference image block is constrained before the refining of the motion information, thus rendering the computing complexity lower and allowing for less processing time.

FIG. 6 is a structural view of an apparatus for coding an image of a video sequence according to an embodiment of the present disclosure. Where the image comprises a current image block, the apparatus includes: an obtaining module 601, a constraining module 602 and a determining module 603.

The obtaining module 601 is configured to obtain initial motion information of the current image block.

The obtaining module 601 is further configured to obtain sample values of at least two candidate reference image blocks corresponding to the current image block according to the initial motion information;

The constraining module 602 is configured to constrain a bit-depth of sample values of each candidate reference image block to a predefined bit-depth;

The obtaining module 601 is further configured to obtain refined motion information based on constrained sample values of the at least two constrained candidate reference image blocks; and

The determining module 603 is configured to determine a prediction image block of the current image block according to the refined motion information.

In an embodiment, the constraining module 602 is specifically configured to: perform a right shift by an amount of (b-d) on the bit-depth of sample values of each candidate reference

image block, where b is the bit-depth of the sample values of the each candidate reference image block, d is the predefined bit-depth, and b is greater than d.

In an embodiment, the obtaining module 601 is specifically configured to: use a first cost function to determine motion information with the least cost from a merge candidate list as the initial motion information, where the merge candidate list includes motion information of adjacent blocks of the current image block.

In an embodiment, the obtaining module 601 is specifically configured to: determine motion information identified by a merge index from a merge candidate list as the initial motion information, where the merge candidate list includes motion information of adjacent blocks of the current image block.

In an embodiment, the obtaining module 601 is specifically configured to: determine a position of a candidate reference image block according to the initial motion information and a position of the current image block; determine whether sample values of the candidate reference image block are available based on the position of the candidate reference image block, if sample values of the candidate reference image block are unavailable, performing an interpolation operation to obtain the sample values of the candidate reference image block.

In an embodiment, the obtaining module 601 is specifically configured to: use a second cost function to determine a position of a constrained candidate reference image block with the least cost; obtain the refined motion information according to the position of the constrained candidate reference image block with the least cost.

In an embodiment, the second cost function indicates a difference between a constrained candidate reference image block and a template of the current image block, where the template is obtained based on the initial motion information and a position of the current image block and a bit-depth of sample values of the template is constrained to the predefined bit-depth.

In an embodiment, the cost is calculated as:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{I0}(i,j) - \boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I1}(i,j) + (\boldsymbol{m0} - \boldsymbol{m1}))))|, \text{ if } \boldsymbol{m0} > \boldsymbol{m1}$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I0}(i,j) + (\boldsymbol{m1} - \boldsymbol{m0}))) - \boldsymbol{I1}(i,j))|, \text{otherwise} \quad (6)$$

Where $\boldsymbol{H}$ is a height of the current image block, $\boldsymbol{W}$ is a width of the current image block, $\boldsymbol{I0(i, j)}$ is a sample value at a position (i, j) of a constrained candidate reference image block, $\boldsymbol{I1(i, j)}$ is a sample value at the position (i, j) of the template, $\boldsymbol{m0}$ is a mean of sample values of the constrained candidate reference image block, $\boldsymbol{m1}$ is a mean of sample values of the template, d is the predefined bit-depth.

In an embodiment, the at least two constrained candidate reference image blocks comprises a set of first constrained candidate reference image blocks and a set of second constrained candidate reference image blocks; where the obtaining module 601 is specifically configured to: obtain a first initial reference image block based on the first initial motion information and a position of the current image block; obtain a second initial reference image block based on the second initial motion information and the position of the current image block; calculate a weighted combination of the first initial reference image block and the second initial reference image block as the template; calculate differences between the template and each first constrained candidate reference image block and determine a position of a first constrained reference image block with the smallest difference; calculate differences between the template and each second constrained candidate reference image block and determine a position of a second constrained reference image block with the smallest difference. Where the determining module 603 is specifically configured to: obtain the refined motion information according to the first position and the second position.

In an embodiment, the at least two constrained candidate reference image blocks comprises a set of first constrained candidate reference image blocks and a set of second constrained candidate reference image blocks, and the second cost function indicates a difference between a first constrained candidate reference image block and a second constrained candidate reference image block.

In an embodiment, the cost is calculated as:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{I0}(i,j) - \boldsymbol{MIN}((\boldsymbol{1} \ll \boldsymbol{d}) - \boldsymbol{1}, (\boldsymbol{I1}(i,j) + (\boldsymbol{m0} - \boldsymbol{m1}))))|, \text{ if } \boldsymbol{m0} > \boldsymbol{m1}$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{MIN}((\boldsymbol{1} \ll \boldsymbol{d}) - \boldsymbol{1}, (\boldsymbol{I0}(i,j) + (\boldsymbol{m1} - \boldsymbol{m0}))) - \boldsymbol{I1}(i,j))|, \text{ otherwise} \quad (7)$$

Where H is a height of the current image block, W is a width of the current image block, $I0(i, j)$ is a sample value at a position (i, j) of a first constrained candidate reference image block, $I1(i, j)$ is a sample value at the position (i, j) of a second constrained candidate reference image block, $m0$ is a mean of sample values of the first constrained candidate reference image block, $m1$ is a mean of sample values of the second constrained candidate reference image block, d is the predefined bit-depth.

In an embodiment, the initial motion information comprises a refined motion vector of the current image block; where the determining module 603 is specifically configured to: determine the prediction image block of the current image block according to the refined motion vector and a reference image, where the reference image is a predefined image or obtained based on a predefined algorithm.

In an embodiment, the initial motion information comprises a refined motion vector of the current image block and indication of a reference image; where the determining module 603 is specifically configured to: determine the prediction image block of the current image block according to the refined motion vector and the indication of the reference image.

With the apparatus for coding an image of a video sequence provided in the present disclosure, the bit-depth of sample values of each candidate reference image block is constrained before the refining of the motion information, thus rendering the computing complexity lower and allowing for less processing time.

The present disclosure also provides an apparatus for coding an image of a video sequence which includes a processor and a memory. The memory is storing instructions that cause the processor to perform the method according to the method described above.

The present disclosure also provides a computer-readable storage medium having stored thereon instructions that when executed cause one or more processors configured to code an image of a video sequence is proposed. The instructions cause the one or more processors to perform a method described above.

The present disclosure also provides an encoder including an apparatus for coding an image of a video sequence according to the above embodiments.

The present disclosure also provides an decoder including an apparatus for coding an image of a video sequence according to the above embodiments.

The present disclosure also provides a terminal device including an above-described encoder and an above-described decoder.

Terms such as "first", "second" and the like in the specification and claims of the present disclosure as well as in the above drawings are intended to distinguish different objects, but not intended to define a particular order.

The term such as "and/or" in the embodiments of the present disclosure is merely used to describe an association between associated objects, which indicates that there may be three relationships, for example, A and/or B may indicate presence of A only, of both A and B, and of B only.

The term "a" or "an" is not intended to specify one or a single element, instead, it may be used to represent a plurality of elements where appropriate.

In the embodiments of the present disclosure, expressions such as "exemplary" or "for example" are used to indicate illustration of an example or an instance. In the embodiments of the present disclosure, any embodiment or design scheme described as "exemplary" or "for example" should not be interpreted as preferred or advantageous over other embodiments or design schemes. In particular, the use of "exemplary" or "for example" is aimed at presenting related concepts in a specific manner.

In one or more examples, the functions described may be implemented in hardware, software, firmware, or any combination thereof. If implemented in software, the functions may be stored on or transmitted over as one or more instructions or code on a computer-readable medium and executed by a hardware-based processing unit. Computer-readable media may include computer-readable storage media, which corresponds to a tangible medium such as data storage media, or communication media including any medium that facilitates transfer of a computer program from one place to another, e.g., according to a communication protocol. In this manner, computer-readable media generally may correspond to (1) tangible computer-readable storage media which is non-transitory or (2) a communication medium such as a signal or carrier wave. Data storage media may be any available media that can be accessed by one or more computers or one or more processors to retrieve instructions, code and/or data structures for implementation of the techniques described in this disclosure. A computer program product may include a computer-readable

medium.

By way of example, and not limitation, such computer-readable storage media can comprise RAM, ROM, EEPROM, CD-ROM or other optical disk storage, magnetic disk storage, or other magnetic storage devices, flash memory, or any other medium that can be used to store desired program code in the form of instructions or data structures and that can be accessed by a computer. Also, any connection is properly termed a computer-readable medium. For example, if instructions are transmitted from a website, server, or other remote source using a coaxial cable, fiber optic cable, twisted pair, digital subscriber line (DSL), or wireless technologies such as infrared, radio, and microwave, then the coaxial cable, fiber optic cable, twisted pair, DSL, or wireless technologies such as infrared, radio, and microwave are included in the definition of medium. It should be understood, however, that computer-readable storage media and data storage media do not include connections, carrier waves, signals, or other transitory media, but are instead directed to non-transitory, tangible storage media. Disk and disc, as used herein, includes compact disc (CD), laser disc, optical disc, digital versatile disc (DVD), floppy disk and Blu-ray disc, where disks usually reproduce data magnetically, while discs reproduce data optically with lasers. Combinations of the above should also be included within the scope of computer-readable media.

Instructions may be executed by one or more processors, such as one or more digital signal processors (DSPs), general purpose microprocessors, application specific integrated circuits (ASICs), field programmable logic arrays (FPGAs), or other equivalent integrated or discrete logic circuitry. Accordingly, the term "processor," as used herein may refer to any of the foregoing structure or any other structure suitable for implementation of the techniques described herein. In addition, in some aspects, the functionality described herein may be provided within dedicated hardware and/or software modules configured for encoding and decoding, or incorporated in a combined codec. Also, the techniques could be fully implemented in one or more circuits or logic elements.

The techniques of this disclosure may be implemented in a wide variety of devices or apparatuses, including a wireless handset, an integrated circuit (IC) or a set of ICs (e.g., a chip set). Various components, modules, or units are described in this disclosure to emphasize functional aspects of devices configured to perform the disclosed techniques, but do not

necessarily require realization by different hardware units. Rather, as described above, various units may be combined in a codec hardware unit or provided by a collection of interoperative hardware units, including one or more processors as described above, in conjunction with suitable software and/or firmware.

CLAIMS

1. A method for coding an image of a video sequence, wherein the image comprises a current image block, the method comprises:

obtaining initial motion information of the current image block;

obtaining sample values of at least two candidate reference image blocks corresponding to the current image block according to the initial motion information;

constraining a bit-depth of sample values of each candidate reference image block to a predefined bit-depth;

obtaining refined motion information based on constrained sample values of at least two constrained candidate reference image blocks; and

determining a prediction image block of the current image block according to the refined motion information.

2. The method according to claim 1, wherein the constraining a bit-depth of sample values of each candidate reference image block to a predefined bit-depth comprises:

performing a right shift by an amount of (b-d) on the bit-depth of sample values of each candidate reference image block, wherein b is the bit-depth of the sample values of the each candidate reference image block, d is the predefined bit-depth, and b is greater than d.

3. The method according to claim 1 or 2, wherein the obtaining initial motion information of a current image block comprises:

using a first cost function to determine motion information with the least cost from a merge candidate list as the initial motion information, wherein the merge candidate list comprises motion information of adjacent blocks of the current image block.

4. The method according to claim 1 or 2, wherein the obtaining initial motion information of a current image block comprises:

determining motion information identified by a merge index from a merge candidate list as the initial motion information, wherein the merge candidate list comprises motion information of adjacent blocks of the current image block.

5. The method according to any one of claims 1-4, wherein the obtaining sample values of at least two candidate reference image blocks corresponding to the current image block comprises:

determining a position of a candidate reference image block according to the initial motion information and a position of the current image block;

determining whether sample values of the candidate reference image block are available based on the position of the candidate reference image block, if the sample values of the candidate reference image block are unavailable, performing an interpolation operation to obtain the sample values of the candidate reference image block.

6. The method according to any one of claims 1-5, wherein the obtaining refined motion information based on constrained sample values of the at least two constrained candidate reference image blocks comprises:

using a second cost function to determine a position of a constrained candidate reference image block with the least cost;

obtaining the refined motion information according to the position of the constrained candidate reference image block with the least cost.

7. The method according to claim 6, wherein the second cost function indicates a difference between a constrained candidate reference image block and a template of the current image block, wherein the template is obtained based on the initial motion information and a position of the current image block and a bit-depth of sample values of the template is constrained to the predefined bit-depth.

8. The method according to claim 7, wherein the cost is calculated as:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{I0}(i,j) - \boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I1}(i,j) + (\boldsymbol{m0} - \boldsymbol{m1}))))|, \text{ if } \boldsymbol{m0} > \boldsymbol{m1}$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I0}(i,j) + (\boldsymbol{m1} - \boldsymbol{m0}))) - \boldsymbol{I1}(i,j))|, \text{otherwise}$$

wherein $\boldsymbol{H}$ is a height of the current image block, $\boldsymbol{W}$ is a width of the current image block, $\boldsymbol{I0(i, j)}$ is a sample value at a position (i, j) of a constrained candidate reference image block, $\boldsymbol{I1(i, j)}$ is a sample value at the position (i, j) of the template, $\boldsymbol{m0}$ is a mean of sample values of the constrained candidate reference image block, $\boldsymbol{m1}$ is a mean of sample values of the template, d is the predefined bit-depth.

9. The method according to claim 7 or 8, wherein the at least two constrained candidate reference image blocks comprises a set of first constrained candidate reference image blocks and a set of second constrained candidate reference image blocks;

wherein the initial motion information comprises first initial motion information and second initial motion information, the method further comprises:

obtaining a first initial reference image block based on the first initial motion information and a position of the current image block;

obtaining a second initial reference image block based on the second initial motion information and the position of the current image block;

calculating a weighted combination of the first initial reference image block and the second initial reference image block as the template;

wherein the using a second cost function to determine a position of a constrained candidate reference image block with the least cost comprises:

calculating differences between the template and each first constrained candidate reference image block and determining a first position of a first constrained reference image block with the smallest difference;

calculating differences between the template and each second constrained candidate reference image block and determining a second position of a second constrained reference image block with the smallest difference;

wherein the obtaining the refined motion information according to a position of the constrained candidate reference image block comprises:

obtaining the refined motion information according to the first position and the second position.

10. The method according to claim 6, wherein the at least two constrained candidate reference image blocks comprises a set of first constrained candidate reference image blocks and a set of second constrained candidate reference image blocks, and the second cost function indicates a difference between a first constrained candidate reference image block and a second constrained candidate reference image block.

11. The method according to claim 10, wherein the cost is calculated as:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(I0(i,j) - MIN((1 \ll d) - 1, (I1(i,j) + (m0 - m1))))|, \text{ if } m0 > m1$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(MIN((1 \ll d) - 1, (I0(i,j) + (m1 - m0))) - I1(i,j))|, \text{ otherwise}$$

wherein H is a height of the current image block, W is a width of the current image block, $I0(i, j)$ is a sample value at a position (i, j) of a first constrained candidate reference image block, $I1(i, j)$ is a sample value at the position (i, j) of a second constrained candidate reference image block, $m0$ is a mean of sample values of the first constrained candidate reference image block, $m1$ is a mean of sample values of the second constrained candidate reference image block, d is the predefined bit-depth.

12. The method according to any one of claims 1-11, wherein the refined motion information comprises a refined motion vector of the current image block;

wherein the determining a prediction image block of the current image block according to the refined motion information comprises:

determining the prediction image block of the current image block according to the refined motion vector and a reference image, wherein the reference image is a predefined image or obtained based on a predefined algorithm.

13. The method according to any one of claims 1-11, wherein the refined motion information comprises a refined motion vector of the current image block and indication of a reference image;

wherein the determining a prediction image block of the current image block according to the refined motion information comprises:

determining the prediction image block of the current image block according to the refined motion vector and the indication of the reference image.

14. The method according to claim 3, wherein the first cost function is one of a sum of absolute difference (SAD), a mean-removed sum of absolute difference (MR-SAD) or a sum of squared differences (SSD).

15. The method according to any one of claims 6-11, wherein the second cost function is one of a sum of absolute difference (SAD), a mean-removed sum of absolute difference (MR-SAD) or a sum of squared differences (SSD).

16. The method according to any one of claims 1-15, wherein the constrained candidate reference image block refers to the candidate reference image block with constrained sample values.

17. An apparatus for coding an image of a video sequence, wherein the image comprises a current image block, the apparatus comprises:

an obtaining module, configured to obtain initial motion information of the current image block;

the obtaining module is further configured to obtain sample values of at least two candidate reference image blocks corresponding to the current image block according to the initial motion information;

a constraining module, configured to constrain a bit-depth of sample values of each candidate reference image block to a predefined bit-depth;

the obtaining module is further configured to obtain refined motion information based on constrained sample values of at least two constrained candidate reference image blocks; and

a determining module, configured to determine a prediction image block of the current image block according to the refined motion information.

18. The apparatus according to claim 17, wherein the constraining module is specifically configured to:

perform a right shift by an amount of (b-d) on the bit-depth of sample values of each candidate reference image block, wherein b is the bit-depth of the sample values of the each candidate reference image block, d is the predefined bit-depth, and b is greater than d.

19. The apparatus according to claim 17 or 18, wherein the obtaining module is specifically configured to:

use a first cost function to determine motion information with the least cost from a merge candidate list as the initial motion information, wherein the merge candidate list comprises motion information of adjacent blocks of the current image block.

20. The apparatus according to claim 17 or 18, wherein the obtaining module is specifically configured to:

determine motion information identified by a merge index from a merge candidate list as the initial motion information, wherein the merge candidate list comprises motion information of adjacent blocks of the current image block.

21. The apparatus according to any one of claims 17-20, wherein the obtaining module is specifically configured to:

determine a position of a candidate reference image block according to the initial motion information and a position of the current image block;

determine whether sample values of the candidate reference image block are available based on the position of the candidate reference image block, if sample values of the candidate reference image block are unavailable, performing an interpolation operation to obtain the sample values of the candidate reference image block.

22. The apparatus according to any one of claims 17-21, wherein the obtaining module is specifically configured to:

use a second cost function to determine a position of a constrained candidate reference image block with the least cost;

obtain the refined motion information according to the position of the constrained candidate reference image block with the least cost.

23. The apparatus according to claim 22, wherein the second cost function indicates a difference between a constrained candidate reference image block and a template of the current image block, wherein the template is obtained based on the initial motion information and a position of the current image block and a bit-depth of sample values of the template is constrained to the predefined bit-depth.

24. The apparatus according to claim 23, wherein the cost is calculated as:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{I0}(i,j) - \boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I1}(i,j) + (\boldsymbol{m0} - \boldsymbol{m1}))))|, \text{ if } \boldsymbol{m0} > \boldsymbol{m1}$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{MIN}((\mathbf{1} \ll \boldsymbol{d}) - \mathbf{1}, (\boldsymbol{I0}(i,j) + (\boldsymbol{m1} - \boldsymbol{m0}))) - \boldsymbol{I1}(i,j))|, \text{otherwise}$$

wherein $\boldsymbol{H}$ is a height of the current image block, $\boldsymbol{W}$ is a width of the current image block, $\boldsymbol{I0(i, j)}$ is a sample value at a position (i, j) of a constrained candidate reference image block, $\boldsymbol{I1(i, j)}$ is a sample value at the position (i, j) of the template, $\boldsymbol{m0}$ is a mean of sample values

of the constrained candidate reference image block, $\boldsymbol{m1}$ is a mean of sample values of the template, d is the predefined bit-depth.

25. The apparatus according to claim 23 or 24, wherein the at least two constrained candidate reference image blocks comprises a set of first constrained candidate reference image blocks and a set of second constrained candidate reference image blocks;

wherein the obtaining module is specifically configured to:

obtain a first initial reference image block based on the first initial motion information and a position of the current image block;

obtain a second initial reference image block based on the second initial motion information and the position of the current image block;

calculate a weighted combination of the first initial reference image block and the second initial reference image block as the template;

calculate differences between the template and each first constrained candidate reference image block and determine a first position of a first constrained reference image block with the smallest difference;

calculate differences between the template and each second constrained candidate reference image block and determine a second position of a second constrained reference image block with the smallest difference;

wherein the determining module is specifically configured to:

obtain the refined motion information according to the first position and the second position.

26. The apparatus according to claim 22, wherein the at least two constrained candidate reference image blocks comprises a set of first constrained candidate reference image blocks and a set of second constrained candidate reference image blocks, and the second cost function indicates a difference between a first constrained candidate reference image block and a second constrained candidate reference image block.

27. The apparatus according to claim 26, wherein the cost is calculated as:

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{I0}(i,j) - \boldsymbol{MIN}((\boldsymbol{1} \ll \boldsymbol{d}) - \boldsymbol{1}, (\boldsymbol{I1}(i,j) + (\boldsymbol{m0} - \boldsymbol{m1}))))|, \text{ if } \boldsymbol{m0} > \boldsymbol{m1}$$

$$\text{cost} = \sum_{i=0}^{H-1}\sum_{j=0}^{W-1}|(\boldsymbol{MIN}((\boldsymbol{1} \ll \boldsymbol{d}) - \boldsymbol{1}, (\boldsymbol{I0}(i,j) + (\boldsymbol{m1} - \boldsymbol{m0}))) - \boldsymbol{I1}(i,j))|, \text{otherwise}$$

wherein H is a height of the current image block, W is a width of the current image block, $I0(i, j)$ is a sample value at a position (i, j) of a first constrained candidate reference image block, $I1(i, j)$ is a sample value at the position (i, j) of a second constrained candidate reference image block, $m0$ is a mean of sample values of the first constrained candidate reference image block, $m1$ is a mean of sample values of the second constrained candidate reference image block, d is the predefined bit-depth.

28. The apparatus according to any one of claims 17-27, wherein the refined motion information comprises a refined motion vector of the current image block;

wherein the determining module is specifically configured to:

determine the prediction image block of the current image block according to the refined motion vector and a reference image, wherein the reference image is a predefined image or obtained based on a predefined algorithm.

29. The apparatus according to any one of claims 17-27, wherein the initial motion information comprises a refined motion vector of the current image block and indication of a reference image;

wherein the determining module is specifically configured to:

determine the prediction image block of the current image block according to the refined motion vector and the indication of the reference image.

30. An encoder, comprising an apparatus for coding an image of a video sequence according to any one of claims 17-29.

31. A decoder, comprising an apparatus for coding an image of a video sequence according to any one of claims 17-29.

32. A terminal device, comprising an encoder according to claim 30 and a decoder according to claim 31.