

corresponding to the top-right sample location of the current block; vx_2 is a horizontal component of the motion vector corresponding to the bottom-left sample location of the current block, and vy_2 is a vertical component of the motion vector corresponding to the bottom-left sample location of the current block; vx_4 is a horizontal component of a motion vector corresponding to the top-left sample location of the neighboring block, and vy_4 is a vertical component of the motion vector corresponding to the top-left sample location of the neighboring block;

5 vx_5 is a horizontal component of a motion vector corresponding to the top-right sample location of the neighboring block, and vy_5 is a vertical component of the motion vector corresponding to the top-right sample location of the neighboring block; x_0 is a horizontal coordinate of the top-left sample location of the current block, and y_0 is a vertical coordinate of the top-left sample location of the current block; x_1 is a horizontal coordinate of the top-right sample location of the current block, and y_1 is a vertical coordinate of the top-right sample location of the current block;

10 x_2 is a horizontal coordinate of the bottom-left sample location of the current block, and y_2 is a vertical coordinate of the bottom-left sample location of the current block; x_4 is a horizontal coordinate of the top-left sample location of the neighboring block, and y_4 is a vertical coordinate of the top-left sample location of the neighboring block; and x_5 is a horizontal coordinate of the top-right sample location of the neighboring block.

15 **[0011]** According to the first aspect, in a possible implementation, $N=3$ and $K=2$. To be specific, if a 6-parameter affine transform model is used for an affine decoding block (an affine coding block on the encoder side), and a 4-parameter affine transform model is used for the current block, candidate motion vectors of two control points of the current block are obtained based on the 6-parameter affine transform model used for the neighboring block of the current block.

20 **[0012]** For example, the candidate motion vectors of the two control points of the current block include a motion vector (vx_0, vy_0) at the top-left sample location (or referred to as the top-left corner, which is the same below) (x_0, y_0) of the current block and a motion vector (vx_1, vy_1) at the top-right sample location (or referred to as the top-right corner, which is the same below) (x_1, y_1) of the current block; and that the candidate motion vectors of the two control points of the current block are obtained based on the 6-parameter affine transform model used for the neighboring block of the current block includes that the candidate motion vectors of the two control points of the current block are calculated according to the following formulas:

25

$$\begin{cases} vx_0 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vy_5 - vy_4)}{y_5 - y_4} \times (y_0 - y_4) \\ vy_0 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vx_5 - vx_4)}{y_5 - y_4} \times (y_0 - y_4) \end{cases}$$

$$\begin{cases} vx_1 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vy_5 - vy_4)}{y_5 - y_4} \times (y_1 - y_4) \\ vy_1 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vx_5 - vx_4)}{y_5 - y_4} \times (y_1 - y_4) \end{cases}$$

where vx_0 is a horizontal component of the motion vector corresponding to the top-left sample location of the current block, and vy_0 is a vertical component of the motion vector corresponding to the top-left sample location of the current block; vx_1 is a horizontal component of the motion vector corresponding to the top-right sample location of the current block, and vy_1 is a vertical component of the motion vector corresponding to the top-right sample location of the current block; vx_4 is a horizontal component of a motion vector corresponding to the top-left sample location of the neighboring block, and vy_4 is a vertical component of the motion vector corresponding to the top-left sample location of the neighboring block; vx_5 is a horizontal component of a motion vector corresponding to the top-right sample location of the neighboring block, and vy_5 is a vertical component of the motion vector corresponding to the top-right sample location of the neighboring block; vx_6 is a horizontal component of a motion vector corresponding to the bottom-left sample location of the neighboring block, and vy_6 is a vertical component of the motion vector corresponding to the bottom-left sample location of the neighboring block; x_0 is a horizontal coordinate of the top-left sample location of the current block, and y_0 is a vertical coordinate of the top-left sample location of the current block; x_1 is a horizontal coordinate of the top-right sample location of the current block, and y_1 is a vertical coordinate of the top-right sample location of the current block; x_4 is a horizontal coordinate of the top-left sample location of the neighboring block, and y_4 is a vertical coordinate of the top-left sample location of the neighboring block; x_5 is a horizontal coordinate of the top-right sample location of the neighboring block; and y_6 is a vertical coordinate of the bottom-left sample location of the neighboring block.

[0013] It can be learned that, in this embodiment of the present invention, in a phase of parsing the current block (for example, in the phase of constructing the candidate motion vector list), the affine transform model of the

motion vectors of the K control points of the current block; and obtaining the predicted motion vector of each subblock of the current block based on the 6-parameter affine transform model of the current block.

[0018] In other words, in this solution, regardless of the affine transform model used for the current block in the parsing phase (the list construction phase), in the phase of reconstructing the current block, the 6-parameter affine transform model is uniformly used to obtain motion vector information of each subblock of the current block, to reconstruct each subblock. For example, if the 4-parameter affine transform model or an 8-parameter bilinear model is used in the parsing phase, the 6-parameter affine transform model of the current block is further constructed. For example, if the 6-parameter affine transform model is used in the parsing phase, the 6-parameter affine transform model of the current block continues to be used in the reconstruction phase.

[0019] For example, the 4-parameter affine transform model is used for the current block in the parsing phase, and the 4-parameter affine transform model or another parameter affine model may be used for the neighboring block. Therefore, after motion vectors of the two control points of the current block are obtained, for example, a motion vector (vx4, vy4) of the top-left control point (x4, y4) of the current block and a motion vector (vx5, vy5) of the top-right control point (x5, y5) of the current block are obtained, in the phase of reconstructing the current block, the 6-parameter affine transform model needs to be constructed based on the motion vectors of the two control points of the current block.

[0020] For example, based on the motion vector (vx0, vy0) of the top-left control point (x0, y0) of the current block and the motion vector (vx1, vy1) of the top-right control point (x1, y1) of the current block, a motion vector of a third control point may be obtained according to the following formula. The motion vector of the third control point is, for example, the motion vector (vx2, vy2) of the bottom-left corner (x2, y2) of the current block.

$$\begin{cases} vx_2 = -\frac{vy_1 - vy_0}{W} H + vx_1 \\ vy_2 = +\frac{vx_1 - vx_0}{W} H + vy_1 \end{cases}$$

[0021] Then, the 6-parameter affine model of the current block in the reconstruction phase is obtained by using the motion vector (vx0, vy0) of the top-left control point (x0, y0) of the current block, the motion vector (vx1, vy1) of the top-right control point (x1, y1) of the current block, and the motion vector (vx2, vy2) of the bottom-left control point (x2, y2) of the current block. The formula for the 6-parameter affine model is as follows:

$$\begin{cases} vx = \frac{vx_1 - vx_0}{W} x + \frac{vx_2 - vx_0}{H} y + vx_0 \\ vy = \frac{vy_1 - vy_0}{W} x + \frac{vy_2 - vy_0}{H} y + vy_0 \end{cases}$$

- [0029]** In some specific embodiments of the present invention, the candidate motion vector of the control point of the current block may also be obtained by using the first motion model-based motion vector prediction method described in this specification, and added to the candidate motion vector list corresponding to the merge mode.
- 5 **[0030]** In some other specific embodiments of the present invention, the candidate motion vector of the control point of the current block may alternatively be obtained separately by using the first motion model-based motion vector prediction method and the constructed control point motion vector prediction method, and added to the candidate motion vector list corresponding to the merge mode.
- 10 **[0031]** According to the first aspect, in a possible implementation, when there are a plurality of neighboring blocks, in other words, when the current block has a plurality of neighboring affine decoding blocks, in a possible embodiment, both the encoder side and the decoder side can first obtain a candidate motion vector of a control point of the current block by using an affine decoding block whose quantity of model parameters is the same as that of the current block, and add the obtained candidate motion vector of the control point to the candidate motion vector list corresponding to the AMVP mode. Then, a candidate motion vector of the control point of the current block may be
15 obtained by using an affine decoding block whose quantity of model parameters is different from that of the current block, and added to the candidate motion vector list corresponding to the AMVP mode. In this way, the candidate motion vector that is of the control point of the current block and that is obtained by using the affine decoding block whose quantity of model parameters is the same as that of the current block is located at a front location of the list. This design helps reduce a quantity of bits transmitted in the bitstream.
- 20 **[0032]** According to the first aspect, in a possible implementation, in a process in which the decoder side derives the candidate motion vector of the control point of the current block, flag information (flag) of an affine transform model of an affine decoding block may need to be obtained. The flag is pre-stored locally on the decoder side, and is used to indicate the affine transform model that is of the affine decoding block and that is actually used to predict a subblock of the affine decoding block.
- 25 **[0033]** For example, in an application scenario, when the decoder side determines, by identifying the flag of the affine decoding block, that a quantity of model parameters of the affine transform model actually used for the affine decoding block is different from (or the same as) that of the affine transform model used for the current block, the decoder side is triggered to derive the candidate motion vector of the control point of the current block by using the affine transform model actually used for the affine decoding block.
- 30 **[0034]** According to the first aspect, in a possible implementation, in the process in which the decoder side derives the candidate motion vector of the control point of the current block, the flag of the affine transform model

of the affine decoding block may not be required.

[0035] For example, in an application scenario, after the decoder side determines the affine transform model used for the current block, the decoder side obtains control points of a specific quantity (the specific quantity is the same as or different from a quantity of control points of the current block) of the affine decoding block, constructs the affine transform model by using the control points of the specific quantity of the affine decoding block, and then derives the candidate motion vector of the control point of the current block by using the affine transform model.

[0036] According to a second aspect, an embodiment of the present invention provides another motion vector prediction method. The method includes: parsing a bitstream to obtain an index value of a candidate motion vector list; constructing the candidate motion vector list, where the candidate motion vector list includes candidate motion vectors of K control points of a current block, the candidate motion vectors of the K control points of the current block are obtained based on a 2N-parameter affine transform model used for a neighboring block of the current block, the 2N-parameter affine transform model is obtained based on motion vectors of N control points of the neighboring block, N is an integer greater than or equal to 2 and less than or equal to 4, K is an integer greater than or equal to 2 and less than or equal to 4, the neighboring block is a decoded image block spatially adjacent to the current block, and the current block includes a plurality of subblocks; determining, in the candidate motion vector list, target candidate motion vectors of the K control points of the current block based on the index value; obtaining a 6-parameter affine transform model of the current block based on the target candidate motion vectors of the K control points of the current block; and obtaining a predicted motion vector of each subblock of the current block based on the 6-parameter affine transform model of the current block.

[0037] It can be learned that, in this embodiment of the present invention, the 6-parameter affine transform model can be uniformly used to predict the current block in a phase of reconstructing the current block. A larger quantity of parameters of a motion model describing affine motion of the current block indicates higher precision and higher calculation complexity. In this solution, the 6-parameter affine transform model constructed in the reconstruction phase can describe affine transform such as translation, scaling, and rotation of an image block, and achieve a good balance between model complexity and a modeling capability. Therefore, this solution can improve coding efficiency and accuracy in predicting the current block, and satisfy a user requirement.

[0038] According to the second aspect, in a possible implementation, $N=2$ and $K=3$.

[0039] Correspondingly, candidate motion vectors of two control points of the current block are obtained based on a 4-parameter affine transform model used for the neighboring block of the current block.

[0040] According to the second aspect, in a possible implementation, $N=3$ and $K=2$. Correspondingly, candidate motion vectors of two control points of the current block are obtained based on the 6-parameter affine

a decoder, configured to: construct a candidate motion vector list, where the candidate motion vector list includes candidate motion vectors of K control points of a current block, the candidate motion vectors of the K control points of the current block are obtained based on a 2×N-parameter affine transform model used for a neighboring block of the current block, the 2×N-parameter affine transform model is obtained based on motion
5 vectors of N control points of the neighboring block, N is an integer greater than or equal to 2 and less than or equal to 4, K is an integer greater than or equal to 2 and less than or equal to 4, N is not equal to K, the neighboring block is a decoded image block spatially adjacent to the current block, and the current block includes a plurality of subblocks; determine, in the candidate motion vector list, target candidate motion vectors of the K control points of the current block based on the index value; and obtain a predicted motion vector of each subblock of the current
10 block based on the target candidate motion vectors of the K control points of the current block.

[0049] According to the fifth aspect, in some implementations, N is equal to 2, and K is equal to 3. Correspondingly, candidate motion vectors of three control points of the current block are obtained based on a 4-parameter affine transform model used for the neighboring block of the current block.

[0050] According to the fifth aspect, in some implementations, the candidate motion vectors of the three
15 control points of the current block include a motion vector at the top-left sample location of the current block, a motion vector at the top-right sample location of the current block, and a motion vector at the bottom-left sample location of the current block.

[0051] The decoder is configured to calculate the candidate motion vectors of the three control points of the current block according to the following formulas:

$$\begin{cases} vx_0 = vx_4 + \frac{(vx_1 - vx_4)}{x_1 - x_4} \times (x_0 - x_4) - \frac{(vy_1 - vy_4)}{x_1 - x_4} \times (y_0 - y_4) \\ vy_0 = vx_4 + \frac{(vy_1 - vy_4)}{x_1 - x_4} \times (x_0 - x_4) + \frac{(vx_1 - vx_4)}{x_1 - x_4} \times (y_0 - y_4) \end{cases}$$

$$\begin{cases} vx_1 = vx_4 + \frac{(vx_1 - vx_4)}{x_1 - x_4} \times (x_1 - x_4) - \frac{(vy_1 - vy_4)}{x_1 - x_4} \times (y_1 - y_4) \\ vy_1 = vx_4 + \frac{(vy_1 - vy_4)}{x_1 - x_4} \times (x_1 - x_4) + \frac{(vx_1 - vx_4)}{x_1 - x_4} \times (y_1 - y_4) \end{cases}$$

$$\begin{cases} vx_2 = vx_4 + \frac{(vx_1 - vx_4)}{x_1 - x_4} \times (x_2 - x_4) - \frac{(vy_1 - vy_4)}{x_1 - x_4} \times (y_2 - y_4) \\ vy_2 = vx_4 + \frac{(vy_1 - vy_4)}{x_1 - x_4} \times (x_2 - x_4) + \frac{(vx_1 - vx_4)}{x_1 - x_4} \times (y_2 - y_4) \end{cases}$$

where vx_0 is a horizontal component of the motion vector corresponding to the top-left sample
location of the current block, and vy_0 is a vertical component of the motion vector corresponding to the top-left
25 sample location of the current block; vx_1 is a horizontal component of the motion vector corresponding to the

top-right sample location of the current block, and vy_1 is a vertical component of the motion vector corresponding to the top-right sample location of the current block; vx_2 is a horizontal component of the motion vector corresponding to the bottom-left sample location of the current block, and vy_2 is a vertical component of the motion vector corresponding to the bottom-left sample location of the current block; vx_4 is a horizontal component of a motion vector corresponding to the top-left sample location of the neighboring block, and vy_4 is a vertical component of the motion vector corresponding to the top-left sample location of the neighboring block; vx_5 is a horizontal component of a motion vector corresponding to the top-right sample location of the neighboring block, and vy_5 is a vertical component of the motion vector corresponding to the top-right sample location of the neighboring block; x_0 is a horizontal coordinate of the top-left sample location of the current block, and y_0 is a vertical coordinate of the top-left sample location of the current block; x_1 is a horizontal coordinate of the top-right sample location of the current block, and y_1 is a vertical coordinate of the top-right sample location of the current block; x_2 is a horizontal coordinate of the bottom-left sample location of the current block, and y_2 is a vertical coordinate of the bottom-left sample location of the current block; x_4 is a horizontal coordinate of the top-left sample location of the neighboring block, and y_4 is a vertical coordinate of the top-left sample location of the neighboring block; and x_5 is a horizontal coordinate of the top-right sample location of the neighboring block.

[0052] According to the fifth aspect, in some implementations, N is equal to 3, and K is equal to 2. Correspondingly, candidate motion vectors of two control points of the current block are obtained based on a 6-parameter affine transform model used for the neighboring block of the current block.

[0053] According to the fifth aspect, in some implementations, the candidate motion vectors of the two control points of the current block include a motion vector at the top-left sample location of the current block and a motion vector at the top-right sample location of the current block.

[0054] The decoder is configured to calculate the candidate motion vectors of the two control points of the current block according to the following formulas:

$$\begin{cases} vx_0 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vy_5 - vy_4)}{y_5 - y_4} \times (y_0 - y_4) \\ vy_0 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vx_5 - vx_4)}{y_5 - y_4} \times (y_0 - y_4) \end{cases}$$

$$\begin{cases} vx_1 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vy_5 - vy_4)}{y_5 - y_4} \times (y_1 - y_4) \\ vy_1 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vx_5 - vx_4)}{y_5 - y_4} \times (y_1 - y_4) \end{cases}$$

where vx_0 is a horizontal component of the motion vector corresponding to the top-left sample location of the current block, and vy_0 is a vertical component of the motion vector corresponding to the top-left sample location of the current block; vx_1 is a horizontal component of the motion vector corresponding to the top-right sample location of the current block, and vy_1 is a vertical component of the motion vector corresponding to the top-right sample location of the current block; vx_4 is a horizontal component of a motion vector corresponding to the top-left sample location of the neighboring block, and vy_4 is a vertical component of the motion vector corresponding to the top-left sample location of the neighboring block; vx_5 is a horizontal component of a motion vector corresponding to the top-right sample location of the neighboring block, and vy_5 is a vertical component of the motion vector corresponding to the top-right sample location of the neighboring block; vx_6 is a horizontal component of a motion vector corresponding to the bottom-left sample location of the neighboring block, and vy_6 is a vertical component of the motion vector corresponding to the bottom-left sample location of the neighboring block; x_0 is a horizontal coordinate of the top-left sample location of the current block, and y_0 is a vertical coordinate of the top-left sample location of the current block; x_1 is a horizontal coordinate of the top-right sample location of the current block, and y_1 is a vertical coordinate of the top-right sample location of the current block; x_4 is a horizontal coordinate of the top-left sample location of the neighboring block, and y_4 is a vertical coordinate of the top-left sample location of the neighboring block; x_5 is a horizontal coordinate of the top-right sample location of the neighboring block; and y_6 is a vertical coordinate of the bottom-left sample location of the neighboring block.

[0055] According to the fifth aspect, in some implementations, the decoder is specifically configured to:

obtain a 2×K-parameter affine transform model of the current block based on the target candidate motion vectors of the K control points of the current block; and

obtain the predicted motion vector of each subblock of the current block based on the 2×K-parameter affine transform model of the current block.

DESCRIPTION OF EMBODIMENTS

[0083] The following briefly describes related concepts in the embodiments of the present invention first. The technical solutions in the embodiments of the present invention may not only be applied to existing video coding standards (for example, standards such as H.264 and HEVC), but also be applied to future video coding standards (for example, the H.266 standard).

[0084] Video coding usually refers to processing a sequence of images that constitute a video or a video sequence. In the field of video coding, the terms "picture (picture)", "frame (frame)", and "image (image)" may be used as synonyms. Video coding used in this specification includes video encoding and video decoding. Video coding is performed on a source side, and usually includes processing (for example, by compressing) an original video image to reduce an amount of data for representing the video image, for more efficient storage and/or transmission. Video decoding is performed on a destination side, and usually includes inverse processing compared with an encoder to reconstruct the video image. "Coding" of a video image in the embodiments should be understood as "encoding" or "decoding" of a video sequence. A combination of an encoding part and a decoding part is also referred to as CODEC (encoding and decoding).

[0085] The video sequence includes a series of images (picture), the image is further split into slices (slice), and the slice is further split into blocks (block). Video coding is performed by block. In some new video coding standards, a concept "block" is further extended. For example, a macroblock (macroblock, MB) is introduced in the H.264 standard. The macroblock may be further split into a plurality of prediction blocks (predictor) that can be used for predictive coding. In the high efficiency video coding (high efficiency video coding, HEVC) standard, basic concepts such as "coding unit" (coding unit, CU), "prediction unit" (prediction unit, PU), and "transform unit" (transform unit, TU) are used. A plurality of block units are obtained through functional division, and are described by using a new tree-based structure. For example, a CU may be split into smaller CUs based on a quadtree, and the smaller CU may further be split, to generate a quadtree structure. The CU is a basic unit for splitting and encoding a coded image. A PU and a TU also have similar tree structures. The PU may correspond to a prediction block, and is a basic unit for predictive coding. The CU is further split into a plurality of PUs based on a splitting pattern. The TU may correspond to a transform block, and is a basic unit for transforming a prediction residual. However, in essence, all of the CU, the PU, and the TU are conceptually blocks (or image blocks).

[0086] For example, in HEVC, a CTU is split into a plurality of CUs by using a quadtree structure represented as a coding tree. A decision on whether to encode an image area by using inter-image (temporal) or intra-image (spatial) prediction is made at a CU level. Each CU may be further split into one, two, or four PUs based on a PU

splitting pattern. Inside one PU, a same prediction process is applied, and related information is transmitted to a decoder on a PU basis. After obtaining a residual block by applying the prediction process based on the PU splitting pattern, the CU may be partitioned into transform units (transform unit, TU) based on another quadtree structure similar to the coding tree used for the CU. In the recent development of video compression technologies, a quadtree plus binary tree (Quad-tree plus binary tree, QTBT) partition frame is used to partition a coding block. In a QTBT block structure, the CU may be square or rectangular.

[0087] In this specification, for ease of description and understanding, a to-be-encoded image block in a current coded image may be referred to as a current block. For example, in encoding, the current block is a block currently being encoded, and in decoding, the current block is a block currently being decoded. A decoded image block, in a reference image, used to predict the current block is referred to as a reference block. In other words, the reference block is a block that provides a reference signal for the current block, and the reference signal represents a pixel value in the image block. A block that provides a prediction signal for the current block in the reference image may be referred to as a prediction block, and the prediction signal represents a pixel value, a sampling value, or a sampling signal in the prediction block. For example, after a plurality of reference blocks are traversed, an optimal reference block is found. The optimal reference block provides prediction for the current block, and may be referred to as the prediction block.

[0088] The following describes a video coding system in the embodiments of the present invention. FIG. 1 is a block diagram of an example video coding system according to an embodiment of the present invention. As used in this specification, the term "video codec" generally refers to a video encoder and a video decoder. In the embodiments of the present invention, the term "video coding" or "coding" may generally refer to video encoding or video decoding. A video encoder 100 and a video decoder 200 in the video coding system are configured to predict motion information, for example, a motion vector, of a current coded image block or a subblock of the current coded image block according to various method examples described in any one of a plurality of new inter prediction modes provided in the embodiments of the present invention, so that a predicted motion vector maximally approximates a motion vector obtained by using a motion estimation method. In this way, a motion vector difference does not need to be transmitted during encoding, thereby further improving coding performance.

[0089] As shown in FIG. 1, the video coding system includes a source apparatus 10 and a destination apparatus 20. The source apparatus 10 generates encoded video data. Therefore, the source apparatus 10 may be referred to as a video encoding apparatus. The destination apparatus 20 may decode the encoded video data generated by the source apparatus 10. Therefore, the destination apparatus 20 may be referred to as a video decoding apparatus. In various implementation solutions, the source apparatus 10, the destination apparatus 20, or both the

(blocking artifacts). Then, the reconstructed image block is stored as the reference block in the decoded image buffer 107, and may be used by the inter predictor 110 as the reference block to perform inter prediction on a block in a subsequent video frame or image.

[0122] It should be understood that other structural variants of the video encoder 100 can be used to encode a video stream. For example, for some image blocks or image frames, the video encoder 100 may directly quantize a residual signal, and correspondingly, processing by the transformer 101 and the inverse transformer 105 is not required. Alternatively, for some image blocks or image frames, the video encoder 100 does not generate residual data, and correspondingly, processing by the transformer 101, the quantizer 102, the inverse quantizer 104, and the inverse transformer 105 is not required. Alternatively, the video encoder 100 may directly store the reconstructed image block as the reference block, without processing by the filter unit 106. Alternatively, the quantizer 102 and the inverse quantizer 104 in the video encoder 100 may be combined.

[0123] Specifically, in this embodiment of the present invention, the video encoder 100 is configured to perform the motion vector prediction method described in the following embodiments.

[0124] FIG. 2B is a block diagram of an example video decoder 200 according to an embodiment of the present invention. In the example in FIG. 2B, the video decoder 200 includes an entropy decoder 203, a prediction processing unit 208, an inverse quantizer 204, an inverse transformer 205, a summator 211, a filter unit 206, and a decoded image buffer 207. The prediction processing unit 208 may include an inter predictor 210 and an intra predictor 209. In some examples, the video decoder 200 may perform a decoding process that is substantially inverse to the encoding process described with respect to the video encoder 100 in FIG. 2A.

[0125] During decoding, the video decoder 200 receives, from the video encoder 100, an encoded video bitstream that represents an image block of an encoded video slice and an associated syntax element. The video decoder 200 may receive video data from a network entity 42, and optionally, may further store the video data in a video data memory (which is not shown in the figure). The video data memory may store video data that is to be decoded by a component of the video decoder 200, for example, the encoded video bitstream. The video data stored in the video data memory may be obtained from, for example, a local video source such as a storage apparatus 40 or a camera through wired or wireless network communication of the video data, or by accessing a physical data storage medium. The video data memory may be used as the decoded image buffer (DPB) configured to store encoded video data that is from the encoded video bitstream. Therefore, although the video data memory is not shown in FIG. 2B, the video data memory and the DPB 207 may be a same memory, or may be separately disposed memories. The video data memory and the DPB 207 each may be constituted by any one of a plurality of memory apparatuses, for example, a dynamic random access memory (DRAM) including a synchronous DRAM (SDRAM),

configured to receive data; a processor, a logic unit, or a central processing unit (CPU) 430 that is configured to process the data; a transmitter unit (Tx) 440 and egress ports 450 that are configured to transmit data; and a memory 460 configured to store the data. The video coding device 400 may further include an optical-to-electrical and electrical-to-optical (EO) conversion component coupled to the ingress ports 410, the receiver unit 420, the transmitter unit 440, and the egress ports 450, for egress or ingress of an optical or electrical signal.

[0137] The processor 430 is implemented by hardware and software. The processor 430 may be implemented as one or more CPU chips, cores (for example, a multi-core processor), FPGAs, ASICs, or DSPs. The processor 430 communicates with the ingress ports 410, the receiver unit 420, the transmitter unit 440, the egress ports 450, and the memory 460. The processor 430 includes a coding module 470 (for example, an encoding module 470 or a decoding module 470). The encoding/decoding module 470 implements the embodiments disclosed in this specification, to implement the motion vector prediction method provided in the embodiments of the present invention. For example, the encoding/decoding module 470 performs, processes, or provides various coding operations. Therefore, the encoding/decoding module 470 substantially improves functions of the video coding device 400 and affects transform of the video coding device 400 to a different state. Alternatively, the encoding/decoding module 470 is implemented as an instruction stored in the memory 460 and executed by the processor 430.

[0138] The memory 460 includes one or more disks, tape drives, and solid state drives, and may be used as an overflow data storage device, to store programs when such programs are selectively executed, and to store instructions and data that are read during program execution. The memory 460 may be volatile and/or non-volatile, and may be a read-only memory (ROM), a random access memory (RAM), a ternary content-addressable memory (ternary content-addressable memory, TCAM), and/or a static random access memory (SRAM).

[0139] It should be understood that, in the video encoder 100 and the video decoder 200 in this application, a processing result for a procedure may be output to a next procedure after being further processed. For example, after a procedure such as interpolation filtering, motion vector derivation, or loop filtering, an operation such as clipping or shift is further performed on a processing result for the corresponding procedure.

[0140] For example, a motion vector that is of a control point of a current image block and that is derived based on a motion vector of a neighboring affine coding block may be further processed. This is not limited in this application. For example, a value range of the motion vector is restricted, so that the motion vector is within a specific bit depth. Assuming that an allowed bit depth of the motion vector is bitDepth, a motion vector range is from $-2^{(\text{bitDepth}-1)}$ to $2^{(\text{bitDepth}-1)}-1$, where the symbol " $^{\wedge}$ " represents a power. If bitDepth is 16, the value range is from -32768 to 32767 . If bitDepth is 18, the value range is from -131072 to 131071 . The value range of the

conventional processor or the like.

[0147] The memory 1230 may include a read-only memory (ROM) device or a random access memory (RAM) device. Any other appropriate type of storage device may also be used as the memory 1230. The memory 1230 may include code and data 1231 that are accessed by the processor 1210 by using the bus system 1250. The memory
5 1230 may further include an operating system 1233 and an application program 1235. The application program 1235 includes at least one program that allows the processor 1210 to perform the video encoding or decoding methods described in the embodiments of the present invention (particularly the motion vector prediction method described in the embodiments of the present invention). For example, the application program 1235 may include applications 1 to N, and further includes a video encoding or decoding application (video coding application for short) that
10 performs the video encoding or decoding methods described in the embodiments of the present invention.

[0148] In addition to a data bus, the bus system 1250 may further include a power bus, a control bus, a status signal bus, and the like. However, for clear description, various types of buses in the figure are marked as the bus system 1250.

[0149] Optionally, the coding device 1200 may further include one or more output devices, for example, a
15 display 1270. In an example, the display 1270 may be a touch display that combines a display and a touch unit that operably senses touch input. The display 1270 may be connected to the processor 1210 by using the bus 1250.

[0150] To better understand the technical solutions in the embodiments of the present invention, the following further describes an inter prediction mode, a non-translational motion model, an inherited control point motion vector prediction method, and a constructed control point motion vector prediction method in the embodiments of
20 the present invention.

[0151] (1) Inter prediction mode. In HEVC, two inter prediction modes are used: an advanced motion vector prediction (advanced motion vector prediction, AMVP) mode and a merge (merge) mode.

[0152] In the AMVP mode, spatially or temporally neighboring encoded blocks (denoted as neighboring blocks) of a current block are first traversed. A candidate motion vector list (which may also be referred to as a
25 motion information candidate list) is constructed based on motion information of the neighboring blocks. Then, an optimal motion vector is determined in the candidate motion vector list based on a rate-distortion cost, and candidate motion information with a minimum rate-distortion cost is used as a motion vector predictor (motion vector predictor, MVP) of the current block. Locations and a traversal order of the neighboring blocks are predefined. The rate-distortion cost is calculated according to a formula (1), where J represents the rate-distortion cost RD cost, SAD
30 is a sum of absolute differences (sum of absolute differences, SAD) between an original pixel value and a predicted pixel value obtained through motion estimation by using a candidate motion vector predictor, R represents a bit rate,

compensation subunit of the current coding block is derived according to the following formula (7), where (x, y) is coordinates of a motion compensation subunit relative to the sample in the top-left corner of the current coding block, and W and H are the width and the height of the current coding block respectively.

$$\begin{cases} v_x = \frac{v_{1x}-v_{0x}}{W}x + \frac{v_{2x}-v_{0x}}{H}y + \frac{v_{2x}+v_{0x}-v_{1x}-v_{0x}}{WH}y + v_{0x} \\ v_y = \frac{v_{1y}-v_{0y}}{W}x + \frac{v_{2y}-v_{0y}}{H}y + \frac{v_{2y}+v_{0y}-v_{1y}-v_{0y}}{WH}y + v_{0y} \end{cases} \quad (7)$$

5 **[0167]** A coding block predicted by using an affine transform model may also be referred to as an affine coding block. It can be learned from the foregoing description that the affine transform model is directly related to motion information of a control point of the affine coding block.

[0168] Usually, the motion information of the control point of the affine coding block may be obtained by using an affine transform model-based AMVP mode or an affine transform model-based merge mode. In the affine transform model-based AMVP mode or the affine transform model-based merge mode, the motion information of the control point of the affine coding block may be obtained according to an inherited control point motion vector prediction method or a constructed control point motion vector prediction method. The following further describes the two methods.

10

[0169] (3) Inherited control point motion vector prediction method. In the inherited control point motion vector prediction method, a candidate motion vector of a control point of a current block is determined by using an affine transform model of a neighboring encoded affine coding block of the current block. A parameter quantity (for example, four parameters, six parameters, or eight parameters) of the affine transform model of the affine coding block is the same as that of an affine transform model of the current block.

15

[0170] A current block shown in FIG. 6 is used as an example. Neighboring blocks of the current block are traversed in a specified order, for example, A1->B1->B0->A0->B2, to find an affine coding block in which a neighboring block of the current block is located, and obtain motion information of a control point of the affine coding block. Further, a motion vector (for a merge mode) of a control point or a motion vector predictor (for an AMVP mode) of a control point is derived for the current block by using an affine transform model constructed based on the motion information of the control point of the affine coding block. The order A1->B1->B0->A0->B2 is merely used as an example. Another combination order is also applicable to the embodiments of the present invention. In addition, the neighboring blocks are not limited to A1, B1, B0, A0, and B2. The neighboring block may a sample, or may be a pixel block that is of a preset size and that is obtained according to a particular splitting method, for example, a 4×4 pixel block, a 4×2 pixel block, or a pixel block of another size. This is not limited. The affine coding block is an encoded block (which may also be referred to as a neighboring affine coding block for

20

25

control points of the current block is combined to constitute a 2-tuple, to construct the 4-parameter affine transform model. The two control points may be combined as follows: {CP1, CP4}, {CP2, CP3}, {CP1, CP2}, {CP2, CP4}, {CP1, CP3}, and {CP3, CP4}. For example, a 4-parameter affine transform model constructed by using a 2-tuple including the control points CP1 and CP2 may be denoted as Affine (CP1, CP2).

5 **[0202]** If the 6-parameter affine transform model is used for the current block, motion information of three control points of the current block is combined to constitute a triplet, to construct the 6-parameter affine transform model. The three control points may be combined as follows: {CP1, CP2, CP4}, {CP1, CP2, CP3}, {CP2, CP3, CP4}, and {CP1, CP3, CP4}. For example, a 6-parameter affine transform model constructed by using a triplet including the control points CP1, CP2, and CP3 may be denoted as Affine (CP1, CP2, CP3).

10 **[0203]** If an 8-parameter bilinear model is used for the current block, motion information of four control points of the current block is combined to constitute a quadruple, to construct the 8-parameter bilinear model. An 8-parameter bilinear model constructed by using a quadruple including the control points CP1, CP2, CP3, and CP4 may be denoted as Bilinear (CP1, CP2, CP3, CP4).

[0204] In this specification, for ease of description, a combination of motion information of two control points (or two encoded blocks) is referred to as a 2-tuple for short, a combination of motion information of three control points (or three encoded blocks) is referred to as a triplet for short, and a combination of motion information of four control points (or four encoded blocks) is referred to as a quadruple for short.

[0205] These models are traversed in a preset order. If motion information of a control point corresponding to a combination model is unavailable, it is considered that the model is unavailable. Otherwise, a reference frame index of the model is determined, and a motion vector of the control point is scaled. If motion information of all control points after scaling is consistent, the model is invalid. If all motion information of control points controlling the model is available, and the model is valid, the motion information of the control points used to construct the model is added to a motion information candidate list.

[0206] A method for scaling a control point motion vector is shown in the following formula (14):

$$25 \quad MV_s = \frac{CurPoc - DesPoc}{CurPoc - SrcPoc} \times MV \quad (14)$$

[0207] $CurPoc$ represents a POC number of a current frame, $DesPoc$ represents a POC number of a reference frame of the current block, $SrcPoc$ represents a POC number of a reference frame of a control point, MV_s represents a motion vector obtained through scaling, and MV represents a motion vector of the control point.

[0208] It should be noted that a combination of different control points may alternatively be converted into

$$\begin{cases} vx_1 = \frac{vx_3 - vx_0}{W * W + H * H} W * W + \frac{vy_3 - vy_0}{W * W + H * H} H * W + vx_0 \\ vy_1 = \frac{vy_3 - vy_0}{W * W + H * H} W * W - \frac{vx_3 - vx_0}{W * W + H * H} H * W + vy_0 \end{cases} \quad (18)$$

$$\begin{cases} vx_2 = \frac{vx_3 - vx_1}{W * W + H * H} H * H - \frac{vy_3 - vy_0}{W * W + H * H} H * W + vx_0 \\ vy_2 = \frac{vy_3 - vy_0}{W * W + H * H} W * H + \frac{vx_3 - vx_0}{W * W + H * H} H * H + vy_0 \end{cases} \quad (19)$$

[0215] {CP2, CP4} may be converted into {CP1, CP2} according to the following formula (20), and {CP2,

5 {CP4} may be converted into {CP1, CP2, CP3} according to the following formulas (20) and (21):

$$\begin{cases} vx_0 = -\frac{vy_3 - vy_1}{H} W + vx_1 \\ vy_0 = +\frac{vx_3 - vx_1}{H} W + vy_1 \end{cases} \quad (20)$$

$$\begin{cases} vx_2 = -\frac{vy_3 - vy_1}{H} W + vx_3 \\ vy_2 = +\frac{vx_3 - vx_1}{H} W - vy_3 \end{cases} \quad (21)$$

[0216] {CP3, CP4} may be converted into {CP1, CP2} according to the following formula (22), and {CP3,

CP4} may be converted into {CP1, CP2, CP3} according to the following formulas (22) and (23):

$$\begin{cases} vx_0 = +\frac{vy_3 - vy_2}{W} H + vx_2 \\ vy_0 = -\frac{vx_3 - vx_2}{W} H + vy_2 \end{cases} \quad (22)$$

$$\begin{cases} vx_1 = +\frac{vy_3 - vy_2}{W} H + vx_3 \\ vy_1 = -\frac{vx_3 - vx_2}{W} H + vy_3 \end{cases} \quad (23)$$

[0217] For example, a 6-parameter affine transform model obtained through a combination {CP1, CP2, CP4},

{CP2, CP3, CP4}, or {CP1, CP3, CP4} is represented by the control points {CP1, CP2, CP3}. A conversion method

is: substituting a motion vector and coordinate information of a control point into the foregoing formula (4), to

15 obtain a model parameter; and then substituting coordinate information of {CP1, CP2, CP3} into the foregoing

formula (5), to obtain the motion vector of the control point.

[0218] More directly, conversion may be performed according to the following formulas (24) to (26), where

W represents the width of the current block, and H represents the height of the current block. In the formulas (24) to

(26), $(\begin{smallmatrix} vx_0 & vy_0 \end{smallmatrix})$ represents the motion vector of CP1, and $(\begin{smallmatrix} vx_1 & vy_1 \end{smallmatrix})$ represents the motion vector of CP2,

20 $(\begin{smallmatrix} vx_1 & vy_1 \end{smallmatrix})$ represents the motion vector of CP3, and $(\begin{smallmatrix} vx_3 & vy_3 \end{smallmatrix})$ represents the motion vector of CP4.

prediction method, and the second improved solution may also be referred to as a second motion model-based motion vector prediction method. The following separately describes the two methods.

[0232] (5) First motion model-based motion vector prediction method. In the first motion model-based motion vector prediction method, affine transform models used for different blocks of an image in an image sequence are not limited, in other words, different affine transform models may be used for the different blocks. In a process of encoding and decoding a current block, an affine transform model used for the current block is first determined. The affine transform model used for the current block may be predefined, or may be selected from a plurality of affine transform models based on an actual motion status or an actual requirement of the current block. It is assumed that a $2 \times N$ -parameter affine transform model is used for a neighboring block (also referred to as an affine coding block at an encoder side or an affine decoding block at a decoder side) of the current block, a $2 \times K$ -parameter affine transform model is used for the current block, and $N \neq K$. In this case, motion vectors (candidate motion vectors) of K control points of the current block are obtained through interpolation calculation based on the $2 \times N$ -parameter affine transform model used for the neighboring block.

[0233] The following uses A1 shown in FIG. 10 as an example to describe a process of determining a candidate motion vector of a control point of the current block. The determining process is mainly described from a perspective of the decoder side. In this case, a neighboring block in which A1 is located is the affine decoding block. It may be understood that implementation on the encoder side may be deduced by analogy. In other words, on the encoder side, the neighboring block of the current block is the affine coding block. Details about the implementation are not described in this specification again.

[0234] For example, if a 6-parameter affine transform model is used for the affine decoding block in which A1 is located, and a 4-parameter affine transform model is used for the current block, a motion vector (vx_4, vy_4) of the top-left corner (x_4, y_4) of the affine decoding block, a motion vector (vx_5, vy_5) of the top-right corner (x_5, y_5) of the affine decoding block, and a motion vector (vx_6, vy_6) of the bottom-left corner (x_6, y_6) of the affine decoding block are obtained. According to the following formulas (27) and (28) for the 6-parameter affine transform model, interpolation calculation is separately performed by using a 6-parameter affine transform model constituted by the motion vectors of the foregoing three control points of the affine decoding block, to obtain a motion vector (vx_0, vy_0) of the top-left corner (x_0, y_0) of the current block and a motion vector (vx_1, vy_1) of the top-right corner (x_1, y_1) of the current block:

$$\begin{cases} vx_0 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vx_6 - vx_4)}{y_6 - y_4} \times (y_0 - y_4) \\ vy_0 = vy_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vy_6 - vy_4)}{y_6 - y_4} \times (y_0 - y_4) \end{cases} \quad (27)$$

$$\begin{cases} vx_1 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vy_5 - vy_4)}{y_5 - y_4} \times (y_1 - y_4) \\ vy_1 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vx_5 - vx_4)}{y_5 - y_4} \times (y_1 - y_4) \end{cases} \quad (28)$$

[0235] For another example, if a 4-parameter affine transform model is used for the affine decoding block in which A1 is located, and a 6-parameter affine transform model is used for the current block, a motion vector (vx4, vy4) of the top-left corner (x4, y4) of the affine decoding block and a motion vector (vx5, vy5) of the top-right corner (x5, y5) of the affine decoding block are obtained. In this case, motion vectors of two control points of the affine decoding block are obtained: the motion vector (vx4, vy4) of the top-left control point (x4, y4) and the motion vector (vx5, vy5) of the top-right control point (x5, y5). According to the following formulas (29), (30), and (31) for the 4-parameter affine transform model, interpolation calculation is separately performed by using a 4-parameter affine transform model constituted by the motion vectors of the two control points of the affine decoding block, to obtain a motion vector (vx0, vy0) of the top-left corner (x0, y0) of the current block, a motion vector (vx1, vy1) of the top-right corner (x1, y1) of the current block, and a motion vector (vx2, vy2) of the bottom-left corner (x2, y2) of the current block:

$$\begin{cases} vx_0 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_0 - x_4) - \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (y_0 - y_4) \\ vy_0 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (y_0 - y_4) \end{cases} \quad (29)$$

$$\begin{cases} vx_1 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_1 - x_4) - \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (y_1 - y_4) \\ vy_1 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (y_1 - y_4) \end{cases} \quad (30)$$

$$\begin{cases} vx_2 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_2 - x_4) - \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (y_2 - y_4) \\ vy_2 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_2 - x_4) + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (y_2 - y_4) \end{cases} \quad (31)$$

[0236] It should be noted that the foregoing examples are merely used to explain the technical solutions of the present invention, but are not intended to limit the present invention. In addition, for a case in which other affine transform models are used for the current block and the neighboring block (for example, the 4-parameter affine transform model is used for the current block, and an 8-parameter bilinear model is used for the neighboring block; or the 6-parameter affine transform model is used for the current block, and an 8-parameter bilinear model is used for the neighboring block), refer to the implementation of the foregoing examples. Details are not described herein again.

[0237] It should further be noted that, in this solution, whether a quantity of model parameters of the current block is the same as that of the neighboring block is not limited. Therefore, in some implementation scenarios, the

block. Therefore, motion vectors of two control points of the current block may be obtained according to the inherited control point motion vector prediction method described in "(3)" or the first motion model-based motion vector prediction method described in "(5)", for example, a motion vector (vx4, vy4) of the top-left control point (x4, y4) of the current block and a motion vector (vx5, vy5) of the top-right control point (x5, y5) of the current block.

5 Then, in the phase of reconstructing the current block, the 6-parameter affine transform model needs to be constructed based on the motion vectors of the two control points of the current block.

[0245] For example, based on a motion vector (vx0, vy0) of the top-left control point (x0, y0) of the current block and a motion vector (vx1, vy1) of the top-right control point (x1, y1) of the current block, a motion vector of a third control point may be obtained according to the following formula (40). The motion vector of the third control point is, for example, a motion vector (vx2, vy2) of the bottom-left corner (x2, y2) of the current block.

$$\begin{cases} vx_2 = -\frac{vy_1 - vy_0}{W}H + vx_0 \\ vy_2 = +\frac{vx_1 - vx_0}{H}H + vy_0 \end{cases} \quad (40)$$

[0246] W represents the width of the current block, and H represents the height of the current block.

[0247] Then, the 6-parameter affine model of the current block in the reconstruction phase is obtained by using the motion vector (vx0, vy0) of the top-left control point (x0, y0) of the current block, the motion vector (vx1, vy1) of the top-right control point (x1, y1) of the current block, and the motion vector (vx2, vy2) of the bottom-left control point (x2, y2) of the current block. A formula for the 6-parameter affine model is shown in the following formula (37):

$$\begin{cases} vx = \frac{vx_1 - vx_0}{W}x + \frac{vx_2 - vx_0}{H}y + vx_0 \\ vy = \frac{vy_1 - vy_0}{W}x + \frac{vy_2 - vy_0}{H}y + vy_0 \end{cases} \quad (37)$$

[0248] Then, coordinates $(x_{(i,j)}, y_{(i,j)})$ of the center point of each subblock (or each motion compensation unit) of the current block relative to the top-left corner (or another reference point) of the current block are substituted into the foregoing formula (37) to obtain motion information of the center point of each subblock (or each motion compensation unit), to subsequently reconstruct each subblock.

[0249] It should be noted that the foregoing example is merely used to explain the technical solutions of the present invention, but is not intended to limit the present invention. In addition, for a case in which another affine transform model (for example, the 6-parameter affine transform model or an 8-parameter bilinear model) is used for the current block in the parsing phase, refer to the implementation of the foregoing example. Details are not

(or referred to as a control point motion vector predictor candidate list) based on the AMVP mode may also be constructed by using a first motion model-based motion vector prediction method and/or a constructed control point motion vector prediction method. In another embodiment, a candidate motion vector list (or referred to as a control point motion vector predictor candidate list) based on the AMVP mode may be constructed by using an inherited
5 control point motion vector prediction method and/or a constructed control point motion vector prediction method. Control point motion vector predictors in the list may include two candidate control point motion vectors (for example, when a 4-parameter affine transform model is used for a current block), or include three candidate control point motion vectors (for example, when a 6-parameter affine transform model is used for a current block), or include four candidate control point motion vectors (for example, when an 8-parameter bilinear model is used for a
10 current block).

[0258] In a possible application scenario, the control point motion vector predictor candidate list may further be pruned and sorted according to a particular rule, and may be truncated or padded to obtain control point motion vector predictor candidates of a particular quantity.

[0259] Then, on an encoder side, an encoder (for example, the video encoder 100) obtains a motion vector of
15 each motion compensation subunit of the current coding block by using each control point motion vector predictor in the control point motion vector predictor candidate list and according to the foregoing formula (3), (5), or (7). Further, the encoder obtains a pixel value of a corresponding location in a reference frame to which the motion vector of each motion compensation subunit points, and uses the pixel value as a pixel predictor of the motion compensation subunit to perform affine transform model-based motion compensation. An average difference
20 between an original value and a predictor of each sample in the current coding block is calculated. A control point motion vector predictor corresponding to a minimum average difference is selected as an optimal control point motion vector predictor, and used as motion vector predictors of two, three, or four control points of the current coding block. In addition, on the encoder side, a control point motion vector predictor may further be used as a search start point to perform motion search within a specific search range to obtain a control point motion vector
25 (control point motion vectors, CPMV), and a difference (control point motion vectors differences, CPMVD) between the control point motion vector and the control point motion vector predictor is calculated. Then, the encoder encodes an index value indicating a location of the control point motion vector predictor in the control point motion vector predictor candidate list and the CPMVD into a bitstream, and transfers the bitstream to a decoder side.

[0260] On the decoder side, a decoder (for example, the video decoder 200) parses the bitstream to obtain the
30 index value and the control point motion vector difference (CPMVD), determines the control point motion vector predictor (control point motion vectors predictor, CPMVP) in the control point motion vector predictor candidate list

based on the index value, and adds the CPMVP and the CPMVD to obtain the control point motion vector.

[0261] The following describes the affine transform model-based merge mode.

[0262] For the affine transform model-based merge mode, in an embodiment, a candidate motion vector list (or referred to as a control point motion vector merge candidate list) for the merge mode may be constructed by using an inherited control point motion vector prediction method and/or a constructed control point motion vector prediction method. In another embodiment, a candidate motion vector list (or referred to as a control point motion vector merge candidate list) for the merge mode may alternatively be constructed by using a first motion model-based motion vector prediction method and/or a constructed control point motion vector prediction method.

[0263] In a possible application scenario, the control point motion vector merge candidate list may further be pruned and sorted according to a particular rule, and may be truncated or padded to obtain control point motion vector candidates of a particular quantity.

[0264] Then, on an encoder side, an encoder (for example, the video encoder 100) obtains a motion vector of each motion compensation subunit (sample or pixel block with a size of $M \times N$ and obtained through splitting according to a particular method) of a current coding block by using each control point motion vector in the merge candidate list and according to the foregoing formula (3), (5), or (7). Further, the encoder obtains a pixel value of a location in a reference frame to which the motion vector of each motion compensation subunit points, and uses the pixel value as a pixel predictor of the motion compensation subunit to perform affine motion compensation. An average difference between an original value and a predictor of each sample in the current coding block is calculated. A control point motion vector corresponding to a minimum average difference is selected as motion vectors of two, three, or four control points of the current coding block. An index number indicating a location of the control point motion vector in the candidate list is encoded into a bitstream and sent to a decoder side.

[0265] On the decoder side, a decoder (for example, the video decoder 200) parses the bitstream to obtain the index value, and determines the control point motion vector (control point motion vectors, CPMVP) in the control point motion vector merge candidate list based on the index value.

[0266] In addition, it should be noted that, in the embodiments of the present invention, "at least one" means one or more, and "a plurality of" means two or more than two. The term "and/or" describes an association relationship for describing associated objects and represents that three relationships may exist. For example, A and/or B may indicate the following cases: Only A exists, both A and B exist, and only B exists, where A and B may be singular or plural. The character "/" generally indicates an "or" relationship between the associated objects. "At least one of the following items (pieces)" or a similar expression thereof indicates any combination of these items, including a single item (piece) or any combination of a plurality of items (pieces). For example, at least one of a, b,

AMVP mode may be a quadruple list. The quadruple list includes one or more quadruples used to construct the 8-parameter bilinear model.

[0278] In a possible application scenario, the candidate motion vector 2-tuple/triplet/quadruple list may be pruned and sorted according to a particular rule, and may be truncated or padded to obtain candidate motion vector

5 candidates of a particular quantity.

[0279] For the first motion model-based motion vector prediction method, for example, as shown in FIG. 10, neighboring blocks of the current block may be traversed in an order of A1→B1→B0→A0→B2 in FIG. 10, to find an affine decoding block in which a neighboring block is located (for example, an affine decoding block in which A1 is located in FIG. 10). An affine transform model of the affine decoding block is constructed by using a control

10 point of the affine decoding block, and then the candidate motion vector (for example, a candidate motion vector 2-tuple/triplet/quadruple) of the control point of the current block is derived by using the affine transform model of the affine decoding block, and added to the candidate motion vector list corresponding to the AMVP mode. It should be noted that another search order may also be applicable to the embodiments of the present invention. Details are not described herein.

[0280] It should be noted that, when there are a plurality of neighboring blocks, in other words, when the current block has a plurality of neighboring affine decoding blocks, in a possible embodiment, both the encoder side and the decoder side can first obtain a candidate motion vector of a control point of the current block by using an affine decoding block whose quantity of model parameters is the same as that of the current block, and add the obtained candidate motion vector to the candidate motion vector list corresponding to the AMVP mode. Then, a

20 candidate motion vector of the control point of the current block may be obtained by using an affine decoding block whose quantity of model parameters is different from that of the current block, and added to the candidate motion vector list corresponding to the AMVP mode. In this way, the candidate motion vector that is of the control point of the current block and that is obtained by using the affine decoding block whose quantity of model parameters is the same as that of the current block is located at a front location of the list. This design helps reduce a quantity of bits

25 transmitted in the bitstream.

[0281] FIG. 10 is used as an example. It is assumed that a parameter model of the current decoding block is the 4-parameter affine transform model. After the neighboring blocks of the current block are traversed, it is determined that the 4-parameter affine transform model is used for an affine decoding block in which B1 is located and the 6-parameter affine transform model is used for the affine decoding block in which A1 is located. In this case,

30 motion vectors of two control points of the current block may be first derived by using the affine decoding block in which B1 is located, and added to the list. Then, motion vectors of two control points of the current block are

derived by using the affine decoding block in which A1 is located, and added to the list.

[0282] Alternatively, it is assumed that a parameter model of the current decoding block is the 6-parameter affine transform model. After the neighboring blocks of the current block are traversed, it is determined that the 6-parameter affine transform model is used for the affine decoding block in which A1 is located and the 4-parameter affine transform model is used for an affine decoding block in which B1 is located. In this case, motion vectors of three control points of the current block may be first derived by using the affine decoding block in which A1 is located, and added to the list. Then, motion vectors of three control points of the current block are derived by using the affine decoding block in which B1 is located, and added to the list.

[0283] It should be noted that the technical solutions of the present invention are not limited to the foregoing examples, and another neighboring block, motion model, and search order may also be applicable to the present invention. Details are not described herein.

[0284] In the step 602a, affine transform models used for different blocks are not limited. In other words, a quantity of parameters of an affine transform model used for the current block may be different from or the same as that of an affine decoding block. In an embodiment, the affine transform model used for the current block may be determined by parsing the bitstream. In other words, in this case, the bitstream includes indication information of the affine transform model of the current block. In an embodiment, the affine transform model used for the current block may be pre-configured. In an embodiment, the affine transform model used for the current block may be selected from a plurality of affine transform models based on an actual motion status or an actual requirement of the current block.

[0285] Some content about obtaining the candidate motion vector of the control point of the current block by using the constructed control point motion vector prediction method has been described in detail in the foregoing "(4)". For brevity of this specification, details are not described herein again.

[0286] Some content about obtaining the candidate motion vector of the control point of the current block by using the first motion model-based motion vector prediction method has been described in detail in the foregoing "(5)". For brevity of this specification, details are not described herein again.

[0287] It should be noted that, in some embodiments in which the first motion model-based motion vector prediction method is used, in a process in which the decoder side derives the candidate motion vector of the control point of the current block, flag information (flag) of an affine transform model of an affine decoding block may need to be obtained. The flag is pre-stored locally on the decoder side, and is used to indicate the affine transform model that is of the affine decoding block and that is actually used to predict a subblock of the affine decoding block.

[0288] For example, in an application scenario, when the decoder side determines, by identifying the flag of

the affine decoding block, that a quantity of model parameters of the affine transform model actually used for the affine decoding block is different from (or the same as) that of the affine transform model used for the current block, the decoder side is triggered to derive the candidate motion vector of the control point of the current block by using the affine transform model actually used for the affine decoding block.

5 **[0289]** For example, when the 4-parameter affine transform model is used for the current block, if the decoder side identifies the flag of the affine decoding block, and determines that the quantity of model parameters of the affine transform model actually used for the affine decoding block is different from that of the affine transform model used for the current block, for example, the 6-parameter affine transform model is used for the affine decoding block, the decoder side obtains motion vectors of three control points of the affine decoding block: a
10 motion vector (vx4, vy4) of the top-left corner (x4, y4), a motion vector (vx5, vy5) of the top-right corner (x5, y5), and a motion vector (vx6, vy6) of the bottom-left corner (x6, y6). Based on the 6-parameter affine transform model constituted by the three control points of the affine decoding block, candidate motion vectors of the top-left and top-right control points of the current block are respectively derived according to the formulas (27) and (28) for the 6-parameter affine transform model.

15 **[0290]** For another example, when the 4-parameter affine transform model is used for the current block, if the decoder side identifies the flag of the affine decoding block, and determines that the quantity of model parameters of the affine transform model actually used for the affine decoding block is the same as that of the affine transform model used for the current block, for example, the 4-parameter affine transform model is also used for the affine decoding block, the decoder side obtains motion vectors of two control points of the affine decoding block: a motion
20 vector (vx4, vy4) of the top-left control point (x4, y4) and a motion vector (vx5, vy5) of the top-right control point (x5, y5). Based on the 4-parameter affine transform model constituted by the two control points of the affine decoding block, candidate motion vectors of the top-left and top-right control points of the current block are respectively derived according to the formulas (32) and (33) for the 4-parameter affine transform model.

25 **[0291]** It should be noted that, in some other embodiments in which the first motion model-based motion vector prediction method is used, in a process in which the decoder side derives the candidate motion vector of the control point of the current block, a flag of an affine transform model of an affine decoding block may not be required.

30 **[0292]** For example, in an application scenario, after the decoder side determines the affine transform model used for the current block, the decoder side obtains control points of a specific quantity (the specific quantity is the same as or different from a quantity of control points of the current block) of the affine decoding block, constructs the affine transform model by using the control points of the specific quantity of the affine decoding block, and then

derives the candidate motion vector of the control point of the current block by using the affine transform model.

[0293] For example, when the 4-parameter affine transform model is used for the current block, the decoder side does not determine the affine transform model actually used for the affine decoding block (the affine transform model actually used for the affine decoding block may be the 4-parameter affine transform model, the 6-parameter affine transform model, or the 8-parameter bilinear model), but directly obtains motion vectors of two control points of the affine decoding block: a motion vector (vx_4, vy_4) of the top-left control point (x_4, y_4) and a motion vector (vx_5, vy_5) of the top-right control point (x_5, y_5). Based on the 4-parameter affine model constituted by the two control points of the affine decoding block, motion vectors of the top-left and top-right control points of the current block are respectively derived according to the formulas (32) and (33) for the 4-parameter affine model.

[0294] It should be noted that the technical solutions of the present invention are not limited to the foregoing examples, and another control point, motion model, candidate location, and search order may also be applicable to the present invention. Details are not described herein.

[0295] Step 603a: Determine an optimal motion vector predictor of the control point based on an index value.

[0296] Specifically, the index value of the candidate motion vector list is obtained by parsing the bitstream, and the optimal motion vector predictor of the control point is determined, based on the index value, in the candidate motion vector list constructed in the step 602a.

[0297] For example, if the 4-parameter affine motion model is used for the current block, the index value is obtained through parsing, and optimal motion vector predictors of two control points are determined in the candidate motion vector 2-tuple list based on the index value.

[0298] For another example, if the 6-parameter affine motion model is used for the current block, the index value is obtained through parsing, and optimal motion vector predictors of three control points are determined in the candidate motion vector triplet list based on the index value.

[0299] For another example, if the 8-parameter bilinear model is used for the current block, the index value is obtained through parsing, and optimal motion vector predictors of four control points are determined in the candidate motion vector quadruple list based on the index value.

[0300] Step 604a: Determine an actual motion vector of the control point based on a motion vector difference.

[0301] Specifically, the motion vector difference of the control point is obtained by parsing the bitstream, and then the motion vector of the control point is obtained based on the motion vector difference of the control point and the optimal motion vector predictor that is of the control point and that is determined in the step 603a.

[0302] For example, if the 4-parameter affine motion model is used for the current block, motion vector differences of two control points of the current block are obtained by decoding the bitstream. For example, a motion

vector difference of the top-left control point and a motion vector difference of the top-right control point may be obtained by decoding the bitstream. Then, a motion vector difference and a motion vector predictor of each control point are added to obtain an actual motion vector of the control point. In other words, the motion vectors of the top-left and top-right control points of the current block are obtained.

5 **[0303]** For another example, if the 6-parameter affine motion model is used for the current block, motion vector differences of three control points of the current block are obtained by decoding the bitstream. For example, a motion vector difference of the top-left control point, a motion vector difference of the top-right control point, and a motion vector difference of the bottom-left control point may be obtained by decoding the bitstream. Then, a motion vector difference and a motion vector predictor of each control point are added to obtain an actual motion vector of
10 the control point. In other words, motion vectors of the top-left, top-right, and bottom-left control points of the current block are obtained.

[0304] It should be noted that, in this embodiment of the present invention, another affine motion model and another control point location may also be used. Details are not described herein.

[0305] Step 605a: Obtain a motion vector of each subblock of the current block based on the affine transform
15 model used for the current block.

[0306] For each $M \times N$ subblock in a $P \times Q$ current block (one subblock may be equivalent to one motion compensation unit, and at least one of the width or the height of the $M \times N$ subblock is less than the width or the height of the current block), motion information of a sample at a preset location in the motion compensation unit may be used to represent motion information of all samples in the motion compensation unit. Assuming that a size
20 of the motion compensation unit is $M \times N$, the sample at the preset location may be the center point ($M/2$, $N/2$), the top-left sample (0, 0), the top-right sample ($M-1$, 0), or a sample at another location of the motion compensation unit.

[0307] The following uses the center point of the motion compensation unit as an example for description. Refer to FIG. 11A and FIG. 11B.

25 **[0308]** FIG. 11A shows an example of a current block and a motion compensation unit of the current block. Each small box in the figure represents one motion compensation unit. In the figure, a specification of each motion compensation unit is 4×4 , and a gray point in each motion compensation unit represents the center point of the motion compensation unit. In FIG. 11A, V_0 represents a motion vector of the top-left control point of the current block, V_1 represents a motion vector of the top-right control point of the current block, and V_2 represents a motion
30 vector of the bottom-left control point of the current block.

[0309] FIG. 11B shows an example of another current block and a motion compensation unit of the current

block. Each small box in the figure represents one motion compensation unit. In the figure, a specification of each motion compensation unit is 8×8, and a gray point in each motion compensation unit represents the center point of the motion compensation unit. In FIG. 11B, V0 represents a motion vector of the top-left control point of the current block, V1 represents a motion vector of the top-right control point of the current block, and V2 represents a motion vector of the bottom-left control point of the current block.

[0310] Coordinates of the center point of the motion compensation unit relative to the top-left pixel of the current block may be calculated according to the following formula (38):

$$\begin{cases} x_{(i,j)} = M \times i + \frac{M}{2}, i = 0, 1, \dots \\ y_{(i,j)} = N \times j + \frac{N}{2}, j = 0, 1, \dots \end{cases} \quad (38)$$

[0311] In the foregoing formula, i is the ith motion compensation unit (from left to right) in a horizontal direction, j is the jth motion compensation unit (from top to bottom) in a vertical direction, and ($x_{(i,j)}$, $y_{(i,j)}$) indicates coordinates of the center point of the (i, j)th motion compensation unit relative to a pixel at the top-left control point of a current affine decoding block.

[0312] If the 6-parameter affine motion model is used for the current affine decoding block, ($x_{(i,j)}$, $y_{(i,j)}$) is substituted into the following formula (37) for the 6-parameter affine motion model to obtain a motion vector of the center point of each motion compensation unit, and the obtained motion vector is used as motion vectors ($vx_{(i,j)}$, $vy_{(i,j)}$) of all samples in the motion compensation unit:

$$\begin{cases} vx = \frac{vx_1 - vx_0}{W} x + \frac{vy_1 - vy_0}{H} y + vx_0 \\ vy = \frac{vy_1 - vy_0}{W} x + \frac{vx_1 - vx_0}{H} y + vy_0 \end{cases} \quad (37)$$

[0313] If the 4-parameter affine motion model is used for the current affine decoding block, ($x_{(i,j)}$, $y_{(i,j)}$) is substituted into the following formula (39) for the 4-parameter affine motion model to obtain a motion vector of the center point of each motion compensation unit, and the obtained motion vector is used as motion vectors ($vx_{(i,j)}$, $vy_{(i,j)}$) of all samples in the motion compensation unit:

$$\begin{cases} vx = \frac{vx_1 - vx_0}{W} x - \frac{vy_1 - vy_0}{W} y + vx_0 \\ vy = \frac{vy_1 - vy_0}{W} x + \frac{vx_1 - vx_0}{W} y + vy_0 \end{cases} \quad (39)$$

[0314] Step 606a: Perform motion compensation for each subblock based on the determined motion vector of the subblock, to obtain a pixel predictor of the subblock.

[0315] Step 602b: Construct a candidate motion vector list for the affine transform model-based merge mode.

[0316] In some specific embodiments of the present invention, a candidate motion vector of a control point of the current block may alternatively be obtained by using the first motion model-based motion vector prediction method, and added to the candidate motion vector list corresponding to the merge mode.

5 **[0317]** In some other specific embodiments of the present invention, a candidate motion vector of a control point of the current block may alternatively be obtained separately by using the first motion model-based motion vector prediction method and a constructed control point motion vector prediction method, and added to the candidate motion vector list corresponding to the merge mode.

[0318] Likewise, for the candidate motion vector list corresponding to the merge mode, if a 4-parameter affine transform model is used for the current block, the candidate motion vector list may be a 2-tuple list. The 2-tuple list includes one or more 2-tuples used to construct the 4-parameter affine transform model.

10 **[0319]** If a 6-parameter affine transform model is used for the current block, the candidate motion vector list may be a triplet list. The triplet list includes one or more triplets used to construct the 6-parameter affine transform model.

15 **[0320]** If an 8-parameter bilinear model is used for the current block, the candidate motion vector list may be a quadruple list. The quadruple list includes one or more quadruples used to construct the 8-parameter bilinear model.

[0321] In a possible application scenario, the candidate motion vector 2-tuple/triplet/quadruple list may be pruned and sorted according to a particular rule, and may be truncated or padded to obtain candidate motion vector candidates of a particular quantity.

20 **[0322]** Likewise, for the first motion model-based motion vector prediction method, for example, as shown in FIG. 10, neighboring blocks of the current block may be traversed in an order of $A1 \rightarrow B1 \rightarrow B0 \rightarrow A0 \rightarrow B2$ in FIG. 10, to find an affine decoding block in which a neighboring block is located. An affine transform model of the affine decoding block is constructed by using a control point of the affine decoding block, and then the candidate motion vector (for example, a candidate motion vector 2-tuple/triplet/quadruple) of the control point of the current block is derived by using the affine transform model of the affine decoding block, and added to the candidate motion vector list corresponding to the merge mode. It should be noted that another search order may also be applicable to the embodiments of the present invention. Details are not described herein.

25 **[0323]** Specifically, in the foregoing traversal process, if the candidate motion vector list is empty, candidate motion information of the control point is added to the candidate list. Otherwise, motion information in the candidate motion vector list is sequentially traversed, and whether motion information that is the same as the

30

candidate motion information of the control point exists in the candidate motion vector list is checked. If the motion information that is the same as the candidate motion information of the control point does not exist in the candidate motion vector list, the candidate motion information of the control point is added to the candidate motion vector list.

[0324] To determine whether two pieces of candidate motion information are the same, it is necessary to sequentially determine whether forward reference frames, backward reference frames, horizontal and vertical components of each forward motion vector, and horizontal and vertical components of each backward motion vector in the two pieces of candidate motion information are the same. The two pieces of candidate motion information are considered as different only when all these elements are different.

[0325] If a quantity of pieces of motion information in the candidate motion vector list reaches a maximum list length, the candidate list construction is completed; otherwise, a next neighboring block is traversed.

[0326] Some content about obtaining the candidate motion vector of the control point of the current block by using the constructed control point motion vector prediction method has been described in detail in the foregoing "(4)". For brevity of this specification, details are not described herein again.

[0327] Some content about obtaining the candidate motion vector of the control point of the current block by using the first motion model-based motion vector prediction method has been described in detail in the foregoing "(5)". For brevity of this specification, details are not described herein again.

[0328] It should be noted that, in some embodiments in which the first motion model-based motion vector prediction method is used, for the affine transform model-based merge mode, affine transform models such as the 4-parameter affine transform model, the 6-parameter affine transform model, and the 8-parameter bilinear model may alternatively not be distinguished for different blocks in an image, in other words, affine transform models with a same quantity of parameters may be used for the different blocks.

[0329] For example, the 6-parameter affine transform model is used for all blocks in the image. A1 in FIG. 10 is used as an example. Motion vectors of three control points of an affine decoding block in which A1 is located are obtained: a motion vector (vx_4, vy_4) of the top-left control point (x_4, y_4), a motion vector (vx_5, vy_5) of the top-right control point (x_5, y_5), and a motion vector (vx_6, vy_6) of the bottom-left control point (x_6, y_6). Then, based on the 6-parameter affine model constituted by the three control points of the neighboring affine decoding block, motion vectors of the top-left control point, the top-right control point, and the bottom-left control point of the current block are respectively derived according to the formulas (34), (35), and (36).

[0330] It should be noted that the technical solutions of the present invention are not limited to the foregoing example, and another control point, motion model, candidate location, and search order may also be applicable to the present invention. Details are not described herein.

[0331] Step 603b: Determine a motion vector of the control point based on an index value.

[0332] Specifically, the index value of the candidate motion vector list is obtained by parsing the bitstream, and the actual motion vector of the control point is determined, based on the index value, in the candidate motion vector list constructed in the step 602b.

5 **[0333]** For example, if the 4-parameter affine motion model is used for the current block, the index value is obtained through parsing, and motion vectors of two control points are determined in the candidate motion vector 2-tuple list based on the index value.

[0334] For another example, if the 6-parameter affine motion model is used for the current block, the index value is obtained through parsing, and motion vectors of three control points are determined in the candidate motion vector triplet list based on the index value.

10 **[0335]** For another example, if the 8-parameter bilinear model is used for the current block, the index value is obtained through parsing, and motion vectors of four control points are determined in the candidate motion vector quadruple list based on the index value.

[0336] Step 604b: Obtain a motion vector of each subblock of the current block based on an affine transform model used for the current block. For detailed implementation of this step, refer to descriptions of the step 605a. For brevity of this specification, details are not described herein again.

15 **[0337]** Step 605b: Perform motion compensation for each subblock based on the corresponding motion vector, to obtain a pixel predictor of the subblock.

[0338] It can be learned that, in this embodiment of the present invention, the decoder side uses the first motion model-based motion vector prediction method in a process of predicting the current block. In this way, an affine transform model of a neighboring block can be used to construct the affine transform model of the current block in a phase of parsing the current block (for example, in a phase of constructing the candidate motion vector list for the AMVP mode or the merge mode). The affine transform models of the two blocks may be different or the same. The affine transform model of the current block better satisfies the actual motion status/actual requirement of the current block. Therefore, this solution can improve the coding efficiency and the accuracy in predicting the current block, and satisfy the user requirement.

20 **[0339]** Refer to FIG. 12. Based on a design solution of a second motion model-based motion vector prediction method, an embodiment of the present invention provides another motion vector prediction method. The method may be performed by a video decoder 200, and specifically, may be performed by an inter predictor 210 of the video decoder 200. The video decoder 200 may perform, based on a video data stream having a plurality of video frames, a part or all of the following steps to predict motion information of each subblock of a current decoding block

25

30

[0351] Some content about obtaining the candidate motion vector of the control point of the current block by using the inherited control point motion vector prediction method has been described in detail in the foregoing "(3)". For brevity of this specification, details are not described herein again.

[0352] Some content about obtaining the candidate motion vector of the control point of the current block by using the constructed control point motion vector prediction method has been described in detail in the foregoing "(4)". For brevity of this specification, details are not described herein again.

[0353] Some content about obtaining the candidate motion vector of the control point of the current block by using the first motion model-based motion vector prediction method has been described in detail in the foregoing "(5)" and in the step 602a in the embodiment in FIG. 9. For brevity of this specification, details are not described herein again.

[0354] For example, if a 4-parameter affine transform model is used for the current block, the candidate motion vector list for the AMVP mode may be a 2-tuple list. The 2-tuple list includes one or more 2-tuples used to construct the 4-parameter affine transform model.

[0355] If a 6-parameter affine transform model is used for the current block, the candidate motion vector list for the AMVP mode may be a triplet list. The triplet list includes one or more triplets used to construct the 6-parameter affine transform model.

[0356] If an 8-parameter bilinear model is used for the current block, the candidate motion vector list for the AMVP mode may be a quadruple list. The quadruple list includes one or more quadruples used to construct the 8-parameter bilinear model.

[0357] In a possible application scenario, the candidate motion vector 2-tuple/triplet/quadruple list may further be pruned and sorted according to a particular rule, and may be truncated or padded to obtain candidate motion vector candidates of a particular quantity.

[0358] Step 703a: Determine an optimal motion vector predictor of the control point based on an index value. For specific content, refer to related descriptions in the step 603a in the embodiment in FIG. 9. Details are not described herein again.

[0359] Step 704a: Determine motion vectors of three control points of the current block based on motion vector differences.

[0360] Specifically, a motion vector difference of the control point is obtained by parsing the bitstream, and then a motion vector of the control point is obtained based on the motion vector difference of the control point and the optimal motion vector predictor that is of the control point and that is determined in the step 703a. Then, the motion vectors of the three control points of the current block are determined based on obtained motion

vectors of the control points.

[0361] For example, if the candidate motion vector list constructed by the decoder side in step 702a is the 2-tuple list, the index value is obtained through parsing in the step 703a, and motion vector predictors (MVP) of two control points (namely, a 2-tuple) are determined in the candidate motion vector list based on the index value.

5 Motion vector differences (MVD) of the two control points of the current block are obtained by parsing the bitstream in the step 704a. Then, motion vectors (MV) of the two control points are respectively obtained based on the MVPs and the MVDs of the two control points. The motion vectors of the two control points are, for example, a motion vector (vx_0, vy_0) of the top-left control point (x_0, y_0) of the current block and a motion vector (vx_1, vy_1) of the top-right control point (x_1, y_1) of the current block. Then, the 4-parameter affine transform model is constructed
10 based on the motion vectors of the two control points of the current block. A motion vector of a third control point is obtained according to a formula (40) for the 4-parameter affine transform model. The motion vector of the third control point is, for example, a motion vector (vx_2, vy_2) of the bottom-left corner (x_2, y_2) of the current block. In this way, the motion vectors of the top-left control point, the top-right control point, and the bottom-left control point of the current block are determined.

15 **[0362]** For another example, if the candidate motion vector list constructed by the decoder side in step 702a is the triplet list, the index value is obtained through parsing in the step 703a, and motion vector predictors (MVP) of three control points (namely, a triplet) are determined in the candidate motion vector list based on the index value. Motion vector differences (MVD) of the three control points of the current block are obtained by parsing the bitstream in the step 704a. Then, motion vectors (MV) of the three control points are respectively obtained based on
20 the MVPs and the MVDs of the three control points. The motion vectors of the three control points are, for example, a motion vector (vx_0, vy_0) of the top-left control point (x_0, y_0) of the current block, a motion vector (vx_1, vy_1) of the top-right control point (x_1, y_1) of the current block, and a motion vector (vx_2, vy_2) of the bottom-left corner (x_2, y_2) of the current block.

[0363] In this way, the motion vectors of the top-left control point, the top-right control point, and the
25 bottom-left control point of the current block are determined.

[0364] For another example, if the candidate motion vector list constructed by the decoder side in step 702a is the quadruple list, the index value is obtained through parsing in the step 703a, and motion vector predictors (MVP) of four control points (namely, a quadruple) are determined in the candidate motion vector list based on the index value. Motion vector differences (MVD) of the four control points of the current block are obtained by parsing the
30 bitstream in the step 704a. Then, motion vectors (MV) of the four control points are respectively obtained based on the MVPs and the MVDs of the four control points. The motion vectors of the four control points are,

for example, a motion vector (vx0, vy0) of the top-left control point (x0, y0) of the current block, a motion vector (vx1, vy1) of the top-right control point (x1, y1) of the current block, a motion vector (vx2, vy2) of the bottom-left corner (x2, y2) of the current block, and a motion vector (vx3, vy3) of the bottom-right corner (x3, vy3) of the current block. Then, the decoder side may use only motion vectors of the top-left control point, the top-right control point, and the bottom-left control point of the current block.

[0365] It should be noted that the technical solutions of the present invention are not limited to the foregoing examples, and another control point and motion model may also be applicable to the present invention. Details are not described herein.

[0366] Step 705a: Obtain a motion vector of each subblock based on the three control points of the current block and by using the 6-parameter affine transform model.

[0367] Specifically, the motion vectors of the three control points of the current block have been determined in the step 704a. Therefore, the 6-parameter affine transform model may be constructed based on the motion vectors of the three control points of the current block, and the motion vector of each subblock is obtained by using the 6-parameter affine transform model.

[0368] For example, the motion vectors of the three control points are, for example, the motion vector (vx0, vy0) of the top-left control point (x0, y0) of the current block, the motion vector (vx1, vy1) of the top-right control point (x1, y1) of the current block, and the motion vector (vx2, vy2) of the bottom-left control point (x2, y2) of the current block. In this case, the 6-parameter affine model of the current block in a reconstruction phase is obtained by using the motion vector (vx0, vy0) of the top-left control point (x0, y0) of the current block, the motion vector (vx1, vy1) of the top-right control point (x1, y1) of the current block, and the motion vector (vx2, vy2) of the bottom-left corner (x2, y2) of the current block. A formula for the 6-parameter affine model is shown in the formula (37).

[0369] Then, coordinates $(x(i,j), y(i,j))$ of a sample at a preset location in each subblock (or each motion compensation unit) of the current block relative to the top-left corner (or another reference point) of the current block are substituted into the foregoing formula (37) to obtain the motion vector of each subblock. The sample at the preset location may be the center point of each subblock (or each motion compensation unit). Coordinates $(x(i,j), y(i,j))$ of the center point of each subblock (or each motion compensation unit) relative to the top-left pixel of the current block may be calculated according to the following formula (38). For specific content, refer to related descriptions in the embodiment in FIG. 11A and the embodiment in FIG. 11B. Details are not described herein again.

- [0370]** Step 706a: Perform motion compensation for each subblock based on the corresponding motion vector, to obtain a pixel predictor of the subblock.
- [0371]** Step 702b: Construct a candidate motion vector list for the affine transform-based merge mode.
- [0372]** Likewise, in this embodiment of the present invention, affine transform models used for different blocks of an image in an image sequence are not limited, in other words, different affine transform models may be used for the different blocks.
- [0373]** In a specific embodiment, a candidate motion vector of a control point of the current block may be obtained by using an inherited control point motion vector prediction method, and added to the candidate motion vector list corresponding to the merge mode.
- [0374]** In a specific embodiment, a candidate motion vector of a control point of the current block may be obtained by using a first motion model-based motion vector prediction method, and added to the candidate motion vector list corresponding to the merge mode.
- [0375]** In a specific embodiment, a candidate motion vector of a control point of the current block may be obtained by using a constructed control point motion vector prediction method, and added to the candidate motion vector list corresponding to the merge mode.
- [0376]** In some other specific embodiments, a candidate motion vector of a control point of the current block may alternatively be obtained separately by using any two of an inherited control point motion vector prediction method, the second motion model-based motion vector prediction method, or a constructed control point motion vector prediction method, and added to the candidate motion vector list corresponding to the merge mode.
- [0377]** In some other specific embodiments, a candidate motion vector of a control point of the current block may alternatively be obtained separately by using an inherited control point motion vector prediction method, the second motion model-based motion vector prediction method, and a constructed control point motion vector prediction method, and added to the candidate motion vector list corresponding to the merge mode.
- [0378]** Some content about obtaining the candidate motion vector of the control point of the current block by using the inherited control point motion vector prediction method has been described in detail in the foregoing "(3)". For brevity of this specification, details are not described herein again.
- [0379]** Some content about obtaining the candidate motion vector of the control point of the current block by using the constructed control point motion vector prediction method has been described in detail in the foregoing "(4)". For brevity of this specification, details are not described herein again.
- [0380]** Some content about obtaining the candidate motion vector of the control point of the current block by using the first motion model-based motion vector prediction method has been described in detail in the foregoing

"(5)" and in the step 602a in the embodiment in FIG. 9. For brevity of this specification, details are not described herein again.

[0381] It should be noted that, in still some other embodiments, for the affine transform model-based merge mode, the candidate motion vector list established by the decoder side may be a candidate motion vector 2-tuple/triplet/quadruple list. In addition, the candidate motion vector 2-tuple/triplet/quadruple list may further be pruned and sorted according to a particular rule, and may be truncated or padded to obtain candidate motion vector candidates of a particular quantity.

[0382] It should be noted that, in still some other embodiments, for the affine transform model-based merge mode, affine transform models such as a 4-parameter affine transform model, a 6-parameter affine transform model, and an 8-parameter bilinear model may alternatively not be distinguished for different blocks in an image, in other words, affine transform models with a same quantity of parameters may be used for the different blocks.

[0383] Step 703b: Obtain a motion vector of the control point based on an index value. Specifically, the index value of the candidate motion vector list is obtained by parsing the bitstream, and the actual motion vector of the control point is determined, based on the index value, in the candidate motion vector list constructed in the step 702b. For a specific implementation of this step, refer to related descriptions in the step 603b in the embodiment in FIG. 9. Details are not described herein again.

[0384] Step 704b: Determine motion vectors of three control points of the current block based on obtained motion vectors of the control points.

[0385] For example, the decoder side obtains motion vectors of two control points (namely, a 2-tuple) in the step 703b. The motion vectors of the two control points are, for example, a motion vector (vx_0, vy_0) of the top-left control point (x_0, y_0) of the current block and a motion vector (vx_1, vy_1) of the top-right control point (x_1, y_1) of the current block. Then, the 4-parameter affine transform model is constructed based on the motion vectors of the two control points of the current block. A motion vector of a third control point is obtained according to a formula (31) for the 4-parameter affine transform model. The motion vector of the third control point is, for example, a motion vector (vx_2, vy_2) of the bottom-left corner (x_2, y_2) of the current block. In this way, the motion vectors of the top-left control point, the top-right control point, and the bottom-left control point of the current block are determined.

[0386] For another example, the decoder side obtains motion vectors of three control points (namely, a triplet) in the step 703b. The motion vectors of the three control points are, for example, a motion vector (vx_0, vy_0) of the top-left control point (x_0, y_0) of the current block, a motion vector (vx_1, vy_1) of the top-right control point (x_1, y_1) of the current block, and a motion vector (vx_2, vy_2) of the bottom-left corner (x_2, y_2) of the

current block. In this way, the motion vectors of the top-left control point, the top-right control point, and the bottom-left control point of the current block are determined.

[0387] For another example, the decoder side obtains motion vectors of four control points (namely, a quadruple) in the step 703b. The motion vectors of the four control points are, for example, a motion vector (vx_0 , vy_0) of the top-left control point (x_0 , y_0) of the current block, a motion vector (vx_1 , vy_1) of the top-right control point (x_1 , y_1) of the current block, a motion vector (vx_2 , vy_2) of the bottom-left corner (x_2 , y_2) of the current block, and a motion vector (vx_3 , vy_3) of the bottom-right corner (x_3 , vy_3) of the current block. Then, the decoder side may use only motion vectors of the top-left control point, the top-right control point, and the bottom-left control point of the current block.

[0388] It should be noted that the technical solutions of the present invention are not limited to the foregoing examples, and another control point and motion model may also be applicable to the present invention. Details are not described herein.

[0389] Step 705b: Obtain a motion vector of each subblock based on the three control points of the current block and by using the 6-parameter affine transform model. For a specific implementation of this step, refer to related descriptions in the step 705a. Details are not described herein again.

[0390] Step 706b: Perform motion compensation for each subblock based on the corresponding motion vector, to obtain a pixel predictor of the subblock.

[0391] It can be learned that, in this embodiment of the present invention, the decoder side uses the second motion model-based motion vector prediction method in a process of predicting the current block. In this way, in a parsing phase, a quantity of parameters of an affine transform model used for the current block may be different from or the same as that of an affine transform model used for a neighboring block, and the 6-parameter affine transform model can be uniformly used to predict the current block in the phase of reconstructing the current block (including a phase of predicting a motion vector of a subblock). In this solution, the 6-parameter affine transform model constructed in the reconstruction phase can describe affine transform such as translation, scaling, and rotation of an image block, and achieve a good balance between model complexity and a modeling capability. Therefore, this solution can improve coding efficiency and accuracy in predicting the current block, and satisfy a user requirement.

[0392] FIG. 13 is a flowchart of still another motion vector prediction method according to an embodiment of the present invention. The method may be performed by a video encoder 100, and specifically, may be performed by an inter predictor 110 of the video encoder 100. The video encoder 100 may perform, based on a video data stream having a plurality of video frames, a part or all of the following steps to encode a current coding block (referred to as a current block for short) of a current video frame. As shown in FIG. 13, the method includes but is not limited to

the following steps.

[0393] 801: Determine an inter prediction mode of the current coding block.

[0394] In a specific implementation, a plurality of inter prediction modes may be preset for inter prediction on an encoder side. For example, the plurality of inter prediction modes include the affine motion model-based AMVP mode and the affine motion model-based merge mode that are described above. The encoder side traverses the plurality of inter prediction modes to determine an optimal inter prediction mode for predicting the current block.

[0395] In another specific implementation, only one inter prediction mode may be preset for inter prediction on an encoder side. In this case, the encoder side directly determines that a default inter prediction mode is currently used. The default inter prediction mode is the affine motion model-based AMVP mode or the affine motion model-based merge mode.

[0396] In this embodiment of the present invention, if it is determined that the inter prediction mode of the current block is the affine motion model-based AMVP mode, steps 802a to 804a are performed subsequently.

[0397] In this embodiment of the present invention, if it is determined that the inter prediction mode of the current block is the affine motion model-based merge mode, step 802b to step 804b are performed subsequently.

[0398] 802a: Construct a candidate motion vector list for the affine transform-based AMVP mode.

[0399] In some embodiments, the encoder side uses a design solution of a first motion model-based motion vector prediction method. Therefore, for specific implementation of this step, refer to descriptions of the step 602a in the embodiment in FIG. 9. Details are not described herein again.

[0400] In some other embodiments, the encoder side uses a design solution of a second motion model-based motion vector prediction method. Therefore, for specific implementation of this step, refer to descriptions of the step 702a in the embodiment in FIG. 12. Details are not described herein again.

[0401] 803a: Determine an optimal motion vector predictor of a control point based on a rate-distortion cost.

[0402] In some examples, the encoder side may obtain a motion vector of each motion compensation subunit of the current block by using a control point motion vector predictor (for example, a candidate motion vector 2-tuple/triplet/quadruple) in the candidate motion vector list and according to the formula (3), (5), or (7). Further, the encoder side obtains a pixel value of a corresponding location in a reference frame to which the motion vector of each motion compensation subunit points, and uses the pixel value as a pixel predictor of the motion compensation subunit to perform affine motion model-based motion compensation. An average difference between an original value and a predictor of each sample in the current coding block is calculated. A control point motion vector predictor corresponding to a minimum average difference is selected as an optimal control point motion vector predictor, and used as motion vector predictors of two, three, or four control points of the current block.

CLAIMS

1. A motion vector prediction method, wherein the method comprises:

parsing a bitstream to obtain an index value of a candidate motion vector list;

when an affine model of a current block is a $2 \times K$ -parameter affine transform model, and an affine transform

5 model of a neighboring block of the current block is a $2 \times N$ -parameter affine transform model, obtaining, according to the $2 \times N$ -parameter affine transform model used for the neighboring block, candidate motion vectors of K control points of the current block; the $2 \times N$ -parameter affine transform model is obtained based on motion vectors of N control points of the neighboring block, N is an integer greater than or equal to 2 and less than or equal to 4, K is an integer greater than or equal to 2 and less than or equal to 4, N is not equal to K , the neighboring block is a decoded
10 image block spatially adjacent to the current block, and the current block comprises a plurality of subblocks;

constructing the candidate motion vector list, wherein the candidate motion vector list comprises candidate motion vectors of K control points of a current block;

determining, in the candidate motion vector list, target candidate motion vectors of the K control points based on the index value; and

15 obtaining a predicted motion vector of each subblock of the current block based on the target candidate motion vectors of the K control points.

2. The method according to claim 1, wherein N is equal to 2 and K is equal to 3, and correspondingly, candidate motion vectors of three control points of the current block are obtained based on a 4-parameter affine transform model used for the neighboring block of the current block.

20 3. The method according to claim 2, wherein the candidate motion vectors of the three control points of the current block comprise a motion vector at a top-left sample location of the current block, a motion vector at a top-right sample location of the current block, and a motion vector at a bottom-left sample location of the current block; and

that the candidate motion vectors of the three control points of the current block are obtained based on the
25 4-parameter affine transform model used for the neighboring block of the current block comprise that the candidate motion vectors the three control points of the current block are calculated according to the following formulas:

$$\begin{cases} vx_0 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_0 - x_4) - \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (y_0 - y_4) \\ vy_0 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (y_0 - y_4) \end{cases}$$

$$\begin{cases} vx_1 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_1 - x_4) - \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (y_1 - y_4) \\ vy_1 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (y_1 - y_4) \end{cases}$$

$$\begin{cases} vx_2 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_2 - x_4) - \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (y_2 - y_4) \\ vy_2 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_2 - x_4) + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (y_2 - y_4) \end{cases}$$

wherein vx_0 is a horizontal component of the motion vector corresponding to the top-left sample location of the current block, and vy_0 is a vertical component of the motion vector corresponding to the top-left sample location of the current block; vx_1 is a horizontal component of the motion vector corresponding to the top-right sample location of the current block, and vy_1 is a vertical component of the motion vector corresponding to the top-right sample location of the current block; vx_2 is a horizontal component of the motion vector corresponding to the bottom-left sample location of the current block, and vy_2 is a vertical component of the motion vector corresponding to the bottom-left sample location of the current block; vx_4 is a horizontal component of a motion vector corresponding to a top-left sample location of the neighboring block, and vy_4 is a vertical component of the motion vector corresponding to the top-left sample location of the neighboring block; vx_5 is a horizontal component of a motion vector corresponding to a top-right sample location of the neighboring block, and vy_5 is a vertical component of the motion vector corresponding to the top-right sample location of the neighboring block; x_0 is a horizontal coordinate of the top-left sample location of the current block, and y_0 is a vertical coordinate of the top-left sample location of the current block; x_1 is a horizontal coordinate of the top-right sample location of the current block, and y_1 is a vertical coordinate of the top-right sample location of the current block; x_2 is a horizontal coordinate of the bottom-left sample location of the current block, and y_2 is a vertical coordinate of the bottom-left sample location of the current block; x_4 is a horizontal coordinate of the top-left sample location of the neighboring block, and y_4 is a vertical coordinate of the top-left sample location of the neighboring block; and x_5 is a horizontal coordinate of the top-right sample location of the neighboring block.

4. The method according to claim 1, wherein N is equal to 3 and K is equal to 2, and correspondingly, candidate motion vectors of two control points of the current block are obtained based on a 6-parameter affine transform

model used for the neighboring block of the current block.

5 The method according to claim 4, wherein the candidate motion vectors of the two control points of the current block comprise a motion vector at a top-left sample location of the current block and a motion vector at a top-right sample location of the current block; and that the candidate motion vectors of the two control points of the current block are obtained based on the 6-parameter affine transform model used for the neighboring block of the current block comprises that the candidate motion vectors of the two control points of the current block are calculated according to the following formulas:

$$\begin{cases} vx_0 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vx_6 - vx_4)}{y_6 - y_4} \times (y_0 - y_4) \\ vy_0 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vy_6 - vy_4)}{y_6 - y_4} \times (y_0 - y_4) \\ vx_1 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vx_6 - vx_4)}{y_6 - y_4} \times (y_1 - y_4) \\ vy_1 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vy_6 - vy_4)}{y_6 - y_4} \times (y_1 - y_4) \end{cases}$$

10 wherein vx_0 is a horizontal component of the motion vector corresponding to the top-left sample location of the current block, and vy_0 is a vertical component of the motion vector corresponding to the top-left sample location of the current block; vx_1 is a horizontal component of the motion vector corresponding to the top-right sample location of the current block, and vy_1 is a vertical component of the motion vector corresponding to the top-right sample location of the current block; vx_4 is a horizontal component of a motion vector corresponding to a top-left sample location of the neighboring block, and vy_4 is a vertical component of the motion vector corresponding to the top-left sample location of the neighboring block; vx_5 is a horizontal component of a motion vector corresponding to a top-right sample location of the neighboring block, and vy_5 is a vertical component of the motion vector corresponding to the top-right sample location of the neighboring block; vx_6 is a horizontal component of a motion vector corresponding to a bottom-left sample location of the neighboring block, and vy_6 is a vertical component of the motion vector corresponding to the bottom-left sample location of the neighboring block; x_0 is a horizontal coordinate of the top-left sample location of the current block, and y_0 is a vertical coordinate of the top-left sample location of the current block; x_1 is a horizontal coordinate of the top-right sample location of the current block, and y_1 is a vertical coordinate of the top-right sample location of the current block;

x_4 is a horizontal coordinate of the top-left sample location of the neighboring block, and y_4 is a vertical coordinate of the top-left sample location of the neighboringcurrent block; x_5 is a horizontal coordinate of the top-right sample location of the neighboring block; and y_6 is a vertical coordinate of the bottom-left sample location of the neighboring block.

5 6. The method according to any one of claims 1 to 5, wherein the obtaining a predicted motion vector of each subblock of the current block based on the target candidate motion vectors of the K control points comprises:

obtaining a $2 \times K$ -parameter affine transform model of the current block based on the target candidate motion vectors of the K control points; and

10 obtaining the predicted motion vector of each subblock of the current block based on the $2 \times K$ -parameter affine transform model.

7. The method according to claim 6, wherein the obtaining a $2 \times K$ -parameter affine transform model based on the target candidate motion vectors of the K control points comprises:

15 obtaining motion vectors of the K control points based on the target candidate motion vectors of the K control points and motion vector differences of the K control points, wherein the motion vector differences of the K control points are obtained by parsing the bitstream; and

obtaining the $2 \times K$ -parameter affine transform model of the current block based on the motion vectors of the K control points.

20 8. The method according to any one of claims 1 to 5, wherein after the determining, in the candidate motion vector list, target candidate motion vectors of the K control points of the current block based on the index value, the method further comprises: obtaining the 6-parameter affine transform model of the current block based on the target candidate motion vectors of the K control points of the current block; and

correspondingly, the obtaining a predicted motion vector of each subblock of the current block based on the target candidate motion vectors of the K control points comprises:

25 obtaining the predicted motion vector of each subblock of the current block based on the 6-parameter affine transform model of the current block.

9. A decoding device, wherein the device comprises:

a storage unit, configured to store video data in a form of a bitstream;

an entropy decoding unit, configured to parse the bitstream to obtain an index value of a candidate motion vector list; and

30 a prediction processing unit, configured to: when an affine model of a current block is a $2 \times K$ -parameter affine

transform model, and an affine transform model of a neighboring block of the current block is a $2 \times N$ -parameter affine transform model, obtain, according to the $2 \times N$ -parameter affine transform model used for the neighboring block, candidate motion vectors of K control points of the current block; the $2 \times N$ -parameter affine transform model is obtained based on motion vectors of N control points of the neighboring block, N is an integer greater than or equal to 2 and less than or equal to 4, K is an integer greater than or equal to 2 and less than or equal to 4, N is not equal to K , the neighboring block is a decoded image block spatially adjacent to the current block, and the current block comprises a plurality of subblocks; and construct the candidate motion vector list, wherein the candidate motion vector list comprises candidate motion vectors of K control points of a current block; determine, in the candidate motion vector list, target candidate motion vectors of the K control points of the current block based on the index value; and obtain a predicted motion vector of each subblock of the current block based on the target candidate motion vectors of the K control points of the current block.

10. The device according to claim 9, wherein N is equal to 2 and K is equal to 3, and correspondingly, candidate motion vectors of three control points of the current block are obtained based on a 4-parameter affine transform model used for the neighboring block of the current block.

11. The device according to claim 10, wherein the candidate motion vectors of the three control points of the current block comprise a motion vector at a top-left sample location of the current block, a motion vector at a top-right sample location of the current block, and a motion vector at a bottom-left sample location of the current block; and

the prediction processing unit is configured to calculate the candidate motion vectors of the three control points of the current block according to the following formulas:

$$\begin{cases} vx_0 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_0 - x_4) - \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (y_0 - y_4) \\ vy_0 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (y_0 - y_4) \end{cases}$$

$$\begin{cases} vx_1 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_1 - x_4) - \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (y_1 - y_4) \\ vy_1 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (y_1 - y_4) \end{cases}$$

$$\begin{cases} vx_2 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_2 - x_4) - \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (y_2 - y_4) \\ vy_2 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_2 - x_4) + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (y_2 - y_4) \end{cases}$$

wherein vx_0 is a horizontal component of the motion vector corresponding to the top-left sample location of the current block, and vy_0 is a vertical component of the motion vector corresponding to the top-left sample

location of the current block; vx_1 is a horizontal component of the motion vector corresponding to the top-right sample location of the current block, and vy_1 is a vertical component of the motion vector corresponding to the top-right sample location of the current block; vx_2 is a horizontal component of the motion vector corresponding to the bottom-left sample location of the current block, and vy_2 is a vertical component of the motion vector corresponding to the bottom-left sample location of the current block; vx_4 is a horizontal component of a motion vector corresponding to a top-left sample location of the neighboring block, and vy_4 is a vertical component of the motion vector corresponding to the top-left sample location of the neighboring block; vx_5 is a horizontal component of a motion vector corresponding to a top-right sample location of the neighboring block, and vy_5 is a vertical component of the motion vector corresponding to the top-right sample location of the neighboring block;

x_0 is a horizontal coordinate of the top-left sample location of the current block, and y_0 is a vertical coordinate of the top-left sample location of the current block; x_1 is a horizontal coordinate of the top-right sample location of the current block, and y_1 is a vertical coordinate of the top-right sample location of the current block; x_2 is a horizontal coordinate of the bottom-left sample location of the current block, and y_2 is a vertical coordinate of the bottom-left sample location of the current block; x_4 is a horizontal coordinate of the top-left sample location of the neighboring block, and y_4 is a vertical coordinate of the top-left sample location of the neighboring block; and x_5 is a horizontal coordinate of the top-right sample location of the neighboring block.

12. The device according to claim 9, wherein N is equal to 3 and K is equal to 2, and correspondingly, candidate motion vectors of two control points of the current block are obtained based on a 6-parameter affine transform model used for the neighboring block of the current block.

13. The device according to claim 12, wherein the candidate motion vectors of the two control points of the current block comprise a motion vector at a top-left sample location of the current block and a motion vector at a top-right sample location of the current block; and

the prediction processing unit is configured to calculate the candidate motion vectors of the two control points of the current block according to the following formulas:

$$\begin{cases} vx_0 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vx_6 - vx_4)}{y_6 - y_4} \times (y_0 - y_4) \\ vy_0 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_0 - x_4) + \frac{(vy_6 - vy_4)}{y_6 - y_4} \times (y_0 - y_4) \end{cases}$$

$$\begin{cases} vx_1 = vx_4 + \frac{(vx_5 - vx_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vx_6 - vx_4)}{y_6 - y_4} \times (y_1 - y_4) \\ vy_1 = vx_4 + \frac{(vy_5 - vy_4)}{x_5 - x_4} \times (x_1 - x_4) + \frac{(vy_6 - vy_4)}{y_6 - y_4} \times (y_1 - y_4) \end{cases}$$

wherein vx_0 is a horizontal component of the motion vector corresponding to the top-left sample location of the current block, and vy_0 is a vertical component of the motion vector corresponding to the top-left sample location of the current block; vx_1 is a horizontal component of the motion vector corresponding to the top-right sample location of the current block, and vy_1 is a vertical component of the motion vector corresponding to the top-right sample location of the current block; vx_4 is a horizontal component of a motion vector corresponding to a top-left sample location of the neighboring block, and vy_4 is a vertical component of the motion vector corresponding to the top-left sample location of the neighboring block; vx_5 is a horizontal component of a motion vector corresponding to a top-right sample location of the neighboring block, and vy_5 is a vertical component of the motion vector corresponding to the top-right sample location of the neighboring block; vx_6 is a horizontal component of a motion vector corresponding to a bottom-left sample location of the neighboring block, and vy_6 is a vertical component of the motion vector corresponding to the bottom-left sample location of the neighboring block; x_0 is a horizontal coordinate of the top-left sample location of the current block, and y_0 is a vertical coordinate of the top-left sample location of the current block; x_1 is a horizontal coordinate of the top-right sample location of the current block, and y_1 is a vertical coordinate of the top-right sample location of the current block; x_4 is a horizontal coordinate of the top-left sample location of the neighboring block, and y_4 is a vertical coordinate of the top-left sample location of the neighboring current block; x_5 is a horizontal coordinate of the top-right sample location of the neighboring block; and y_6 is a vertical coordinate of the bottom-left sample location of the neighboring block.

14. The device according to any one of claims 9 to 13, wherein the prediction processing unit is specifically configured to:

obtain a $2 \times K$ -parameter affine transform model of the current block based on the target candidate motion vectors of the K control points of the current block; and

obtain the predicted motion vector of each subblock of the current block based on the $2 \times K$ -parameter affine transform model of the current block.

5 15. The device according to claim 14, wherein the prediction processing unit is specifically configured to:

obtain motion vectors of the K control points of the current block based on the target candidate motion vectors of the K control points of the current block and motion vector differences of the K control points of the current block, wherein the motion vector differences of the K control points of the current block are obtained by parsing the bitstream; and

10 obtain the $2 \times K$ -parameter affine transform model of the current block based on the motion vectors of the K control points of the current block.

16. The device according to any one of claims 9 to 13, wherein after determining, in the candidate motion vector list, the target candidate motion vectors of the K control points of the current block based on the index value, the prediction processing unit is further configured to:

15 obtain the 6-parameter affine transform model of the current block based on the target candidate motion vectors of the K control points of the current block; and

obtain the predicted motion vector of each subblock of the current block based on the 6-parameter affine transform model of the current block.